International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.1 (2025) pp. 1117-1127
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

Implementation Of An Ultra-Low-Power High-Speed CLB Using A Memory Element And Its Power Dissipation Analysis For FPGA with Quantum-dot Cellular Automata

Anjaiah Talamala[1]*, Ajit Kumar Patro[2], Kishore Pinninti[3]

[1]GIET University (Department of ECE), Gunupur, Odisha, India.
***Corresponding Author Email:** anjibabu317@gmail.com- **ORCID:** 0009-0009-3618-620X

[2]GIET University (Department of ECE), Gunupur, Odisha, India.
Email: ajitpatro@giet.edu- **ORCID:** 0000-0002-2007-782X

[3]VNRVJIET (Department of ECE), Hyderabad, India.
Email: kishorepinninti14@gmail.com- **ORCID:**0000-0002-2007-7828

Article Info:

DOI: 10.22399/ijcesen.1066
Received : 21 December 2024
Accepted : 17 February 2025

Keywords :

QCA,
DFF,
CLB,
LUT,
MUX,
FPGA.

Abstract:

Quantum-dot Cellular Automata (QCA) was popular due to its very low power consumption and fast operating speed. QCA Designers are fascinated with developing and implementing nanoscale devices of digital circuits with rapid operating speed, low power consumption, and less area requirement. This paper describes an efficient layout design of CLB for FPGA implemented using Lookup Tables (LUTs) constructed by using efficient Multiplexers and DFF. A 4-input LUT is implemented using 2x1 multiplexers, and D Flip Flop is incorporated into the LUT to frame CLB. Programmable routing resources for signal propagation and control are done by quantum cells. The proposed D Flip Flop was implemented with an area-efficient and less delay requirement. The functionality of the D Flip Flop in the CLB was analyzed using the simulation results of the QCA Designer tool. It is implemented with 53 quantum cells in a small area of 0.07µm2 and a delay of one clock cycle. Furthermore, the flexible logic block developed using the proposed D Flip Flop structure has proven to be the best among existing modules. The CLB architecture has 655 quantum cells spread across 1.32 µm2 with three clock cycles delay and a power dissipation of 43. 33nW. It is clear that the present is a better and more effective solution in terms of area, complexity, cell count, and latency. The QCA Pro and QCA Designer software is used for all designs and simulation results.

1. Introduction

Over the past few decades, semiconductor manufacturing technology has advanced quickly, and certain applications need for more speed and less power. These have prompted academics to look for a workaround or substitute technology. Among these potential technologies, QCA presents a predominant. Lent, in 1993 [1], proposed the QCA approach. The quantum cell is the fundamental element of QCA. A cell has four quantum dots in which two free electrons are presented these dots are located at the corners of the structure. For any digital circuit figurations, QCA structures are created and implemented. Rather than using the conventional charge current for digital information transmission, the QCA system uses a polarized approach [2-5].

1.1 Field-Programmable Gate Array

A Field-Programmable Gate Array (FPGA) is an integrated circuit that can be programmed or reconfigured by the user after manufacturing to perform specific logic functions. Unlike traditional processors that execute software instructions, FPGA is configured using hardware description languages (HDLs) like VHDL or Verilog, which describe the circuit's behaviour. One of the key advantages of FPGAs is their parallel processing capability, allowing multiple operations to be performed simultaneously, which leads to faster processing speed for specific tasks compared to

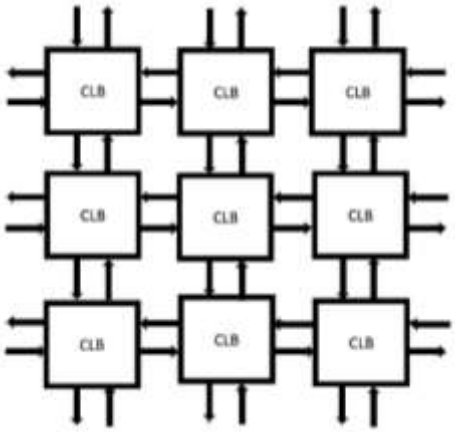


Figure 1. *FPGA architecture*

traditional sequential processors. FPGAs offer low latency and can be optimized for power efficiency, making them suitable for real-time applications. FPGAs comprise a couple of Configurable Logic Blocks (CLBs) arranged in the matrix form accompanied by programmable interconnects. These CLBs can be configured to perform a wide range of logic operations, while the interconnect allows them to be connected in various ways to implement complex digital circuits. Figure 1 is FPGA architecture.

1.2 Configurable Logic Blocks

CLBs are essential components in FPGA, serving as basic building blocks that enable the implementation of complex digital circuits. Each CLB typically contains LUTs, flip flops, and multiplexers, which work together to perform various logic functions. The LUTs handle combinational logic by storing output values for different input combinations, allowing quick retrieval and execution of logic functions. Flip flops are used for storing and transferring data, facilitating the implementation of sequential logic, while multiplexers manage the selection and routing of data within and between CLBs. These elements are interconnected through programmable pathways, giving FPGAs their flexibility and allowing for the creation of intricate digital designs. The configuration and arrangement of CLBs within the FPGA directly impact its capacity to handle complex tasks, so they play a critical role in deciding the device's overall functionality and performance.

1.3 Lookup Table

The heart of the CLB is lookup tables. It is a small memory block that stores the truth tables of logic functions to be implemented, which is constructed with the help of multiplexers (MUXs). LUT is a

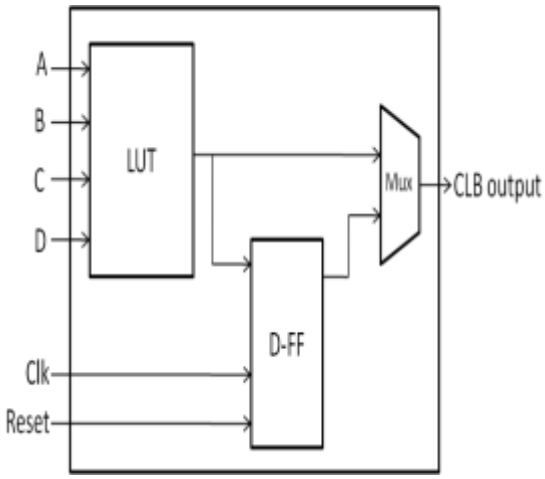


Figure 2. *Configurable Logic Block*

fundamental component in digital logic design, particularly in the FPGAs, Figure 2, where it performs the functions of a programmable logic element that will carry out any boolean operations. The LUT is essentially a memory block that stores the precomputed outputs corresponding to all possible combinations of its input values. When a specific input combination is applied, the LUT retrieves and outputs the corresponding stored value, effectively implementing the desired logic function by configuring the QCA cells, it can implement the programmable logic functions within a compact and efficient nanoscale structure, significantly reducing the power consumption and increasing computational speed compared to traditional CMOS technology. The flexibility of QCA in constructing various logic gates, such as majority gates and inverters, allows for the implementation of complex logic functions, within a QCA-based LUT, making it a promising approach for future nano-electronic devices.

