FACTA UNIVERSITATIS
Series: **Electronics and Energetics** Vol. 38, Nº 1, March 2025, pp. 163 - 186
https://doi.org/10.2298/FUEE2501163M

Original scientific paper

NANOTECHNOLOGY QCA-BASED SUB-COMPONENTS OF PROCESSOR DESIGN AND APPLICATION OF FUTURISTIC LOW-POWER DESIGN

Neeraj Kumar Misra[1], Nirupma Pathak[2], Bandan Kumar Bhoi[3], Seyed-Sajad Ahmadpour[4], Sankit R. Kassa[5], Nima Jafari Navimipour[4,6]

[1]School of Electronics Engineering, VIT-AP University, Amaravati, Andhra Pradesh 522237, India
[2]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Green Fields, Guntur District, Vaddeswaram, Andhra Pradesh, 522502, India
[3]Department of Electronics and Telecommunication,
Veer Surendra Sai University of Technology, Burla, Sambalpur 768018, Odisha, India
[4]Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Kadir Has University, Istanbul, Turkey
[5]Electronics and Telecommunication Engineering, Symbiosis Institute of Technology Pune, Symbiosis International Deemed University, Pune, Maharashtra
[6]Future Technology Research Center,
National Yunlin University of Science and Technology, Douliou, 64002, Taiwan

ORCID iDs:		
	Neeraj Kumar Misra	https://orcid.org/0000-0002-7907-0276
	Nirupma Pathak	https://orcid.org/0000-0003-3441-5987
	Bandan Kumar Bhoi	https://orcid.org/0000-0003-2916-2903
	Seyed-Sajad Ahmadpour	https://orcid.org/0000-0003-2462-8030
	Sankit R Kassa	https://orcid.org/0000-0002-8714-2073
	Nima Jafari Navimipour	https://orcid.org/0000-0003-3259-6841

Abstract. *Many devices consist of low-power processor. Quantum-dot-cellular-automata (QCA) based processor designs provide enhanced performance compared with conventional metal-oxide-semiconductor (MOS) based processors. Nanocomputing-based processors are often energy-efficient. We have developed Nanotechnology QCA-based different sub-components of processor such as 2-to-4 decoder, 3-to-8 decoder, Delay Flip-flop (D-FF), and sequence counter. A potential energy proof has been measured in the 2-to-4 decoder design. The synthesis approach algorithm has been presented for all designs. Further, the potential energy calculation results show for 2-to-4 decoder. According to the synthesis results 2-to-4 decoder has improved 82.3% cell count, 86% area, and 85% latency over previous work. Comparing the primitive results with the prior one, results improved by 64% and 76% in terms of cell count and area in the design of the 3-to-8 decoder. Among the different components of the processor is D-FF, which has an improvement of 66.37% in cell counts and 62.5% in area over the prior design. Primitive results have improved, including latency, cell count, and area, showing the proposed processor design is comparable to low-power devices and high speed. In terms of balance power, the proposed subcomponent of the processor will benefit low power device.*

Received December 7, 2024; revised January 19, 2025 and January 24, 2025; accepted January 26, 2025
Corresponding author: Neeraj Kumar Misra
School of Electronics Engineering, VIT-AP University, Amaravati, Andhra Pradesh 522237, India.
E-mail: neeraj.misra@vitap.ac.in

Key words: *QCA, Computing, Processor, Sequence counter, Decoder, Nanotechnology, Algorithm, low Power*

1. INTRODUCTION

A low-power device has a processor that can calculate logic bits in real time [1]. A processor is usually embedded within most devices, along with Wi-Fi, memory, and peripherals that perform various tasks. In low-power devices, ARM Cortex-M processors are used, which consume less power than traditional ARM processors [2]. The cortex-M series of ARM processors has an inbuilt processor, which is based on transistors. The MOS transistor has a number of limitations, such as hot carrier effects and tunnelling, which affect the performance of the processor during computations [3]. It is a conventional CMOS-based decoder which requires a certain amount of time for active charge or discharge to occur. During the era of QCA field computing based computing, the decoder directly interacts between the quantum cells that creates polarization and the signal propagates to the outputs node with a lower latency. The development of new technologies, such as nanocomputing, therefore resulted in alternative technologies emerging. QCA technology is growing in popularity in this emerging era as a result of high-speed computing, nanotechnology based and energy efficient. It is important to note that CMOS devices provide semiconductor-based computing functions on a micro level in the VLSI domain. The advantage of nanocomputing is, however, that it is an emerging technology, especially when it comes to fast computing and high-performance systems [4]. The layout of CMOS technology is not incredibly useful, due to its limitations such as short channel effect and hot carrier effect [4]. It is also important to note that the compact layout of QCA, combined with the use of nanoscale, reduces the processing time and the interconnection delays. In the field of nanocomputing, QCA is emerging as one of the most promising emerging areas. It was first developed by researchers at the University of Notre Dame in the 1960s [5].

QCA layout design focuses mainly on designing the majority of gates at the bottom level, and then it converts to nanotechnology-based quantum cells for layout [6]. Based on the coulomb interaction (the force of attraction between electronics) between neighbouring cells. The QCA cells are evaluated to determine the influence of polarization on electron behaviour and then the computation is made [7].

