AVLSI

Spring – 2024 Course Project

Name: Eswar Sri Satya Sai Kadali

Roll no: S20210020271, **Email**: eswar.k21@iiits.in

ECE, Indian Institute of Information Technology, Sri city

Quantum-Dot Cellular Automata Layout Generator And Cross-bar Architecture Circuits

Abstract:

Quantum-dot Cellular Automata (QCA) is emerging as a promising alternative to CMOS transistor technology, offering the potential for implementing logic circuits using quantum devices such as quantum dots or single-domain nanomagnets. However, the design and implementation of QCA circuits require specialized tools due to their unique characteristics. While tools like QCADesigner exist for manual layout and simulation, there is a lack of automated tools for QCA layout generation. In response to this need, we propose the QCA-Layout Generator (QCA-LG), a tool integrated into a general QCA technology design flow. QCA-LG accepts commonly used synthesis tool formats and automatically generates layouts according to QCADesigner standards. This allows logical circuits described in VHDL to be automatically converted into layouts, which can then be further optimized by hand and simulated using QCADesigner.

Also, a novel design methodology tailored specifically for QCA circuits, based on universal design rules for QCA and a generic QCA crossbar architecture. This methodology provides a systematic approach to customize the crossbar architecture for implementing any desired logic function. By applying this methodology, we demonstrate the effectiveness, user-friendliness, and unification of QCA circuit design processes for both combinatorial and sequential circuits.

Introduction:

In the near future, the current CMOS technology may not be able to keep up with the industry's growth rates. Quantum-dot Cellular Automata (QCA) offers a promising alternative, as it can be implemented using various physical systems. Quantum dots, in particular, have shown great potential due to their operational speed and integration density. This project focuses on QCA implemented with quantum dots, similar to QCA

Designer. While software tools have been developed to aid QCA circuit and layout design, existing methods for logic mapping to majority gates are suboptimal. To address this, the paper proposes the QCA-LG tool, designed to automatically generate layouts for QCA combinational circuits. The tool maps logic circuits to QCA gates and produces corresponding layouts, supporting popular netlist formats and QCA Designer's layout format. This advancement allows for the visualization and physical simulation of QCA layouts for the first time.

Furthermore, we introduce a detailed methodology for the robust and efficient design of QCA circuits using the proposed architecture. This methodology considers the input/outputs as well as the programming lines of the crossbar architecture to implement Boolean functions and standard QCA circuits.

Background:

Quantum-dot Cellular Automata (QCA) is an innovative computing paradigm that leverages quantum dots to perform computations. In this framework, logic states are defined by the positions of electrons within quantum dots, enabling rapid and energy-efficient operations. A fundamental QCA cell comprises four quantum dots arranged in a square configuration, with two electrons capable of tunnelling between them. These cells interact through electrostatic forces, enabling the implementation of logic gates. The majority gate is a crucial component of QCA logic, producing an output based on the majority value of its inputs. By arranging QCA cells in specific configurations, traditional logic gates such as AND, NOT, and OR can be constructed.

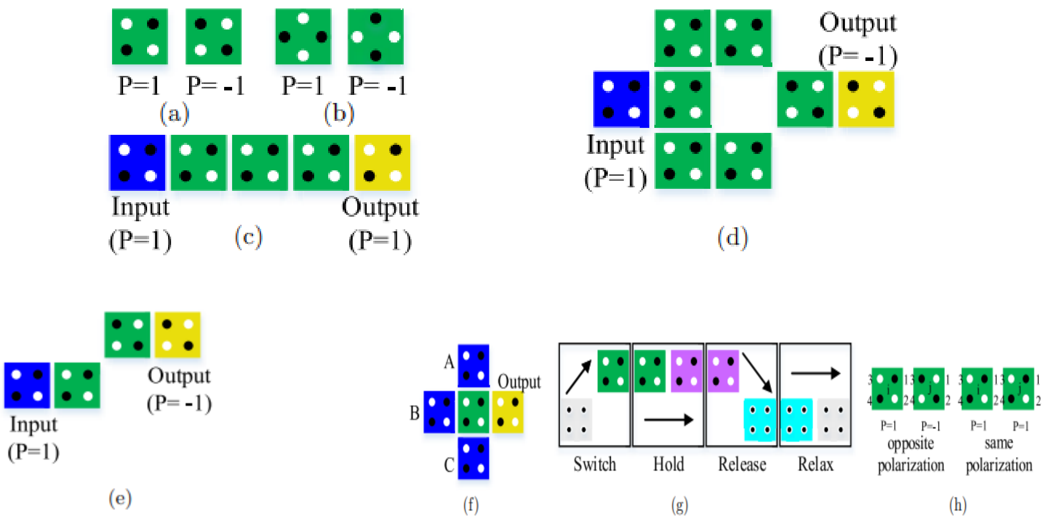


Fig. 1. (a) Normal cells, (b) rotated cells, (c) QCA binary wire, (d) inverter gate, (e) alternate inverter gate, (f) majority gate, (g) QCA clocking and (h) kink energy

The proposed methodology for combinatorial QCA circuit design addresses the lack of a universal design methodology in QCA circuits, aiming to resolve compatibility issues between reported QCA circuits. It utilizes a generic programmable crossbar architecture of quantum-dot cells and follows fundamental design rules. The architecture consists of an array of quantum-dot cells with rules for mapping digital circuits onto the crossbar, handling inputs and outputs, forming logic gates at cross points, and reconfiguring gate operations during circuit operation. The majority gate, a basic logic gate in QCA, is crucial for implementing OR and AND gates. The architecture's straightforward information flow, well-defined I/O interface, fixed cell positions, and programmability make it suitable for automated QCA circuit design, ensuring scalability and productivity.