1.4 Multiplexer

A multiplexer, often called MUX, is a digital circuit that selects one input of several and forwards the chosen input to the output. Generally, MUX has 2^N number of inputs, 'N' number of selection lines, and a single output. It operates based on a set of control or selection inputs, determining which input line is connected to the output. A 2x1 multiplexer has two data inputs, one control input, and one output. The control input dictates which data inputs are transmitted to the output. Multiplexers are essential in digital circuits because they enable the efficient management and routing of data within the system, allowing multiple signals to share a single communication line or resource. This functionality

is widely used in data selection processes. Multiplexers proposed the LUTs in CLB, and the function of MUX is to select the direct output or delayed version of it. Generally, 2x1 MUX has been used for LUT. Here, the *2x1* MUX is implemented by 7 QCA cells only.

1.5 Flip-Flop

A Flip Flop in the FPGA is the basic storage element that captures and holds the data input value on a specific clock edge, typically the rising or falling edge. In FPGA, DFFs store binary data, synchronize signals, and create sequential logic circuits. They are critical components in the design of counters, shift registers, and state machines within the FPGA's configurable logic blocks. The paper proposed the implementation of CLB for FPGA, which consists of D Flip Flop. This CLB is implemented using 4-input LUT along with the flip flop. It enabled the implementation of sequential logic circuits like registers, counters, and more. The present designs were worthy in size, quantity of cells, and latency. An energy analysis will be performed on each design in this article.

2. QCA Fundamentals

A majority gate and inverter gates are the fundamental elements of QCA, and are constructed with the aid of QCA cells. Figure 3 depicts a QCA cell with two electrons and four quantum dots on each corner. Due to their electrostatic repletion, these two electrons have a tendency to occupy their corners or diagonally.

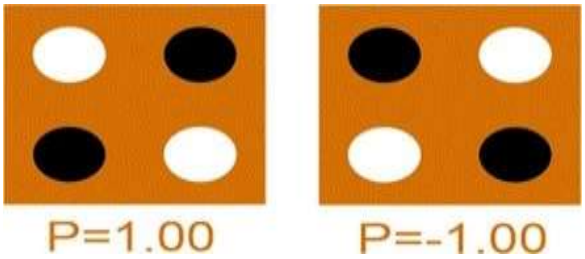


Figure 3. *QCA cell with polarizations*

The binary conditions of a QCA cell are determined by its functional element. If the position of two free electrons is present in an anticlockwise direction, the positional value or the polarization 'P' = ''−1'' is considered as binary "0." If the position of free electrons in quantum dots is in a clockwise direction [6], the value of 'P' = "+1'' is considered binary "1". Furthermore, depending on the polarity of the majority gate inputs, the OR gate and AND gates are developed. If one of the inputs has the fixed polarity '-1' applied to the majority gate, an AND gate is formed, and the fixed polarity of '+1'is applied to one input of the majority gate resembles an OR gate (Figure 4) [6].

2.1 QCA Majority Gate

The majority gate is the fundamental functional component used in QCA to design any digital circuit. It presents three inputs, one output, and one device cell. The device cell is located at the center, and the rest of the four cells are presented in all four directions of the device cell, and the gate output depends on the majority of the inputs.

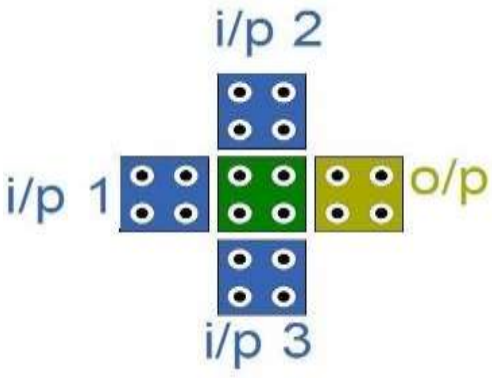


Figure 4. *QCA Majority Gate*

2.2 Inverter Gate

Another essential element in the QCA is the inverter gate. Figure 5 depicts an inverter. There are two ways to construct: keeping the cells arranged diagonally (Figure 5(a)), and constructing the cells at half of the previous cell's location (Figure 5(b). In either case, the output is always the present complement of the input.

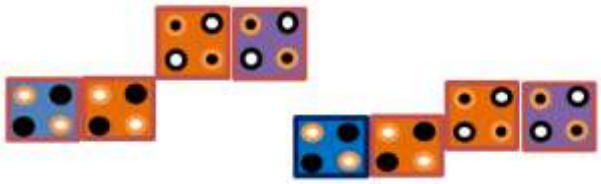


Figure 5. *QCA Inverter Gate (a) Corner cells (b) Half-cell portion*

2.3 QCA Wire

In CMOS circuits wiring is used to provide interconnections and global routing. An aluminium metal layer is preferred to provide connecting paths. In QCA, cells are sequentially arranged to frame wire, which enables efficient and rapid large-volume data transfer. The signal, in this case, travels from input to output due to the cell's electrostatic interaction.

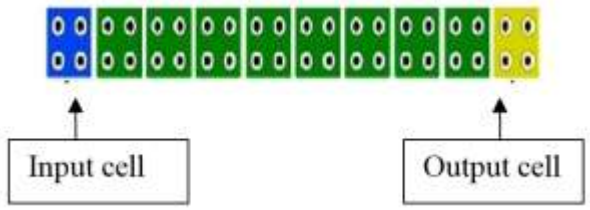


Figure 6. *QCA Wire*

2.4 QCA Clocking

The flow of data within the QCA circuits is governed by a clock system that regulates the power required for circuit operation. QCA employs a 4-zone clock scheme to manage cell interactions and intracell information, the design process becomes more systematic and forward-looking. Figure 6 is QCA wire and the four distinct states of the QCA clocks are displayed in Figure 7 and the clocking techniques are as follows: clock0, clock1, clock2, and clock3 [7-19]. The main difference between these four clocks is their internal pacing and data transfer conditions, no two clocks are ever in the same state. The quantum lagging is the propagation delay of the circuit. The clocking scheme of the clock cycles is a switch, hold, release, and relax, and the cycle time zones repeat across these four phases.