In this paper, we propose new subcomponents of a processor using algorithms, such as 2-to-4 decoder, 3-to-8 decoder, D-FFs, and sequence counters, which are robust in layout and fast computing. The existing work in [8-16] shows only a design without considering potential energy physical proof, whereas the current work focuses on a robust design of 2-to-4 decoders with potential energy physical proof by mathematical equations. Similar to the proposed sequence counter, it uses fewer majority gates and without crossover so that it has a lower latency and a higher computing speed. The proposed counter is more energy-efficient because fewer cells are used and there is a smaller area involved. In addition to being more robust in terms of area and computation speed.

There is a key point that can be pointed out regarding the workaround proposed sub-component of the processor which is summarized as follows:

- A different subcomponent of the processor layout is targeted in nanocomputing-based QCA.
- Design 2-to-4 and 3-to-8 decoders have been implemented in QCA Technology.

- The sequential D-FF implementation and verification are presented using QCADesigner.
- A sequence counter layout has been presented in QCA using a coplanar layout.
- First time in literature, potential energy calculation is explicitly proven in the design of a 2-to-4 decoder.
- Using the comparison tables, the layout design of the 2-to-4 decoder, 3-to-8 decoder, D-FF, and sequence counter are determined as superior.
- An energy dissipation study of a 2-to-4 decoder circuit has been carried out successfully.
- The complete algorithm has been presented for the synthesis of the n-to-2n decoder and sequence counter.

The article is organized as follows: Section 2 provides an overview of QCA technology. A brief literature review of related QCA-based existing work is presented in Section 3. In section 4, we present improved QCA-based 2-to-4 decoder, 3-to-8 decoder, D-FF, sequence counter layouts, energy estimation of 2-to-4 decoders, and potential energy calculation for 2-to-4 decoders with various subsections. We compare and evaluate the performance attributes of QCA designs with those of the energy dissipation study in Section 5. In Section 6, we conclude the presented study.

2. PRELIMINARIES

The objective of the section is to present the existing state-of-the-art work in the design of decoders, D-FFs, and sequence counters, as well as their pros and cons. Many designs and architectures have been proposed for efficient decoders circuits based on nanotechnology, which includes decoders [8-16], D-FF [17-19], and counter design [20-25] in the category of QCA circuits based on nanotechnology. Numerous studies have been published in the literature concerning the design of synchronous counters [20-25]. As part of the present existing decoder design used MG and INV with fewer wire cells and fewer clocks in the design [8-11]. Several optimized designs of counters based on area, delay, and cell count are available in the literature [20-25]. The decoder modules in reference [11] were based on coplanar cells and utilized 200 cells with a 0.22 um2 area and 3 latency. A reference [11] has been published that analyzes the energy dissipation of the proposed design using the QCAPro tool. Due to the long wires in the design and the fact that there are more majority gates (MG), there is an increase in latency. The D-FFs and counters in the design of the decoder provide no crossovers and no rotated cells, so a reduction in latency can be achieved, as well as high computation speed. The existing design of the decoder [8] uses the coplanar technique to optimize and reduce the latency to achieve a compact design of the decoder. Due to the crossing of wires, the latency is increased, as well as the number of clocks. Based on the existing work, it uses a cell count of 212, an area of 0.25, and a latency of 4. In the existing work, MV flip flops [17,18,19] are used with rotated cells, INV to compact the size, however, due to the rotation of the cells, the design requires more clocks. In addition to this, the design of D-FF has an INV that is not robust which increases the area of the design as well as the computational delay. The existing work [14] deals with decoding as well as memory cell layout using direct connections and symmetrical connections. Look-up table implementations of decoders are the subject of this work.

Table 1 Review of the existing work

Existing Work	Methodology	Pros and cons	Technology used
[11]	QCA based decoder	The use of long wires increases latency when there are more clocks. The designer has used a coplanar to make fabrication easier.	Nanotechnology
[12]	QCA based decoders	A modular decoder is being studied in existing work using rotated cell types with a crossing approach.	Nanotechnology
[14]	QCA-based decoder, and memory element based on look-up table	This existing work was to develop a decoder that uses direct connection and symmetrical connection modes. A further investigation demonstrates the use of a look-up table to generate a complete layout using a decoder, memory, and OR tree.	Nanotechnology
[16]	QCA based decoders	It deals with the decoder types 2-to-4 and 3-to-8 using MG because MG and INV are more common in the design of QCA's, despite the fact that the design is not optimal due to the higher cell count.	Nanotechnology
[19]	QCA based D-FF	Using a rotated cell type, a robust QCA D flop flop was designed. The robust cell-based MG was used to achieve this. There is a large number of cells in MG, so the design is not compact due to the large number of cells.	Nanotechnology
[25]	QCA based synchronous counter	In this novel 3-bit synchronous counter, the MG and IV are employed, but the design is not optimal in terms of performance. The presented work discusses the design of synchronized negative edge-triggered 3-bit circuits.	Nanotechnology