(a) (b)

Fig. 2. (a) Cross-bar Majority gate, (b) Cross-bar Invertor

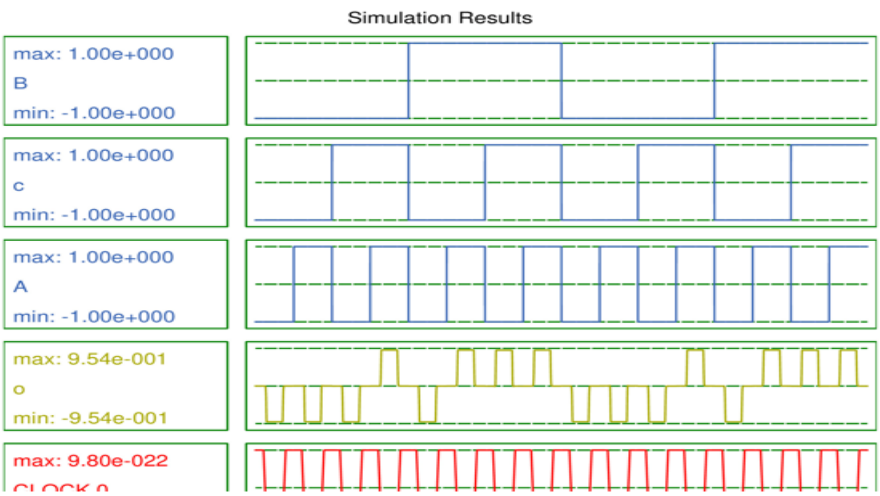


Fig. 3. Fig.2. (a) Cross-bar Majority gate simulation results

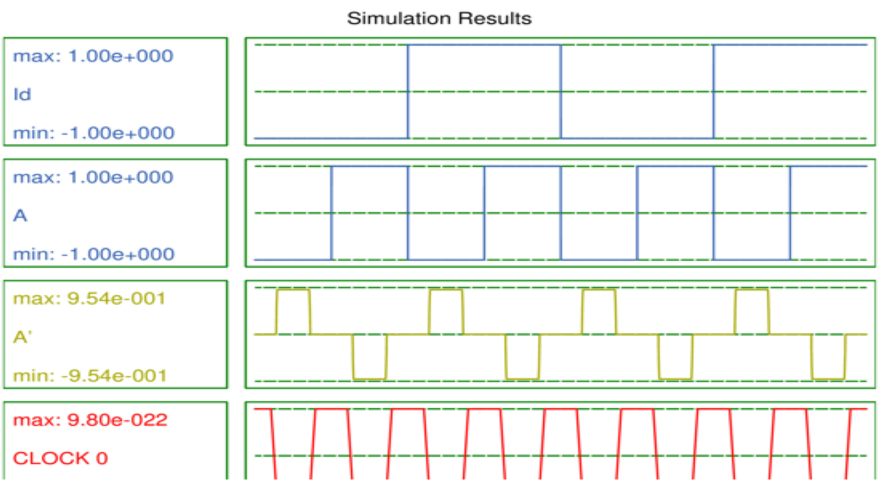


Fig. 4. Fig.2. (b) Cross-bar Invertor simulation results

The method organizes the flow of information in the circuit from left to right, achieved by dividing the circuit into clock zones and using a technique called adiabatic switching. This technique controls how electrons move in the cells using four-phase clock signals, ensuring that information moves smoothly and predictably. To design the circuit, the method rewrites the logic function using Boolean algebra and places blocks based on their logical hierarchy. Blocks are arranged in the crossbar space; with their positions determined by the number of inputs they have. These blocks are divided into two categories: those that perform basic logic operations and those that help with signal routing. The method also ensures that all blocks have consistent delays, making it easier to synchronize the circuit and minimizing delays caused by signal routing.

In summary, the methodology transforms logic functions for QCA implementation, partitions circuits into levels and sub-levels, and places blocks onto the crossbar grid using top and bottom programming lines. This systematic approach enables the design of stable and functional QCA circuits with minimal delay, addressing compatibility issues and providing a generic solution for automated QCA circuit design.

AUTOMATED QCA LAYOUTS GENERATION:

Our main aim is to automate the creation of a layout for a quantum dot circuit that works with the QCA Designer tool. The process flow is illustrated in Figure 3. We've designed the program to read data from files that represent logic circuits in two different formats: LSI, used by Synopsys synthesis tools, and Gate, used by MVSIS and its successor, SIS.

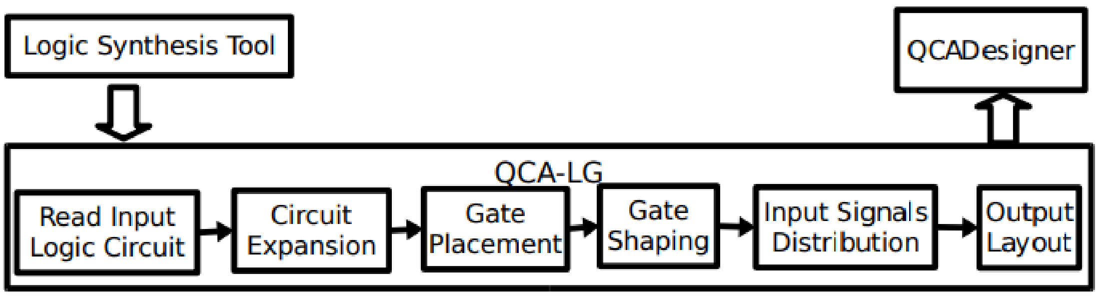
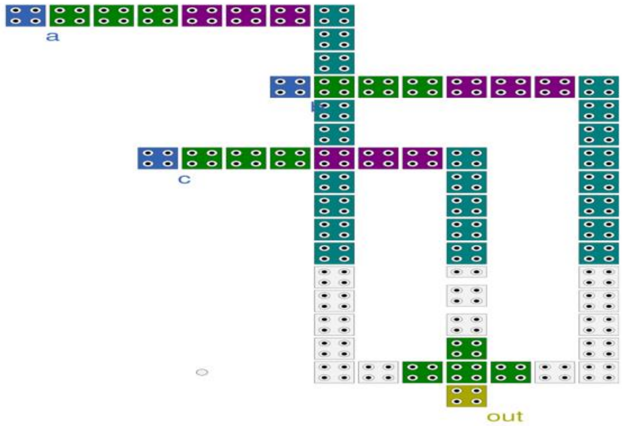


Fig. 5. Flowchart depicting the tool's operations.