Switch Phase: During this phase, electrons aim to transition from their previous state to a new state, initiating the change in information within the circuit. **Hold Phase**: Here, electrons stabilize into a steady state, allowing for the output to be accurately obtained and processed. **Release Phase**:

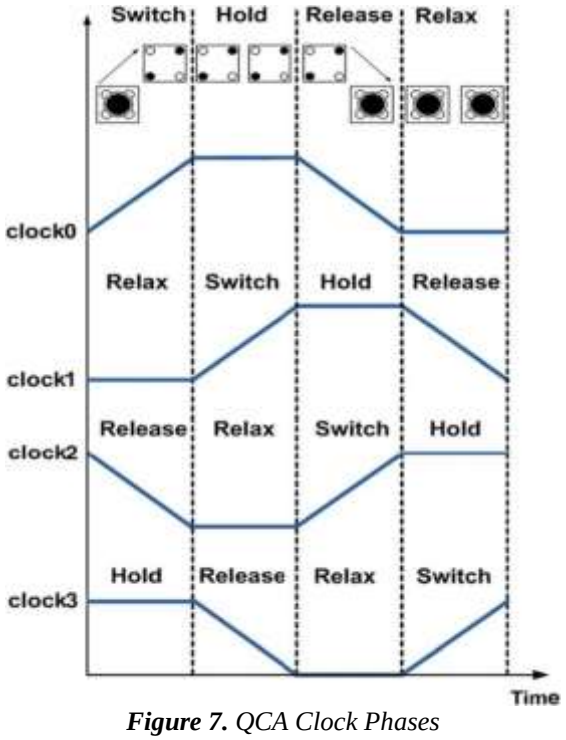


Figure 7. *QCA Clock Phases*

In this phase, electrons begin to move toward the next cell, preparing for the transfer of information. **Relax Phase**: Finally, electrons are poised to receive a new state, marking the completion of the clock cycle and setting the stage for the next iteration of the information flow. The contents of this article were categorized into five sections and are arranged as follows. In the first section, the structure and configuration of quantum cells, fundamental gates, QCA wiring, and the clock analysis are explained. The second section describes the previous designs and their limitations. Section three explains the proposed D Flip Flop schematic, layout implementation, simulation results, and comparisons of various designs. It also explains the multiplexer layout and its simulation results. LUT implementation, D Flip Flop incorporates the LUT and CLB implementation, and comparisons of various CLBs for FPGA implementation are discussed in the fourth section. The fifth section presents the power dissipation analysis of the proposed designs and the conclusion.

3. Literature survey

Craig S. Lent et al. [1] In 1993, they discovered that QCA nanotechnology can be used to represent logic gates. Later, in 1994, the same authors used this QCA to implement majority and inverter gates. The optimized logic gates have been implemented using QCA in [6]. Zoka, S. Gholami, M et al. [2,9] present the proposal for two new QCA-optimized flip flops with reset input. Two clock cycles are needed for the first suggested level-triggered D Flip Flop circuit with 82 quantum cells to operate correctly. Moreover, the second proposed architecture, D Flip Flop, consists of 85 cells and one clock cycle delay. Binaei R. and Gholami M. Introduced multiplexers based edge sensitive D Flip Flop design using QCA [3]. It works with 93 cells and an area of 0.2 um^2 region. Angizi, S., Moaiyeri, M. H., et al [5]. Implemented a T flip flop using QCA. In 2012, S. Hashemi et al. developed resilient DFF [7] for sequential circuits. The DFF contains 84 quantum cells a little higher, 0.09 μm2, with a delay of 2.75 clock cycles. This D Flip Flop has taken a considerable number of clock cycle delays. A 5-input majority gate was used by Zoka et al. (2018) [9] to introduce a rising edge-triggered resettable D Flip Flop. This DFF has a delay of one clock cycle and 95 quantum cells higher spread across 0.11 μm^2. Binaei et al. [11] employed QCA technology to present a DFF with set and reset pins. This architecture requires 73 quantum cells within a 0.1 μm^2 area with a 2.5 clock cycle latency. In this design, the layout has slightly higher quantum cells,

and the delay is significant compared to the earlier designs. Ghosh, in 2020, Raj Marshal et al. [13] Proposed a Configurable logic memory block beyond CMOS technology. The suggested memory block was constructed with D Flip Flop, which comprises 58 quantum cells in an area of 0.07μm^2 with a delay of 1.25 clock cycles. Bahniman, et al. [15] proposed multi-layered QCA for CLB unit to implement FPGA with 3915 quantum cells and an area of 2.42 μm² with a delay of 36 picoseconds. Zhang, Yongqiang, et al. [16] Employing Clocking Mechanism in Quantum-Dot Cellular Automata to Implement SRAM for LUT and CLB, the built CLB featured 6400 quantum cells, 10.67 μm^2 in area, and a 45 picoseconds delay. We can conclude that DFF is crucial to many digital logic circuits, such as Configurable Logic Blocks. It must be implemented with fewer quantum cells, less area, and lower power consumption. In this paper, we will propose a low power, low area, less delay, and low design complexity of DFF that will help the implementation of configurable logic blocks.

4. Implementation, result analysis of DFF and mux

4.1 D Flip Flop

The digital circuit that holds a single bit of data is called a flip flop, also known as a storage element. DFF [4,18] is used to store the applied data within itself. It has one data input (D) and a clock input (CLK). When the clock signal goes from low to high, the value of data input is sent to the output (Q), and if the clock changes from high to low, transition input has no effect on the output. Figure 8(a) shows a DFF constructed using NAND gates and an inverter gate.Sequential circuits like flip flops are crucial to the field of QCA because large-scale sequential circuits like shift registers and counters are anticipated to be designed and implemented using them. The characteristic equation of the DFF is obtained by taking past output Q_n and present input D [12]. The table 1 will give the status of output Q_{n+1} depending on the present input and past outputs. The characteristic equation can be written based on the characteristic table shown in Table: 2.

$$Q_{n+1} = D \qquad (1)$$

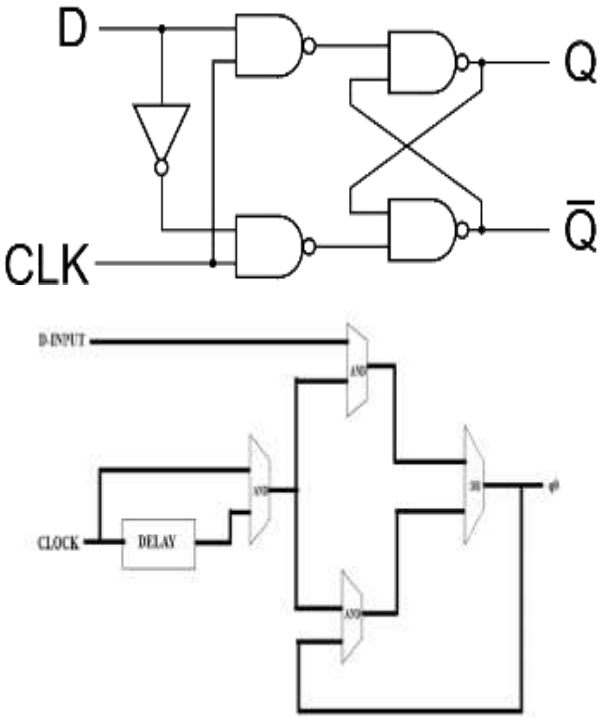


Figure 8. *(a) D Flip Flop using Nand gates (b) D Flip Flop Schematic using Majority Gates*