In addition, the entire layout was designed using look-up tables. This layout view shows a decoder on top, a memory in the middle, and an OR tree on the bottom. As part of this work, a CLB-based layout architecture is also discussed. A nanotechnology approach-based quantum dot layout is further used to simulate all the 4 to 16 decoders, memory, and LUT and CLB blocks. In the published work [12], authors have target with a modular decoder that uses rotated cells with a crossing approach, but the crossing cells create a higher latency due to more delay. Additionally, the design of the 3-to-8 decoder is based on the implementation of the 2-to-4 decoder. As a result of the long length of the wires, it takes a lot more time and information to transfer from the input node to the output node. It is worth noting that the design is not robust and the design of a 2- to-4 decoder takes 4 MG, while the design of a 3-to-8 decoder takes 8 MG. According to the design of the 2 to 4 decoder and the 3 to 8 decoder, the clock phases are 7, and 11, respectively. The literature review work is presented in Table 1.

3. RELATED WORK

QCA is a system that consists of cells made up of electrons [9]. It is important to understand that QCA is a complex network of interconnected cells that interact with each other to form a polarization value used to store information. One of the most significant differences between And gates and Or gates relates to the value of polarization. In Fig 1a,

we can see the polarization value. Using polarization, electrons are adjusted to fit within the diagonals of the cell so that we can represent And gate and Or gate as shown in Fig 1b. This results in the And gate having -1 polarization, which represents Binary '0', and the Or gate having +1 polarization, which represents Binary '1' [10].

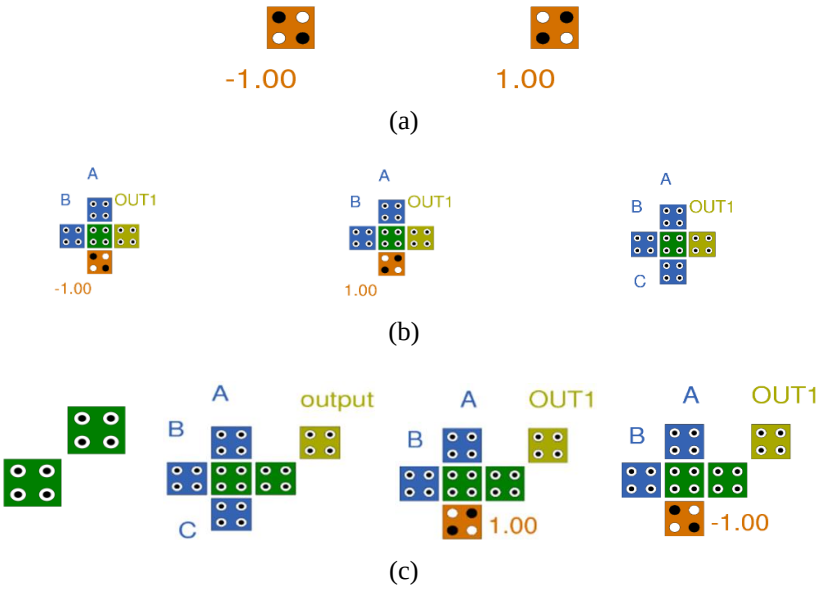


Fig. 1 QCA glance (a) Polarization (b) And, Or, Majority gate (c) Inverter, minority, NOR, NAND

The majority gate has been designed with 3 inputs, 5 inputs, 9 inputs [11]. The 3-input majority gate (MG) design is the fundamental design to understand majority gates as shown in Fig 1b. This design uses three inputs A, B, and C to design the MG. As its name suggests, it works based on the majority of inputs. When the inputs A, B, and C are polarized in the majority, the centre cell, is called the driver cell. The expression for MG as M (A, B, C) = AB + BC + CA. Hence, any input has worked as a polarization concept based on the following equation: M (A, B, 0) = AB + B*0 + 0*A=AB. Polarization '-1' works similarly to store binary '0'. As a result, substituting in the majority gate equation gives the result as And gate. Therefore, if we give polarization '-1', the majority gate behaves like an And gate. The same is true when you give 1 polarization. Taking an example, we can consider C as the polarization input; M (A, B, 1) = AB + B*1 +1*A= AB + B + A=A+B. Here, '1' polarization is given as input then as shown above it works as Or gate. Hence, when polarization is 1 then the majority gate works like Or gate. In the CMOS-based inverter, the behaviour of the output is the opposite of the input. Similar to QCA designers, we also make use of inverters. In the case where two cells are kept diagonally close to each other, then we will get an output that is opposite to the input as shown in Fig 1c. By using this method, we can design different circuits that can be used in a variety of different designs. The following is a list of some of the designs that can be found for an inverter and others. The design of NAND and NOR is depicted in Fig 1c. In

the QCA designer, there are three types of wiring as mentioned 90-degree wiring, 45-degree wiring and 45 and 90-degree wiring as shown in Fig. 2a, 2b, and 2c, respectively.