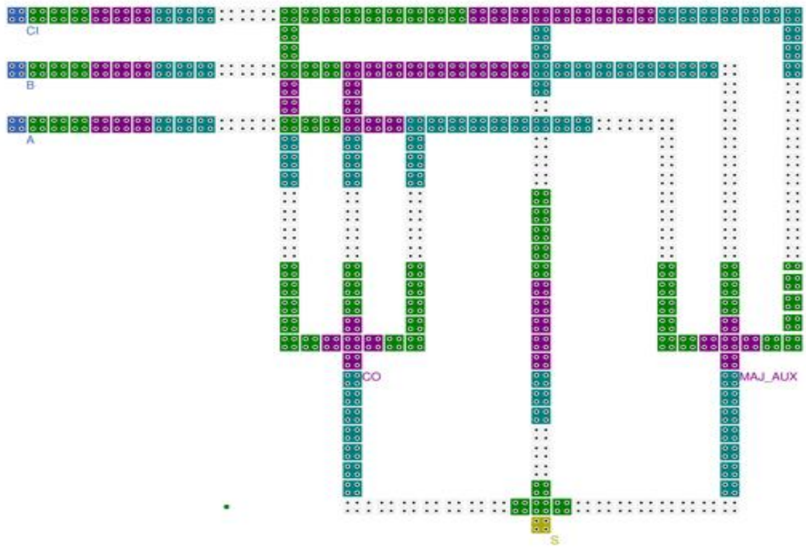
To handle these formats, we've developed parsers using Lex & Yacc. These tools make it easier to add support for new formats in the future. To generate a netlist based only on the components supported by the design, we've created libraries containing just those components. These libraries include majority gates, NOT gates, and 2-input AND, OR, NAND, and NOR gates. Internally, the circuits are represented as directed graphs and stored in a hash table, where each object represents a gate in the circuit or a primary input.

AUTOMATED QCA CIRCUITS:

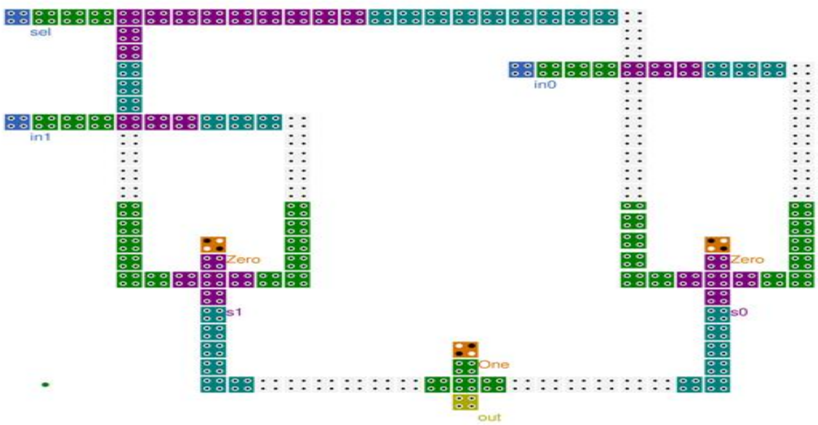
Some of the circuits which are generated are displayed below.



(a)

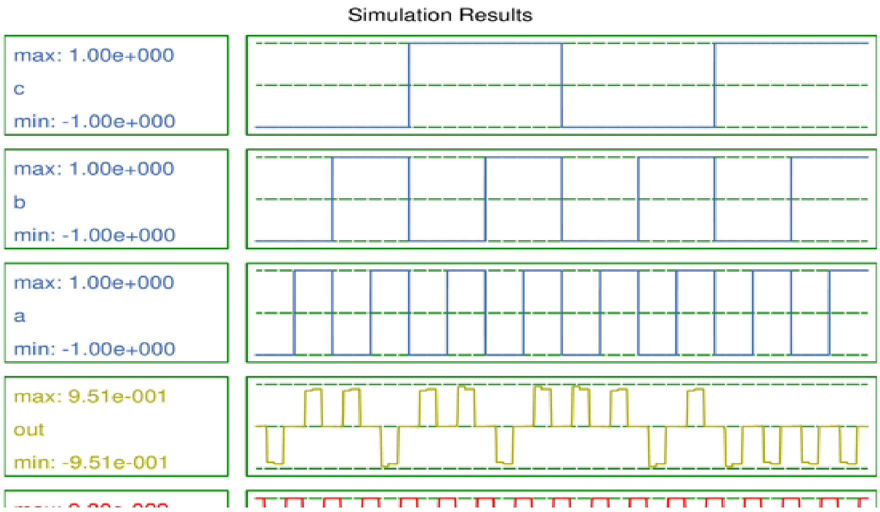


(b)