Table 1. *Truth table for D-FF*

CLK	D	Q	$\hat{Q}$
0	0	Q	$\hat{Q}$
0	1	Q	$\hat{Q}$
1	0	0	1
1	1	1	0

Table 2. *Characteristic table for D-FF*

Qn	D	Qn+1
0	0	0
0	1	1
1	0	0
1	1	1

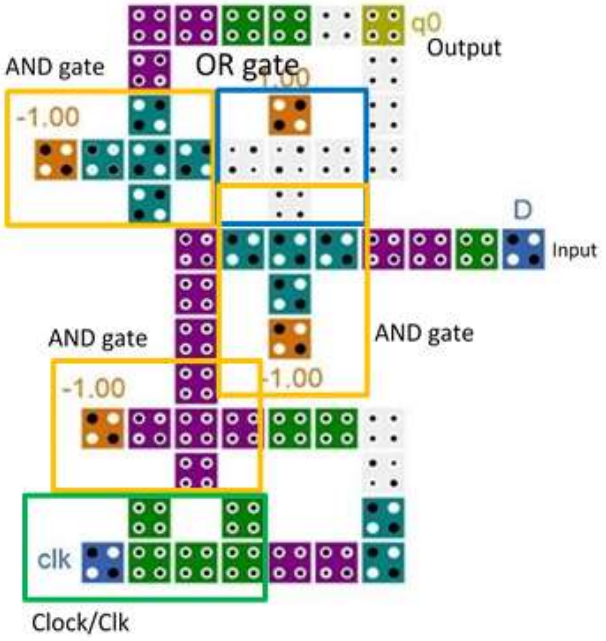


Figure 9. *D Flip Flop QCA Layout*

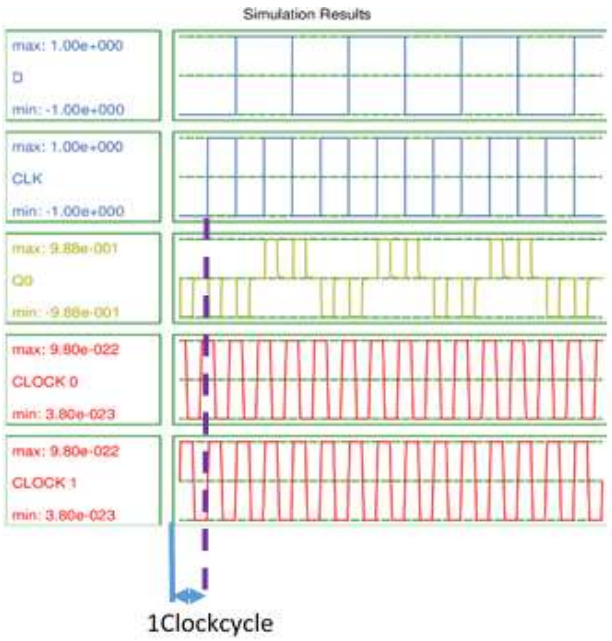
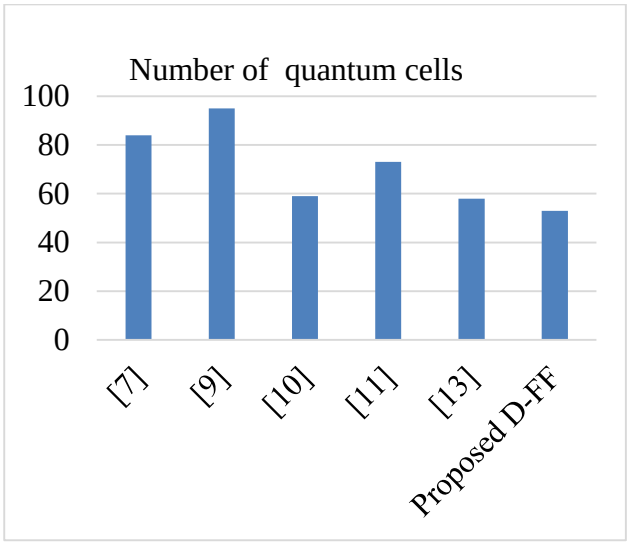


Figure 10. *Simulation results of D FLIP FLOP*

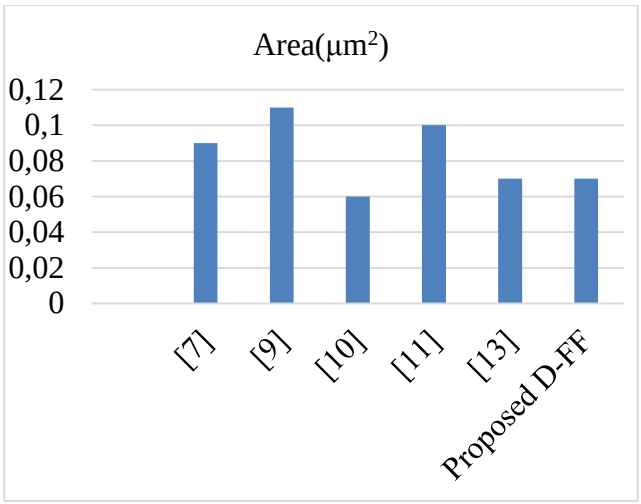
Figure 8(b) is the schematic diagram of DFF using QCA majority gates, and Figure 9 depicts the QCA layout for DFF, and it is constructed using majority gates like AND gates and OR gate [6]. The orange colour represents AND gates, and the blue colour represents an OR gate. The data input and the clock applied to the circuit, output is considered across q0. This D Flip Flop layout is proposed with 53 QCA cells and a delay of 1 clock cycle. It is optimized in terms of cell count, delay, and area and also has low complexity due to single layer. It is recommended that CLB be implemented for FPGA. The simulation results depict the effective implementation of DFF and its working performance. Table: 3 shows comparisons of various DFF, the delay is more in [7] and [11] later reduced in [10,13]. However the number of QCA cells are also reduced. In the design [9] even the required delay is 1 clock cycle and the occupied area is more 0.11μm^2. From Figure 10 when clock signal has its rising edge as shown, the input data is transferred to its output. If the transition of the clock change from high to low the same data is transferred until it can change from low to high.