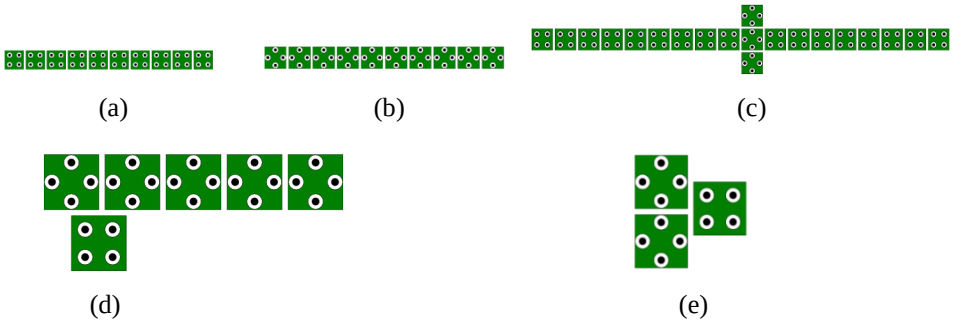


Fig. 2 QCA wiring (a) 90-degree wiring (b) 45-degree wiring (c) 45 and 90-degree wiring (d) Horizontal translation (e) Vertical translation

In QCA design the 90-degree wiring is shown in Fig 2a. This is similar to that of wire, which shows the propagation of binary value through the length of the wire. The propagation of input is from left to right. The left end will be provided with the input and the right end will be the output. The above shown Fig 2a is a horizontal 90-degree wiring. If we want an inverter in this 90-degree wiring we should keep two cells oriented diagonally with 90-degree wiring as shown in Fig 2c. In this 45-degree wiring, the value has been propagating information flow from the length of the wire. The advantage of this wiring is we can get both outputs (same as well as inverted). All the even places will give us the complemented output in a 45-degree wire. To get the output we should place a 90-degree oriented cell to the even place. To get the same value we need to place a 90-degree oriented cell at the odd place. In the rotate wire concept, we use it to rotate the cell to get inverted as well as the same output. That is the cell's electrons get rotated into a 45-degree orientation. In this design, it is easy to get inverted output by using the rotate concept without the need to use an explicit inverter. There are two ways to translate QCA design: horizontally and vertically. The translate option is used in the QCA Designer tool when we put position a/2, where a is the wire's length. Cells in the QCA design have a default width and height of 18. When the translate option is used (18nm/2 = 9nm), the width and height are changed as shown in Fig 2d, 2e, respectively. In QCADesigner, the blue colour cells represent input, the yellow colour cells represent output, the green colour represents clock 0, the pink colour represents clock 1, the light green colour represents clock 2, and the white colour represents clock 3. Field coupling QCA technology uses four types of clocks to flow information [12]. The clocking functionality of QCA Designer plays a major role [13]. The clocking process consists of four stages, namely Switch, Hold, Release, and Relax [14]. In QCA Designing, clocking is used when crossing occurs, so that the majority gate receives the same input. When crossing occurs, we can obtain the same information by changing the clocks. QCAPro used the kink energy to estimate the low and high energy dissipation values. Quantum Dot Cellular Automata pro (QCAPro) is used to determine the power dissipation or the energy dissipation of QCA cells [15]. As part of the QCAPro, four types of colour coding have been implemented: Black colour – this colour indicates that cells in the QCA layout dissipate the most power, Orange colour

– this colour indicates that cells in the QCA layout dissipate medium power, and Yellow colour – this colour indicates that cells in the QCA layout dissipate the least power [16]. Molecular QCA can be used in nanoscale computing devices which are based on QCA [3]. There are four phases used in the QCA clocking zone which are switch, hold, release, and relax. As all of the clocking zones are synchronized in a manner that makes design both reliable and fast, each clocking zone is synchronized.

In a cell, polarization is governed by the amount of energy associated with the polarization of the cell. A polarization measurement with four dots can be defined by equation (1)

$$p = \frac{(\rho_1 + \rho_3) - (\rho_2 + \rho_4)}{\rho_1 + \rho_2 + \rho_3 + \rho_4} \tag{1}$$

In the era of low-power devices. The processor is the main device in computing devices for processing data bits and is used for a variety of emerging applications in the home to advance scientific research [1]. The processor is an integral part of compact devices, such as thermostats, smart appliances, microcontrollers and wearable devices [1]. When the processor device balances low-power designs, the system's performance increases [2].

A low-power balanced processor is used in this scenario to achieve high-performance computations at low power consumption. In most communication devices, built-in modules include Wi-Fi, Bluetooth, and a Wi-Fi port so that servers can be connected, as well as heterogeneous and homogeneous networks, sensors and actuators, Alexa-enabled devices, mobile devices, and Zigbee devices. IoT devices use MQTT (Message Queuing Telemetry Transport), CoAP (Constrained Application Protocol), and HTTP for communication and connectivity [2]. Fig. 3 shows the complete detailed connectivity for IoT devices and many other low-power computing devices.