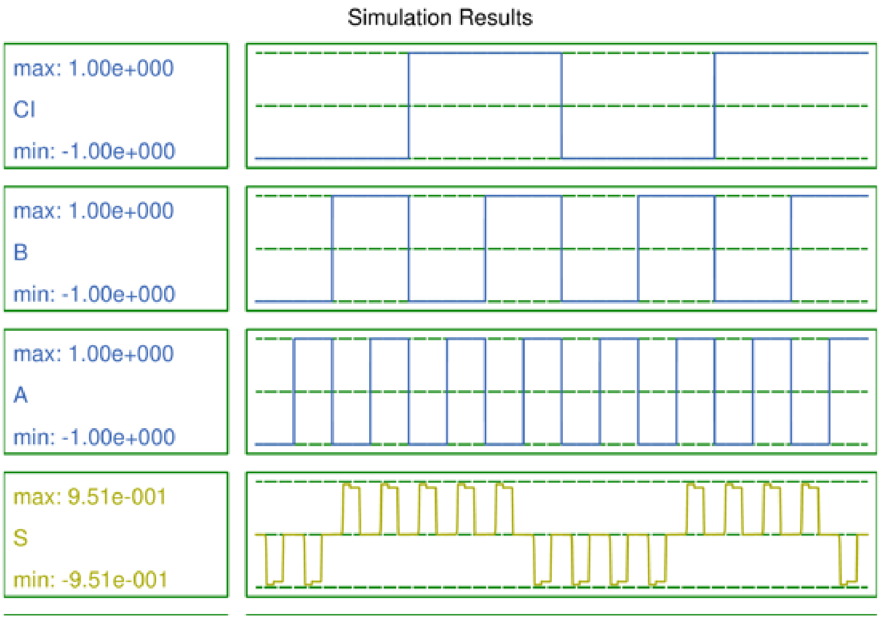


(c)

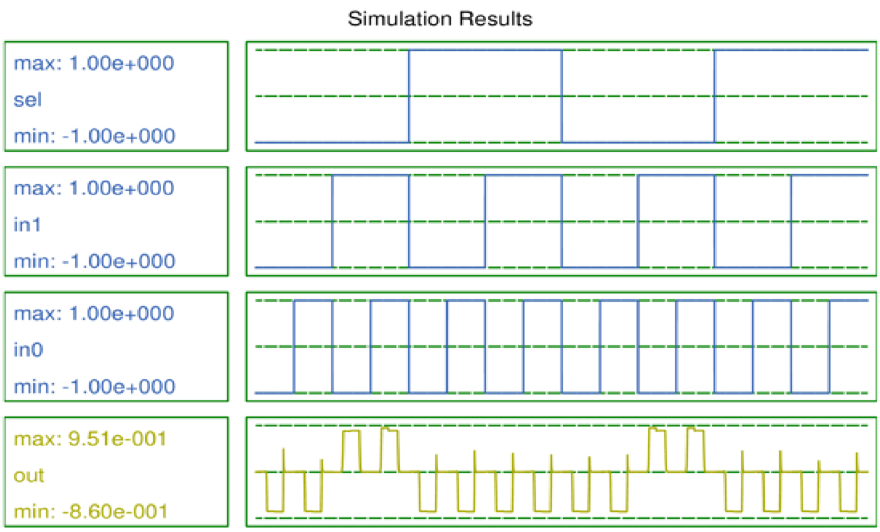
Fig. 6. (a)Simple circuit , (b) Adder, (c) 2:1 Mux



(a)



(b)

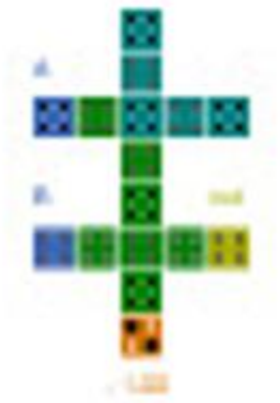


(c)

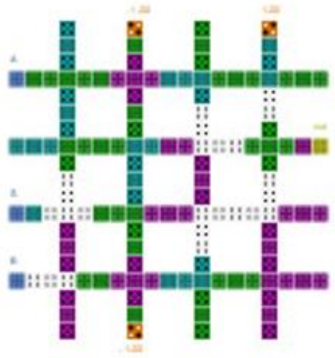
Fig. 7. Fig.6 (a)Simple circuit, (b) Adder, (c) 2:1 Mux Simulation results.

APPLICATION OF DESIGN METHODOLOGY:

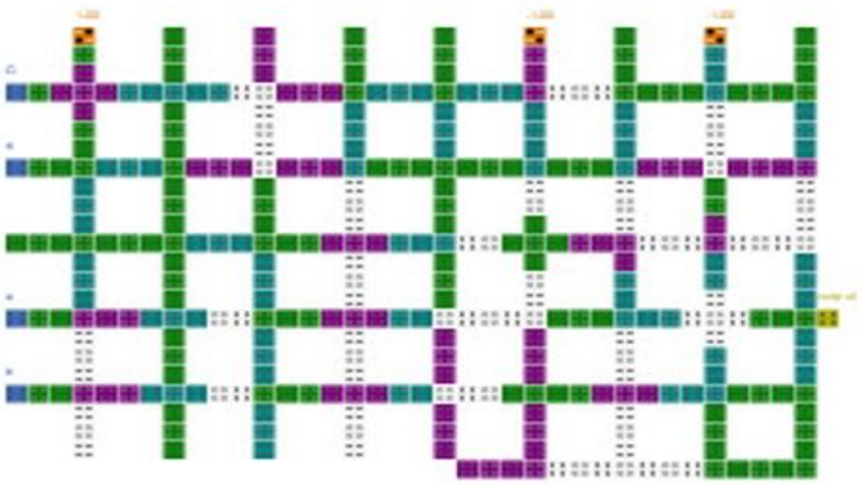
The design methodology presented is applied to create a Boolean function F=a'.b, F = a . b . (c + d), 2:1 and 4:1 multiplexer using Quantum-dot Cellular Automata (QCA).