Table 3. *Comparison of various design parameters of DFF*

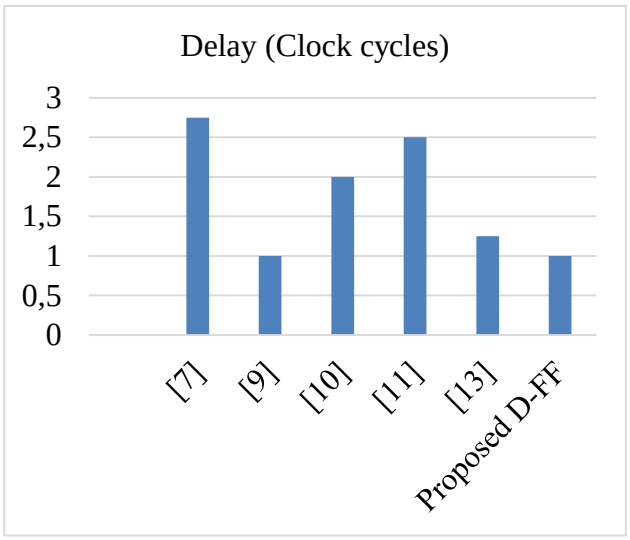
Design	**Number of Quantum cells**	**Area(μm^2)**	**Delay (Clock Cycles)**
[7]	84	0.09	2.75
[9]	95	0.11	1
[10]	59	0.06	2
[11]	73	0.1	2.5
[13]	58	0.07	1.25
Proposed DFF	**53**	**0.07**	**1**



(a)



(b)



(c)

Figure 11. *Graphical representation of (a) Number of cells, (b) Area (μm^2) (c) Delay (Clock cycles)*

The layout required four clock phases to obtain exact output. The proposed design uses 53 QCA cells with an area of 0.07μm^2 and the latency of 1 clock cycle.Figure 11 depicts a graphical representation of the proposed design performance metrics. It has low quantum cells, area, and delay compared to existing models.

4.2 Multiplexer

A QCA multiplexer is a nanoscale device that selects one channel input to the output from several input signals based on control inputs. The multiplexer can be used to implement a lookup table. In the CLB, the multiplexer's function is to select the direct output from the LUT or its delayed version. For this, a *2x1* multiplexer has been typically employed. Figure 12 shows a *2x1* MUX is implemented using QCA with seven cells, which occupy a 0.01μm^2 area. The Figure 13 explains the performance of the designed MUX. If the selected input S_0 is '0', the content of data I_0 is sent to the output. When S_0 is '1', it selects the output $Y = I_1$. The designed mux has no delay, and the LUT of this multiplexer works effectively.

5. Implementation of lut and clb

5.1 Implementation of 4-input Lookup Table

Figure 14(a) shows the schematic of a 4-input LUT constructed using 2x1 multiplexers, and the inputs are A, B, C, and D, with selection lines at different stages. The primary function of the SRAM Cell is to provide data to the inputs of a 2x1 multiplexer. The data in SRAM is set based on the function that the CLB needs to complete. In the layout Figure 14(b), the blue colour quantum cells represent the SRAM cell that holds the input data to be selected by the multiplexer. The 4-input LUT in digital