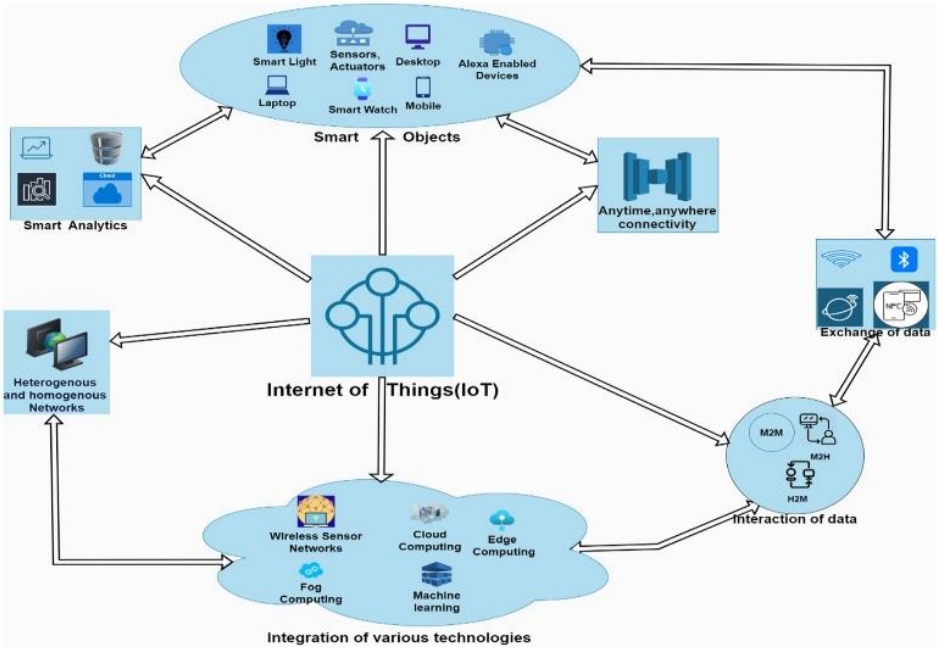


Fig. 3 IoT connectivity with processor [26]

4. The Proposed Layout of the Subcomponent of the Processor

Central processing units (CPU) are controlled by their processors, which are the brains of computers [1]. We have designed various sub-components of processors in this paper. It is shown in the block diagram of the processor that there are different modules. Module 1 consists of the design of a 2-to-4 decoder, module 2 consists of the design of a 3-to-8 decoder, and module 3 consists of the sequence counter as shown in Fig 4a. As a work of this paper, a potential energy estimation and layout design have been presented for a 2-to-4 decoder. We have implemented a robust layout of 3-to-8 decoder, D-FF and sequence counter with the help of the proposed algorithm in this section. With the help of instruction registers, a multiplexer combines these modules and performs processor computations. The block diagram of the processor with its subcomponents is shown in Fig 4a. Fig 4b shows the complete layout flow with a QCA-based layout for each processor subcomponent. With the use of bottom to top approach, we have created the design flow of how higher-level subcomponents have been used for the processor design as shown in Fig 4b.

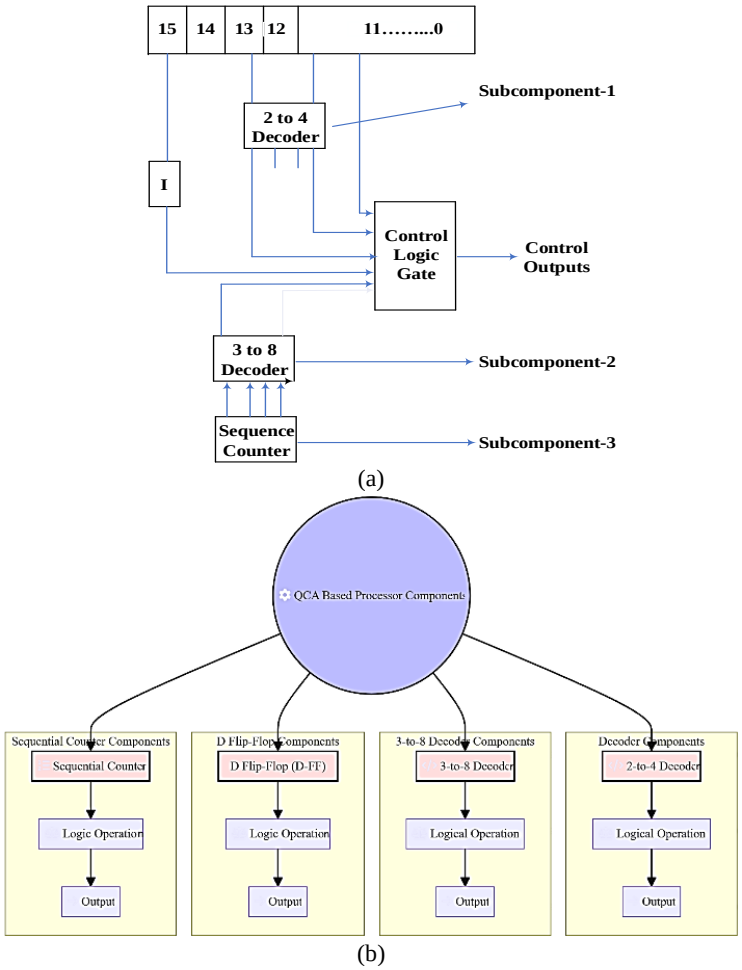


Fig. 4 The processor (a) The block diagram of a processor (b) Subcomponent view in QCA Processor