(a)



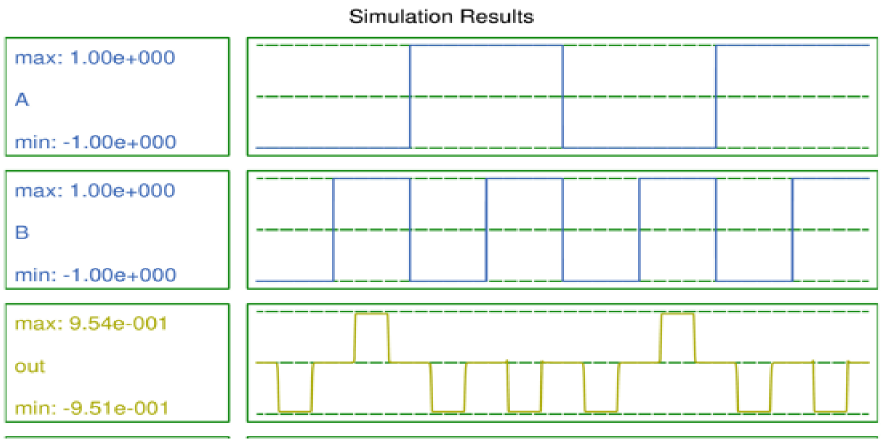
(b)



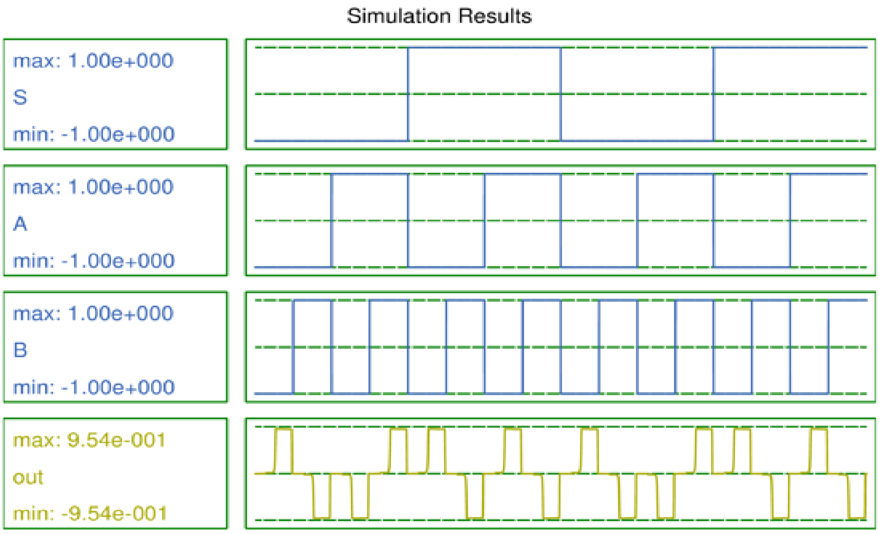
(c)

Fig. 8. (a)F= a'.b, (b) 2:1 mux , (c) F=a.b.(c+d).

(a)



(b)



(c)

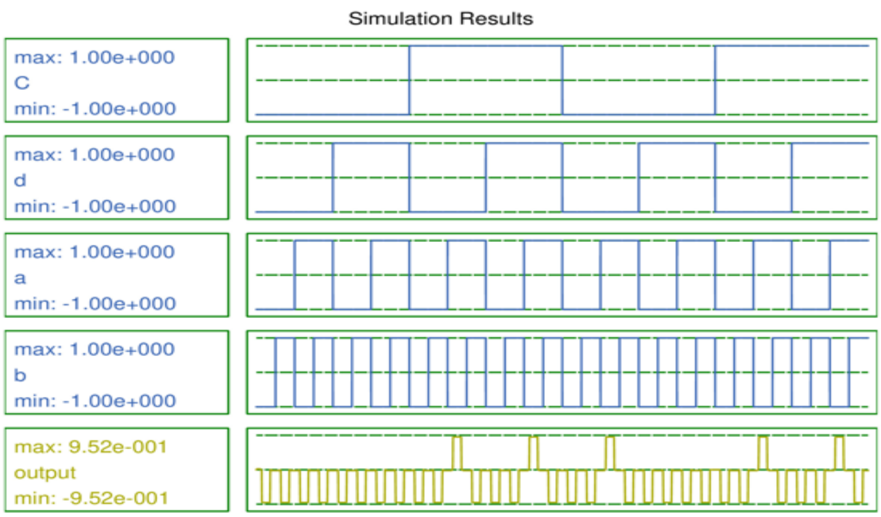


Fig.9 . Simulation results for Fig.8 (a)F= a'.b, (b) 2:1 mux , (c) F=a.b.(c+d).

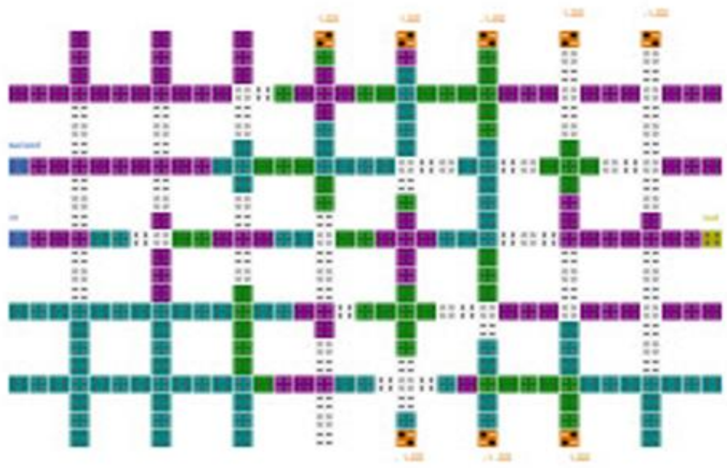


Figure 10. Crossbar mapped QCA Ram cell in writing mode.