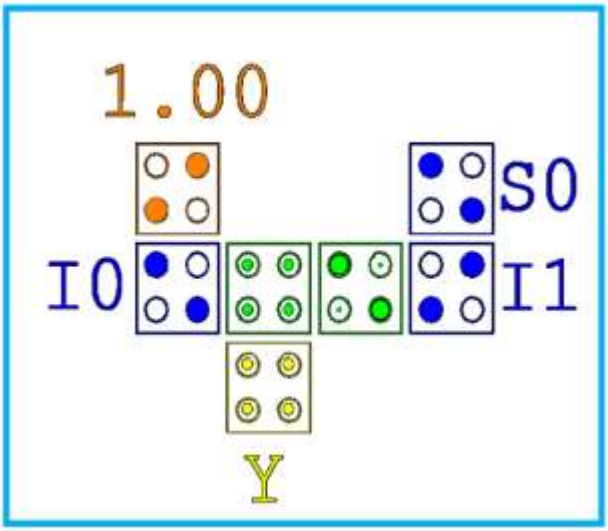


***Figure 12**. 2x1 Multiplexer layout using QCA*

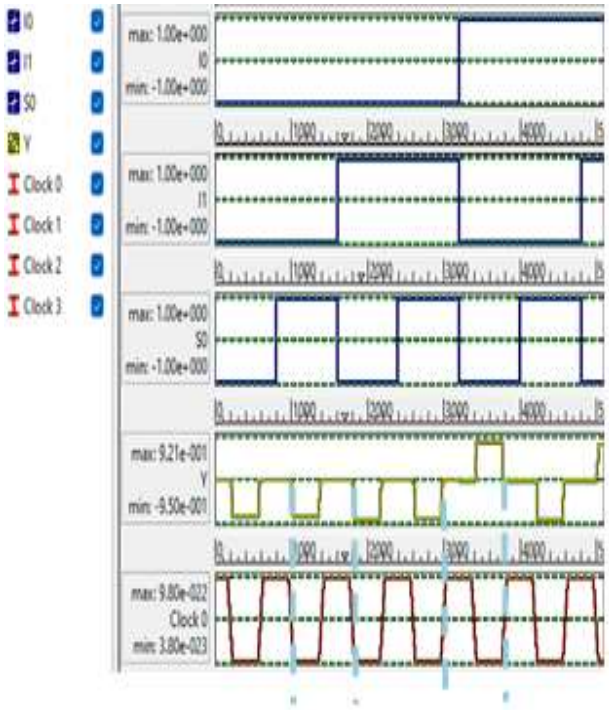


***Figure 13.** Simulation results of Multiplexer*

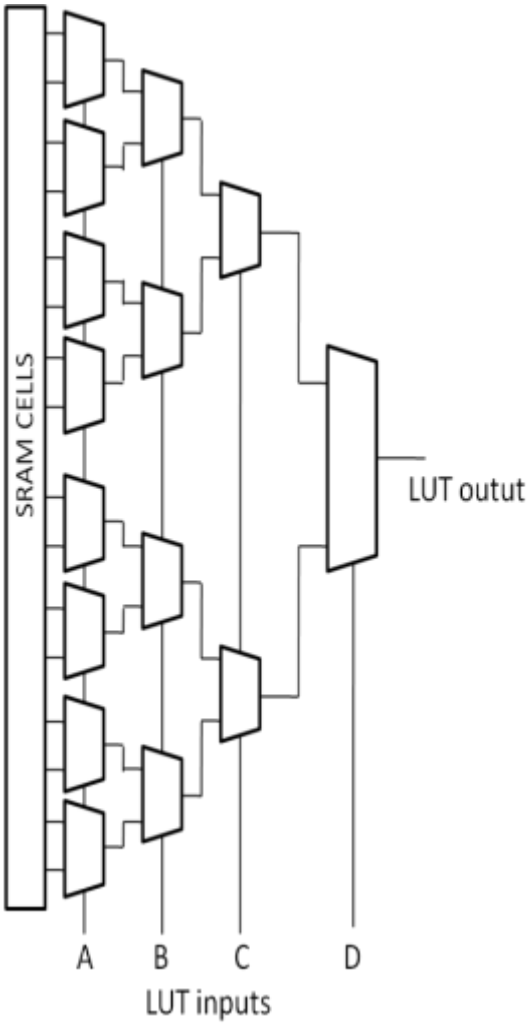


***Figure 14.** (a) Schematic of 4 Input LUT*

logic is a highly flexible component that can implement any Boolean function involving four inputs. The proposed QCA layout of LUT is implemented with 547 quantum cells and an occupied area of 0.88 μm^2 with a propagation delay of 2.5 clock cycles. Each clock cycle will offer four picoseconds. Therefore, the overall delay associated with the proposed LUT is 2.5 x $4x10^{-12}$ sec = 10^{-12} sec is 10 pico sec.

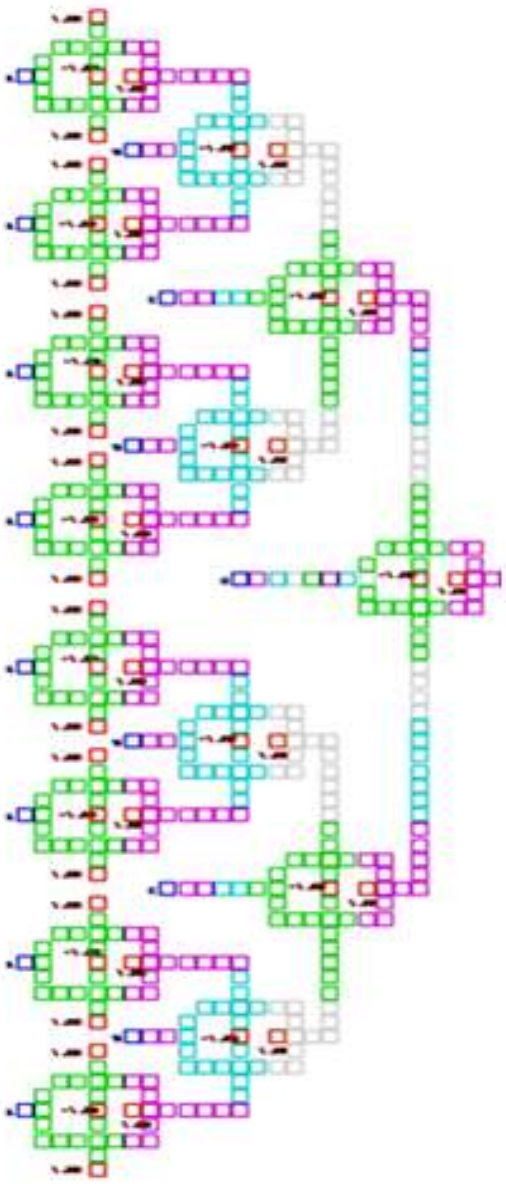


Figure 14. *(b) QCA layout for 4 Input LUT*

Configurable Logic Block

CLBs are the essential components in the FPGA, which is the heart of FPGA. Each CLB typically contains LUTs, D Flip Flop, and multiplexers, which work together to perform various logic functions. Figure 15 shows the detailed construction of CLB implemented with the help of 4-input LUT shown in a yellow coloured box, the DFF in green, and the 2x1 multiplexer is represented with a dark blue coloured box. The layout is implemented with 655 quantum cells, an optimized area of 1.32 μm^2, and the delay occurred in transferring the input to the output in three clock cycles. Each cycle has four clock phases, and each phase will offer $12x10^{-12}$sec. Table: 4 comprises three metric performances of the proposed and the previous designs [8,15,16] in area, delay, and cell count. Figure 16 depicts the graphical representation of the designed metrics visualized in quantum cells area and delay. The result says that the proposed model is the best among the existing models.

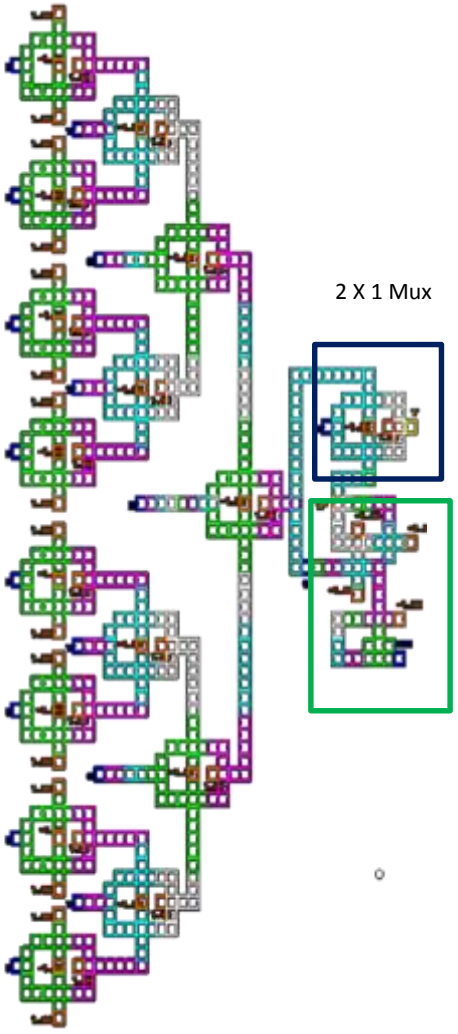
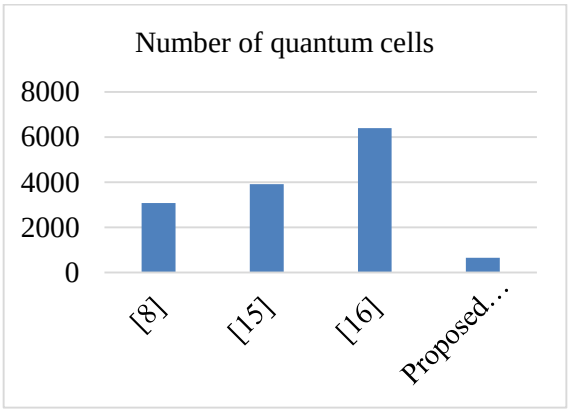