4.2. The proposed Subcomponent 2 to 4 decoder

In this section, we have explained how to implement a 2-to-4 decoder using QCA Designer in the nanocomputing framework. We know that a decoder takes n inputs and produces 2^n outputs. The output consists of binary data represented by each node. There are many applications for decoders in the digital world, including memory addressing, data multiplexing without select line, CPU design as part of a control unit, and display functions with a seven-segment display. There are several benefits to the design of a 2-to-4 decoder layout in QCA, one of which is a high level of performance and high computing speed as a result of the layout. It is for this reason that QCA makes use of a quantum-dot architecture that encodes binary states by using polarization in cells rather than voltage to encode binary states. The signal flow through the wires is synchronized with the help of clocking zones that allow the flow of signals through wires.

We develop decoders using the coplanar technology-based QCA in this paper. The concept presented here uses a coplanar technique to implement 2-to-4 decoders in QCA designer. To create four majority gates, the QCA layout for this design uses expressions used for the 2-to-4 decoders. In our bottom-to-top approach, the first step is to obtain expressions from the truth table using K-maps to get the expressions from the truth table. The next step involves drawing a block diagram with a majority gate. When using the QCA Designer tool, a layout is created by using four majority gates. This layout is designed using four majority gates. Additionally, the amount of cells used in this decoder layout is much lower than in other decoder layouts in the literature. Expressions for the 2 to 4 decoder are as follows in equations 2, 3, 4 and 5

$$Y3 = M(A, B, C) = M(A, B, 0) = A * B. \tag{2}$$

Similarly,

$$Y2 = M(A, \bar{B}, C) = M(A, \bar{B}, 0) = A * \bar{B}. \tag{3}$$

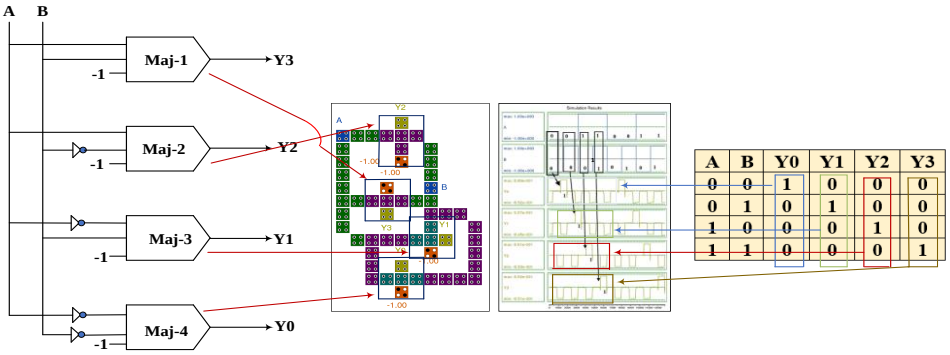
Similarly,

$$Y1 = M(\bar{A}, B, C) = M(\bar{A}, B, 0) = \bar{A} * B. \tag{4}$$

Similarly,

$$Y0 = M(\bar{A}, \bar{B}, C) = M(\bar{A}, \bar{B}, 0) = \bar{A} * \bar{B}. \tag{5}$$

By using the QCA Designer tool version 2.0.3, we can simulate the 2- to-4 decoder. The Bi-stable approximation is selected during the setup of the simulation engine to obtain simulation outputs. The block diagram, layout and simulation results for the 2-to-4 decoder, implemented in the nanotechnology-based tool QCA, as shown in Fig 5a, 5b and



(a) (b) (c)

Fig. 5 Design of 2 to 4 decoder (a) Block diagram (b) Layout (c) Outputs

5c, respectively. In QCADesigner, a vector table is created using the inputs of the truth table, and the outputs are shown in Fig 5c.

4.3. Potential Energy calculation in decoder design

This section presents potential energy dissipation analysis due to cell interaction in the design of a 2-to-4 decoder. The logic computation in QCA is based on cell interaction, and each cell interacts based on polarization. A bistable approximation has been selected as the default simulation engine in the QCADesigner tool. The 2-to-4 decoder design has physical proof showing the measurement of potential energy with the target cell and radius of effect (65nm). Using QCA technology, the potential energy of a design has been calculated by considering the surrounding environment as shown in equation (2). In the default setting in the bistable approximation engine, the default cell size is 18nm in length and 18nm in breadth. The distance between adjacent cells is 2 nm and the radius of effect is 65nm by default setting. In each cell, there is a square shape of the cell with a dimension of 18x18nm. An important aspect when calculating potential energy is the consideration of the positive and negative polarization of cells (P=-1 and P=+1). Fig 6 shows the flow of the total potential energy calculation. Fig 7 shows all electron positions on cell-1, as shown in (x1, y1) and (x2, y2). Affected cells are indicated by the X and Y symbols. Accordingly, if we consider the positive or negative polarization of Cell-1 to be 1 or +1, then the positions of the electrons will be (x1, y1) or (x2, y2). It is important to note that when the polarization of Cell-1 is -1, the positions of the electrons are (x1, y1). In the case where the polarization of Cell-1 is + 1, the electrons' positions are (x2, y2). Depending on the radius of the effect range of cells X and Y, neighbour cell X affects neighbour cell Y. The marked electron on the cell is labelled from e1 to e6 as in the expanded form e1e2, e3e4, e5e6. When polarization P = +1 is considered, the orientation of electrons (x2, y2). There are also six possible e1 to e6 with x2 and e1 to e6 with y2. In this section, all calculations with positive and negative polarization values are presented along with all physical evidence (x1, y1) and (x2, y2). All the potential energy proof are presented in equation (7)- (12)