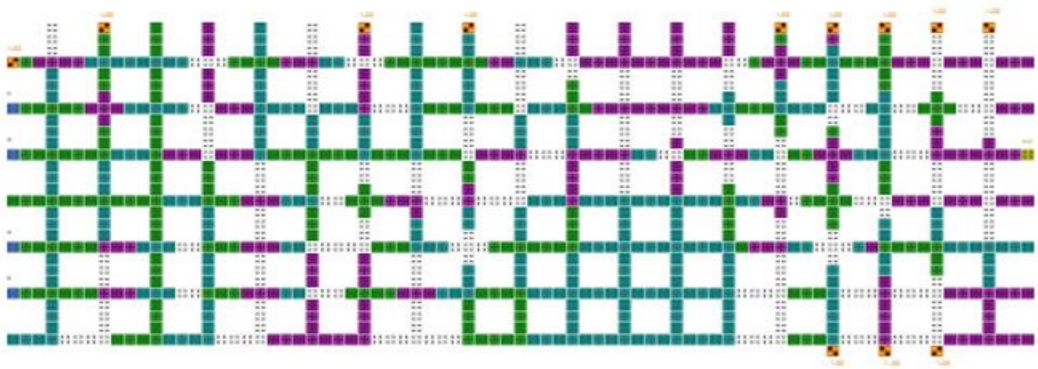


Figure 11. a + b + (c + d) QCA circuit with memory element.

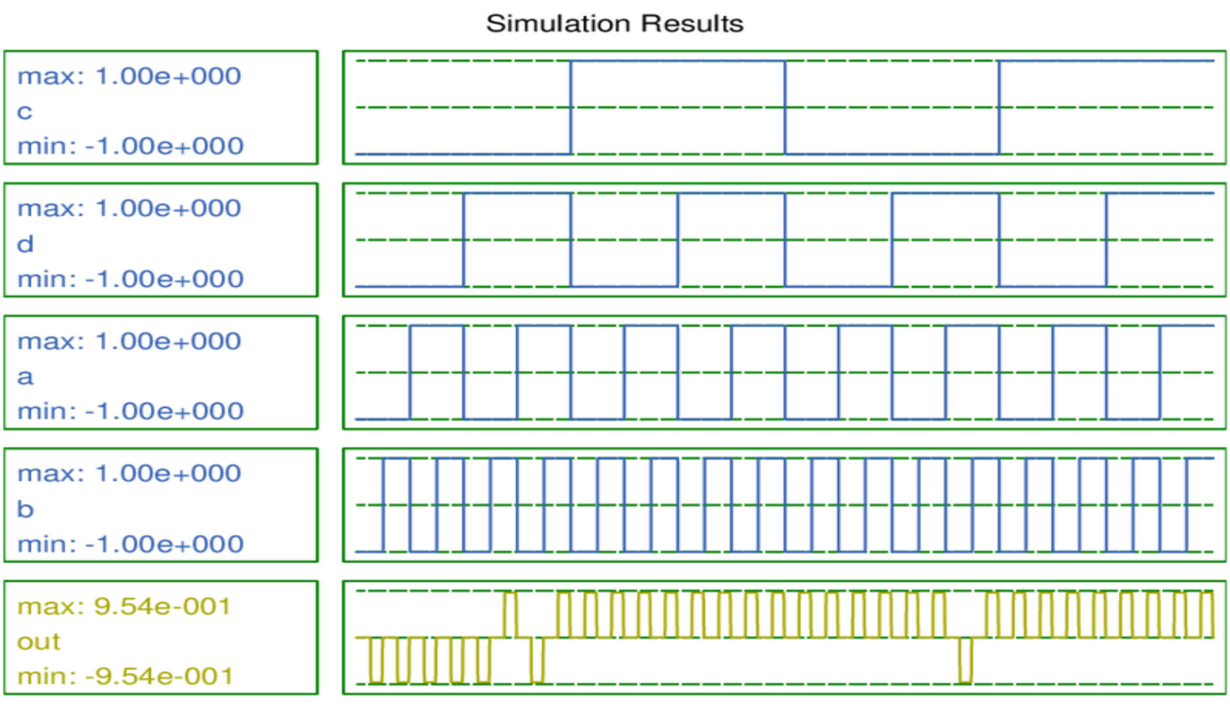


Figure 12. a + b + (c + d) with memory element simulation results.