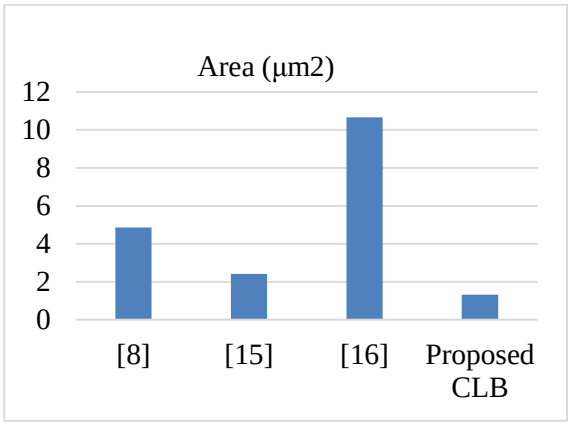
Figure 15*. QCA layout for CLB*

Table 4. *Comparison of various performance metrics for CLBs*

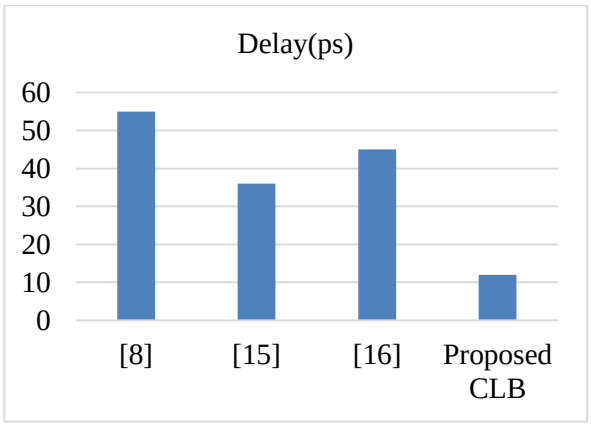
Design	**Number of quantum cells**	**Area (μm^2)**	**Delay(ps)**
[8]	3075	4.86	55
[15]	3915	2.42	36
[16]	6400	10.67	45
Proposed CLB	**655**	**1.32**	**12**



(a)



(b)



(c)

Figure 16. *Graphical representation of performance metrics of CLB (a) Number of quantum cells (b) Area in μm^2 (c) Delay in picoseconds*

6. Power dissipation analysis of proposed CLB

There are several tools available for calculating power dissipation, including QCAPro and QCADesigner E 2.2. [17,18]. The current research used the QCADesignerE 2.2 tool to determine the energy dissipation of the CLB in FPGA which was studied [20,21]. The proposed QCA layout is simulated using the Coherence vector simulation model. Coherent vector simulation using the Euler technique has been used in the power dissipation analysis of the disclosed model [22], as shown in Figure 17. It evaluates the energy dissipation of each quantum cell uniformly in the total environment of the layout [16]. The energy dissipation of a layout is considered in two factors: one is the energy dissipation of all the quantum cells in the layout environment, and the second is the average energy dissipation of each quantum cell in the entire layout environment. The energy dissipation of the proposed CLB is given as Sum_bath of 3.24e-001 eV. This value will help to determine the power consumption or dissipation [19] of the proposed layout. Figure 17 also shows the various parameters like temperature, relaxation time, clock period, input period, time step, total simulation time, clock high, low, shift, and slope, radius of effect, relative permittivity, and layer separation, which are associated with the simulation of the QCA layout design and its simulation. As shown in Figure 17, temperature says the layout is simulated in the mentioned 1^0C. Relaxation time, clock period, clock high and low, clock shift, and clock slope parameters are related to the clock signal. The clock period, by default, is set to T = $4x10^{-12}$ sec, which decides the speed of the layout operation. As this value is meager, the QCA layout works at Tera hertz of frequency (f=1/T=1/($4x10^{-12}$) =$0.25x10^{12}$ Hertz=0.25 THz).

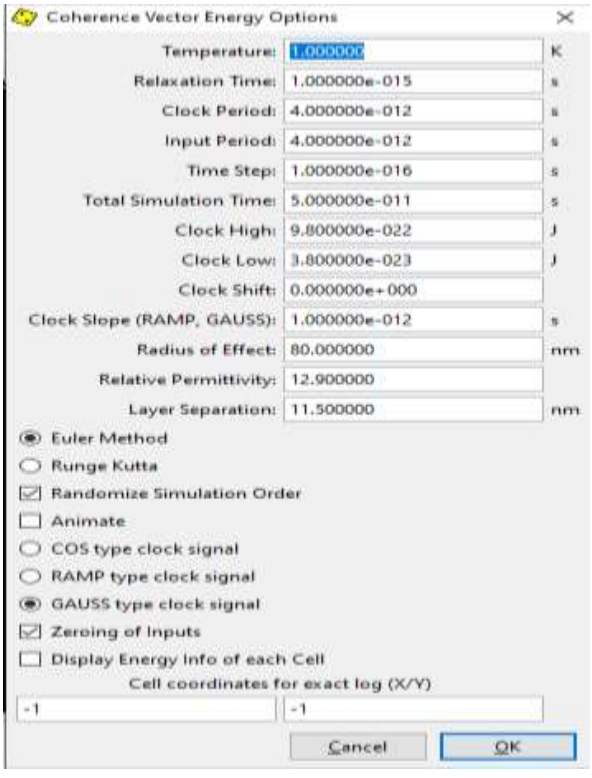


Figure 17. *Specification of each parameter in simulation analysis.*

The proposed layout's energy dissipation is E_d-=3.24eV. The value in electron Volts will be converted into joules by multiplying with the electron's charge.

$$\text{Charge of an electron (eV)} = 1.608\text{X}10^{-19} \text{ Joules}$$

$$E_d = 3.24\text{eV} = 3.24 \text{ X } 1.608\text{X}10^{-19} \text{ Joules}$$

$$E_d = 5.20 \text{ X}10^{-19} \text{ Joules}$$

The delay incurred for the proposed layout is about three clock cycles, and for the selected coherent vector simulation method, the period of each clock cycle is 4 Pico seconds.

Hence, the total time delay can be calculated as follows:

Delay in time, T_d = 3 clock cycles.

The period of each clock in the simulation tool is $4x10^{-12}$ sec.