$$U = \frac{Kq1q2}{d} = \frac{P(let)}{d} \tag{6}$$

$$P = kq1q2 = (9X10^{9})X(1.6x10^{-19})X(1.6x10^{-19}) = 23.04x10^{-29} \tag{7}$$

$$U_{Total} = Sum\ of\ \ all\ potential\ energy = \sum_{P=1}^{m} U_P \tag{8}$$

$$U(In\ one\ cell) = \frac{P}{d} = \frac{23.04x10^{-23}}{d} \tag{9}$$

Consider the electron's orientation (x1, y1) at polarization $P = -1$. There are six possible states associated with x1 and y1 as shown below.

$$U_{e1x1} = \frac{P = 23.04x10^{-23}}{22x10^{-9}} = 10.47x10^{-21}, \quad U_{e2x1} = \frac{P = 23.04x10^{-23}}{42x10^{-9}} = 5.04x10^{-21},$$

$$U_{e3x1}=\frac{P=23.04x10^{-23}}{24x10^{-9}}=8.07x10^{-21},\quad U_{e4x1}=\frac{P=23.04x10^{-23}}{2x10^{-9}}=115.2x10^{-21},$$
$$U_{e5x1}=\frac{P=23.04x10^{-23}}{22x10^{-9}}=10.47x10^{-21},\quad U_{e6x1}=\frac{P=23.04x10^{-23}}{42x10^{-9}}=5.24x10^{-21},$$

$$U_{ex1}=\sum_{m=1}^{m=6}U_{e1x1}+U_{e2x1}+U_{e3x1}+U_{e4x1}+U_{e5x1}+U_{e6x1}=13.63x10^{-20}$$

$$U_{e1y1}=\frac{P=23.04x10^{-23}}{42x10^{-9}}=5.03x10^{-21},\quad U_{e2y1}=\frac{P=23.04x10^{-23}}{58x10^{-9}}=3.96x10^{-21},$$
$$U_{e3y1}=\frac{P=23.04x10^{-23}}{37x10^{-9}}=5.98x10^{-21},\quad U_{e4y1}=\frac{P=23.04x10^{-23}}{22x10^{-9}}=8.07x10^{-21},$$
$$U_{e5y1}=\frac{P=23.04x10^{-23}}{18x10^{-9}}=12.72x10^{-21},\quad U_{e6y1}=\frac{P=23.04x10^{-23}}{22x10^{-9}}=10.47x10^{-21}$$

$$U_{ey1}=\sum\nolimits_{m=1}^{m=6}U_{e1y1}+U_{e2y1}+U_{e3y1}+U_{e4y1}+U_{e5y1}+U_{e6y1}=3.78x10^{-20} \quad (10)$$

For considering the polarization value of P=-1. The first step is to calculate Ue1x1, Ue2x1, Ue3x1, Ue4x1, Ue5x1, Ue6x1, then we calculate Ue1y1, Ue2y1, Ue3y1, Ue4x1, Ue5x1, Ue6y1. After that, add both potential energies.

$$U_{ex1}+U_{ey1}=13.63x10^{-20}+3.78x10^{-20}=17.41x10^{-20} \quad (11)$$

At polarization P=+1, the electron is in the direction of polarization (x2, y2). Below is a list of six possible states that can be associated with x2 and y2 as shown in the illustration.

$$U_{e1x2}=\frac{P=23.04x10^{-23}}{42x10^{-9}}=5.03x10^{-21},\quad U_{e2x2}=\frac{P=23.04x10^{-23}}{64x10^{-9}}=3.48x10^{-21},$$
$$U_{e3x2}=\frac{P=23.04x10^{-23}}{20x10^{-9}}=11.5x10^{-21},\quad U_{e4x2}=\frac{P=23.04x10^{-23}}{18x10^{-9}}=12.98x10^{-21},$$
$$U_{e5x2}=\frac{P=23.04x10^{-23}}{22x10^{-9}}=10.47x10^{-21},\quad U_{e6x2}=\frac{P=23.04x10^{-23}}{38x10^{-9}}=6.06x10^{-21},$$

$$U_{ex2}=\sum_{m=1}^{m=6}U_{e1x2}+U_{e2x2}+U_{e3x2}+U_{e4x2}+U_{e5x2}+U_{e6x2}=3.78x10^{-20}$$