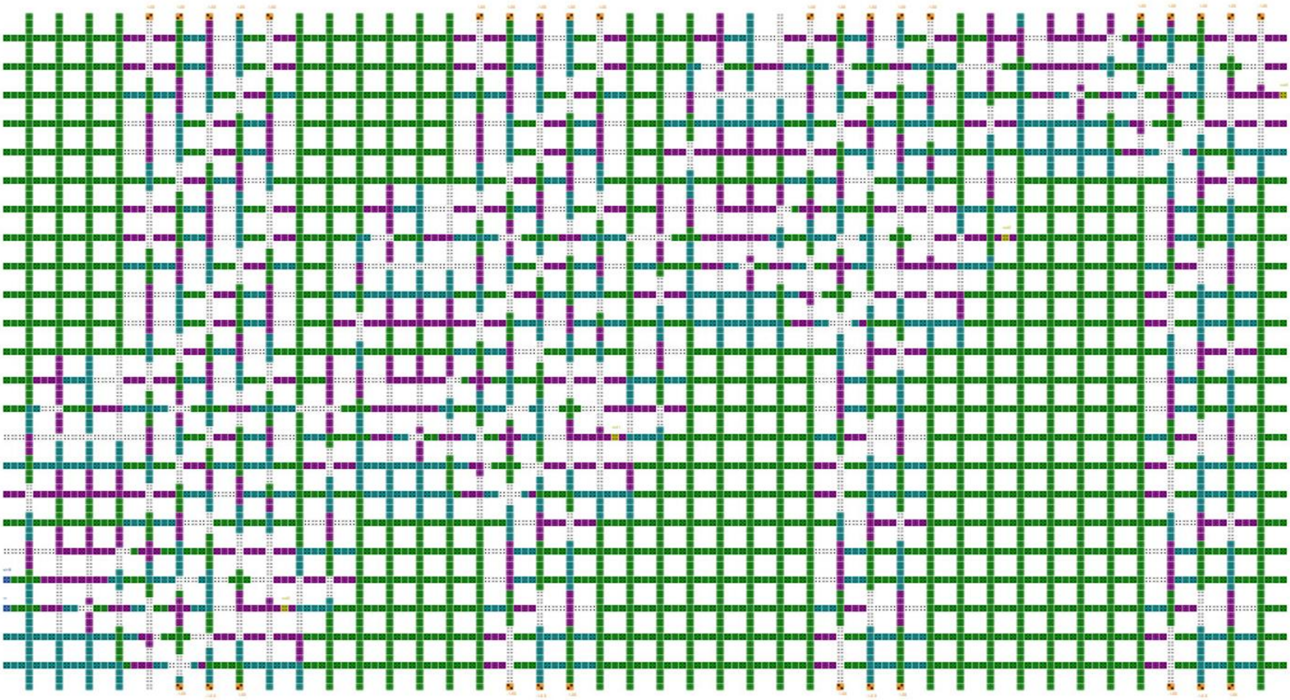


Figure 13. 4-bit shift register QCA circuit.

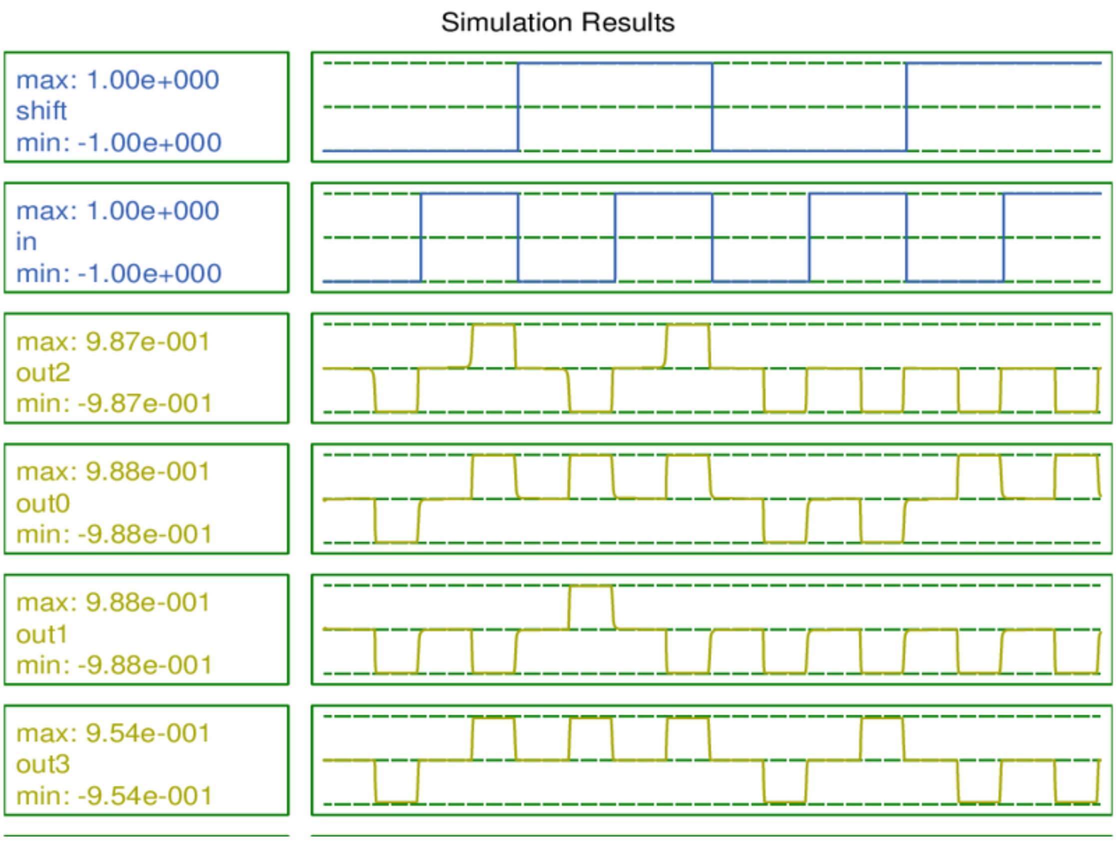


Figure 14. Shift register simulation results.

Conclusion:

This paper addresses the lack of universal design methodologies and architectures for Quantum-dot Cellular Automata (QCA) technology, which has hindered the development of software design tools for large QCA circuits. The authors introduce an automated design methodology using a generic programmable QCA cell crossbar architecture to implement any Boolean logic function. They demonstrate the methodology's effectiveness by designing various Boolean logic circuits, validated using the QCADesigner tool. The methodology is also extended to sequential circuits.

Also, the QCA-LG tool creates layouts for combinational circuits in Quantum-dot Cellular Automata (QCA). It accepts standard netlist formats like LSI and Gate, producing layouts compatible with the QCA Designer tool for physical simulation. While the tool can automatically generate layouts for small circuits, there is room for improvement in optimizing gate placement, especially for medium and large circuits that are not solely composed of majority gates. Further research is needed to enhance efficiency in gate placement.

References:

[1] C. Lent, P. Tougaw, W. Porod, and G. Bernstein, “Quantum cellular automata,” Nanotechnology, vol. 4, no. 1, pp. 49–57, 1993.