Therefore, the time delay, $T_d = 3X4X10^{-12}\ sec.$

$$= 12X10^{-12}\ sec.$$

$$= 12\ pico\ Sec$$

The power dissipation will be evaluated using the standard correlation between energy, power, and time.

$$\text{Energy} = \text{Power X Time}$$

$$\text{Power} = \frac{\text{Energy}}{\text{Time}}$$

$$\text{Power dissipation, } P_d = \frac{E_d}{T_d}$$

$$P_d = \frac{5.20\text{X}10^{-19}}{12\text{X}10^{-12}}$$

$$= 0.4333 \text{ X } 10^{-7}$$

$$\text{Therefore, } P_d = 43.33 \text{ X } 10^{-9} \text{ Watts.}$$

Therefore, the power dissipation of the presented CLB is 43.33 nW.

7. Conclusion

This paper presents the development of a low-power and area-efficient D Flip Flop layout employing the fastest-growing and most advanced nanotechnology Quantum-dot Cellular Automata. The suggested DFF is the best model currently available in terms of area, delay, and quantum cell count. The performance metrics of the suggested D Flip Flop and its layout were integrated into the CLB, which is the heart of FPGA. The CLB with the proposed D Flip Flop layout took 655 cells in a 1.32 μm² area with a delay of 3 full clock cycles. The number of quantum cells, area, power dissipation, and latency of the proposed CLB architecture are also examined. The energy and power dissipations have been evaluated using the QCA Designer-E tool. The energy and power consumed by the proposed configurable logic blocks are 3.24 eV and 43.33 nW, respectively. It is concluded that the proposed D Flip Flop is the best model among the existing, and performs well when integrated into a complex architecture like CLB. Hence, the FPGA architecture can be implemented using the proposed configurable logic block's layout.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1]Lent, C. S., Tougaw, P. D., Porod, W., & Bernstein, G. H. (1993). Quantum cellular automata. *Nanotechnology*, *4*(1), 49.

[2]Zokaa, S., & Gholami, M. (2018). Two Novel Flip Flop with Level Triggered Reset in Quantum Dot Cellular Automata. *IJE TRANSACTIONS C:* Aspects 31(53);415-421

[3]Binaei, R., & Gholami, M. (2019). Design of multiplexer-based D flip-flop with set and reset ability in quantum dot cellular automata nanotechnology. *International Journal of Theoretical Physics*, *58*(3), 687-699.

[4]Vetteth, A., Walus, K., Dimitrov, V. S., & Jullien, G. A. (2003). Quantum-dot cellular automata of flip-flops. *ATIPS Laboratory*, *2500*, 1-5.

[5]Angizi, S., Moaiyeri, M. H., Farrokhi, S., Navi, K., & Bagherzadeh, N. (2015). Designing quantum-dot cellular automata counters with energy consumption analysis. *Microprocessors and Microsystems*, *39*(7), 512-520.

[6]Talamala, A., Patro, A. K., & Pinninti, K. (2023, July). Implementation of Optimized Logic Gate Circuits with Majority Voter using QCA Technology. In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-9). IEEE.

[7]Hashemi, S., & Navi, K. (2012). New robust QCA D flip flop and memory structures. *Microelectronics Journal*, *43*(12), 929-940.

[8]Chakrabarty, R., Mahato, D. K., Banerjee, A., Choudhuri, S., Dey, M., & Mandal, N. K. (2018, January). A novel design of flip-flop circuits using quantum dot cellular automata (QCA). In *2018 IEEE 8th Annual computing and communication workshop and conference (CCWC)* (pp. 408-414). IEEE.

[9]Zoka, S., & Gholami, M. (2018). A novel rising edge triggered resettable D flip-flop using five input majority gate. *Microprocessors and Microsystems*, *61*, 327-335

[10]Sasamal, T. N., Singh, A. K., & Ghanekar, U. (2019). Design and implementation of QCA D-flip-flops and RAM cell using majority gates. *Journal of Circuits, Systems and Computers*, *28*(05), 1950079.

[11]Binaei, R., & Gholami, M. (2019). Design of novel D flip-flops with set and reset abilities in quantum-dot cellular automata nanotechnology. *Computers & Electrical Engineering*, *74*, 259-272.

[12]Sasamal, T. N., Singh, A. K., & Ghanekar, U. (2019). Design of QCA-based D Flip flop and memory cell using rotated majority gate. In *Smart Innovations in Communication and Computational Sciences: Proceedings of ICSICCS 2017, Volume 2* (pp. 233-247). Springer Singapore.

[13]Raj, M., Gopalakrishnan, L., Ko, S. B., Naganathan, N., & Ramasubramanian, N. (2020). Configurable logic blocks and memory blocks for beyond-CMOS FPGA-based embedded systems. *IEEE Embedded Systems Letters*, *12*(4), 113-116.

[14]Ajitha, D., Ahmed, S., Ahmad, F., & Rajini, G. K. (2021, March). Design of area efficient shift register and scan flip-flop based on QCA technology. In *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)* (pp. 716-719). IEEE.

[15]Ghosh, B., Chandra, J. S., & Salimath, A. (2014). Design of a multi-layered QCA configurable logic block for FPGAs. *Journal of Circuits, Systems and Computers*, *23*(06), 1450089.

[16]Gudivada, A. A., & Sudha, G. F. (2022). Single-Precision Floating-Point Multiplier Design Using Quantum-Dot Cellular Automata with Power Dissipation Analysis. In *Advances in Signal Processing and Communication Engineering: Select Proceedings of ICASPACE 2021* (pp. 103-116). Singapore: Springer Nature Singapore.

[17]Gudivada, A. A., & Sudha, G. F. (2021). Novel optimized tree-based stack-type architecture for 2 n-bit comparator at nanoscale with energy dissipation analysis. *The Journal of Supercomputing*, *77*, 4659-4680.

[18]Gudivada, A. A., & Sudha, G. F. (2022). STQCA-FFT: A fast fourier transform architecture using stack-type QCA approach with power and delay reduction. *Journal of Computational Science*, *60*, 101594.

[19]Gudivada, A. A., & Sudha, G. F. (2022). Novel optimized low power design of single-precision floating-point adder using Quantum-dot Cellular Automata. *The Journal of Supercomputing*, 1-19.

[20]Rahul SHANDILYA, & R.K. SHARMA. (2024). ProTECT: A Programmable Threat Evaluation and Control Unit for Zero Trust Networks. *International Journal of Computational and Experimental Science and Engineering*, 10(4). https://doi.org/10.22399/ijcesen.673

[21]SHANDILYA, R., & SHARMA, R. (2024). FIS: A Programmable Multi-Bit Fault Injection Server for Embedded System. *International Journal of Computational and Experimental Science and Engineering*, 11(1). https://doi.org/10.22399/ijcesen.674

[22]Bhavani, K. S., & Alinvinisha, V. (2015, March). Utilization of QCA based T flip flop to design counters. In *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)* (pp. 1-6). IEEE.