[2] S. Seyedi and N. Navimipour, “An optimized design of full adder based on nanoscale quantum-dot cellular automata,” Optik, vol. 158, pp. 243–256, 2018.

[3] S. Heikalabad, M. Asfestani, and M. Hosseinzadeh, “A full adder structure without cross-wiring in quantum-dot cellular automata with energy dissipation analysis,” J. Supercomputing, vol. 74, no. 5, pp. 1994–2005, 2018.

[4] S. Babaie, A. Sadoghifar, and A. Bahar, “Design of an efficient multilayer arithmetic logic unit in quantum-dot cellular automata (QCA),” IEEE Trans. Circuits Syst. II: Exp. Briefs, vol. 66, no. 6, pp. 963–967, Jun. 2019.

[5] N. Safoev and J. Jeon, “A novel controllable inverter and adder/subtractor in quantum-dot cellular automata using cell interaction based XOR gate,” Microelectronic Eng., vol. 222, 2020, Art. no. 111197.

[6] Y. Adelnia and A. Rezai, “A novel adder circuit design in quantumdot cellular automata technology,” Int. J. Theor. Phys., vol. 58, no. 1, pp. 184–200, 2019.

[7] R. Wolkow et al., “Silicon atomic quantum dots enable beyond-CMOS electronics,” in Field-Coupled Nanocomputing, (Lecture Notes in Computer Science Series), vol. 8280, Berlin, Germany: Springer, 2014, pp. 33–58.

[8] R. Achal et al., “Lithography for robust and editable atomic-scale silicon devices and memories,” Nature Commun., vol. 9, no. 1, pp. 1–8, 2018.

[9] T. Huff et al., “Atomic white-out: Enabling atomic circuitry through mechanically induced bonding of single hydrogen atoms to a silicon surface,” ACS Nano, vol. 11, no. 9, pp. 8636–8642, 2017.

[10] C. Lent, “Bypassing the transistor paradigm,” Science, vol. 288, no. 5471, pp. 1597–1599, 2000.

[11] E. Blair, S. Corcelli, and C. Lent, “Electric-field-driven electrontransfer in mixed-valence molecules,” J. Chem. Phys., vol. 145, no. 1, 2016, Art. no. 014307.

[12] V. Kalogeiton, D. Papadopoulos, O. Liolis, V. Mardiris, G. Sirakoulis, and I. Karafyllidis, “Programmable crossbar quantum-dot cellular automata circuits,” IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol. 36, no. 8, pp. 1367–1380, Aug. 2017.

[13] J. Heath, P. Kuekes, G. Snider, and R. Williams, “A defecttolerant computer architecture: Opportunities for nanotechnology,” Science, vol. 280, pp. 1716–1721, 1998.

[14] G. Snider, P. Kuekes, and R. S. Williams, “CMOS-like logic in defective, nanoscale crossbars,” Nanotechnology, vol. 15, no. 8, pp. 881–891, 2004.

[15] Y. Chen et al., “Nanoscale molecular-switch crossbar circuits,” Nanotechnology, vol. 14, no. 4, pp. 462–468, 2003.

[16] A. O. Orlov, I. Amlani, R. K. Kummamuru, R. Ramasubramaniam, G. Toth, C. S. Lent, G. H. Bernstein, G. L. Snider, W. Porod, and J. L. Merz, “Quantum-dot cellular automata: Introduction and experimental overview,” IEEE NANO, pp. 465–470, October 2001.

[17] A. Imre, L. Ji, G. Csaba, A. Orlov, G. H. Bernstein, and W. Porod, “Magnetic logic devices based on field-coupled nanomagnets,” IEEE Semiconductor Device Research Symposium, pp. 25–25, December 2005.

[18] Y. Lu, M. Liu, and C. Lent, “Molecular electronicsfrom structure to circuit dynamics,” IEEE NANO, vol. 1, pp. 62–65, June 2006.

[19] K. Walus, V. Dimitrov, G. Jullien, and W. Miller, “QCADesigner: A CAD tool for an emerging nano-technology,” Micronet Annual Workshop, 2003, web: http://www.qcadesigner.ca.

[20] S. C. Henderson, E. W. Johnson, J. R. Janulis, and P. D. Tougaw, “Incorporating standard CMOS design process methodologies into the QCA logic design process,” IEEE Transactions on nanotechnology, vol. 3, no. 1, pp. 2–9, March 2004.

[21] S. Inc., “Synopsys tools,” May 2003, version 2003.06 for linux.

[22] D. Chai, J.-H. Jiang, Y. Jiang, Y. Li, A. Mishchenko, and R. Brayton, MVSIS 2.0 Users Manual, Department of Electrical Engineering and Computer Sciences.

[23] R. Zhang, P. Gupta, and N. K. Jha, “Synthesis of majority and minority networks and its applications to QCA, TPL and SET based nanotechnologies,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, pp. 229–234, 2005.

[24] Z. Huo, Q. Zhang, S. Haruehanroengra, and W. Wang, “Logic optimization for majority gate-based nanoelectronic circuits,” Circuits and Systems, p. 4 pp., May 2006.

[25] C. S. Lent and P. D. Tougaw, “A device architecture for computing with quantum dots,” in Proceedings of the IEEE, vol. 84, no. 4, April 1997, pp. 541–557.

[26] K. Walus and G. A. Julien, “Design tools for an emerging SoC technology: Quantum-dot cellular automata,” in Proceedings of the IEEE, vol. 96, no. 6, June 2006, pp. 1225–1244.

[27] A. O. Orlov, I. Amlani, R. K. Kummamuru, R. Ramasubramaniam, G. Toth, C. S. Lent, G. H. Bernstein, and G. L. Snider, “Experimental demonstration of clocked single-electron switching in quantum-dot cellular automata,” Applied Physics Letters, vol. 77, no. 2, pp. 295–297, July 2000.