$$U_{e1y2}=\frac{P=23.04x10^{-23}}{24x10^{-9}}=8.78x10^{-21},\quad U_{e2y2}=\frac{P=23.04x10^{-23}}{42x10^{-9}}=5.03x10^{-21},$$
$$U_{e3y2}=\frac{P=23.04x10^{-23}}{41x10^{-9}}=5.78x10^{-21},\quad U_{e4y2}=\frac{P=23.04x10^{-23}}{20x10^{-9}}=11.52x10^{-21},$$
$$U_{e5y2}=\frac{P=23.04x10^{-23}}{2x10^{-9}}=115.1x10^{-21},\quad U_{e6y2}=\frac{P=23.04x10^{-23}}{24x10^{-9}}=9.02x10^{-21},$$

$$U_{ey2} = \sum_{m=1}^{m=6} U_{e1y2} + U_{e2y2} + U_{e3y2} + U_{e4y2} + U_{e5y2} + U_{e6y2} = 16.4x10^{-20} \quad (11)$$

As a result, P=+1 is taken into consideration for the polarization value. First, it is necessary to determine Ue1x2, Ue2x2, Ue3x2, Ue4x2, Ue5x2, Ue6x2, then we will calculate Ue1y2, Ue2y2, Ue3y2, Ue4x2, Ue5x2, Ue6y2. To calculate their potential energy, add them both together.

$$U_{ex2} + U_{ey2} = 3.78x10^{-20} + 16.4x10^{-20} = 20.28x10^{-20} \quad (12)$$

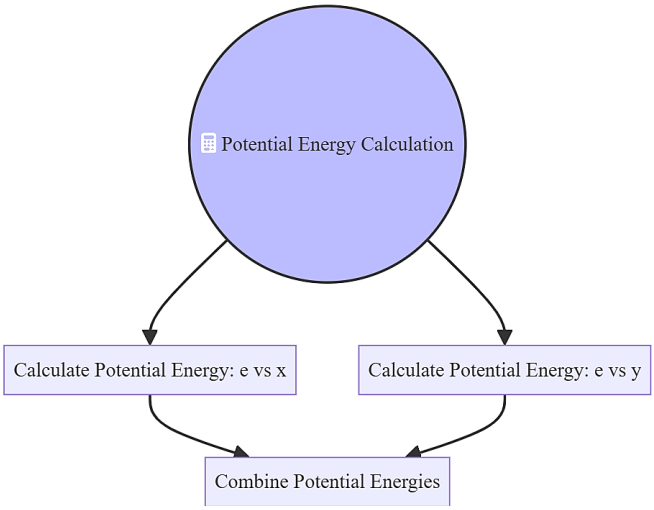


Fig. 6 Method to calculate combined potential energies

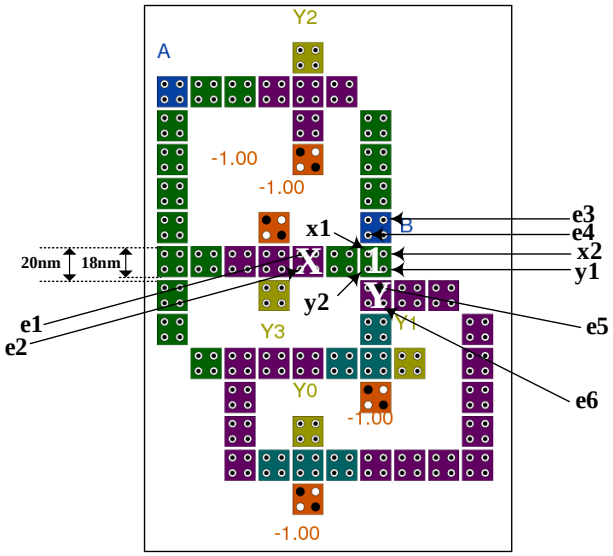


Fig. 7 Energy calculation in the proposed 2-to-4 decoder

4.4. Subcomponent 3- to- 8 decoder

In this section, a 3-to-8 decoding design using a QCA designer is proposed. We have designed the expressions of 3-to-8 decoders using the K-map function. The design does not have a 45-degree orientation due to more polarization change and slow computation. The design of a 3- to- 8 decoder involves the use of 90-degree oriented wires. In this design, using a 45-degree wire has the disadvantage of increasing the circuit's vertical length (i.e., the circuit will require more space) due to its use of a 45-degree wire. Fig. 8 shows the block diagram of the majority gate design of the 3-to-8 decoder. We demonstrate how the QCA designer tool was used to simulate and verify the layout's functionality. Fig. 9a and 9b show a layout design and simulation output using field coupling nanotechnology-based QCA.

According to the algorithm-1 presented in the case of n-to-2^n decoder and taken the example of 3-to-8 in the assumption that the binary input is I=011 (where I2=0, I1=1, I0=1). First go to Step-1, if the binary input is I=011 and convert into decimal $D=0\cdot2^2+1\cdot2^1+1\cdot2^0=0+2+1=3$. There is only one active line in Step 2 when measuring Y3 = 1. As a result of the decoder output, there is a display showing [00010000]. The same approach would likely apply to n-to-2^n decoding as well.

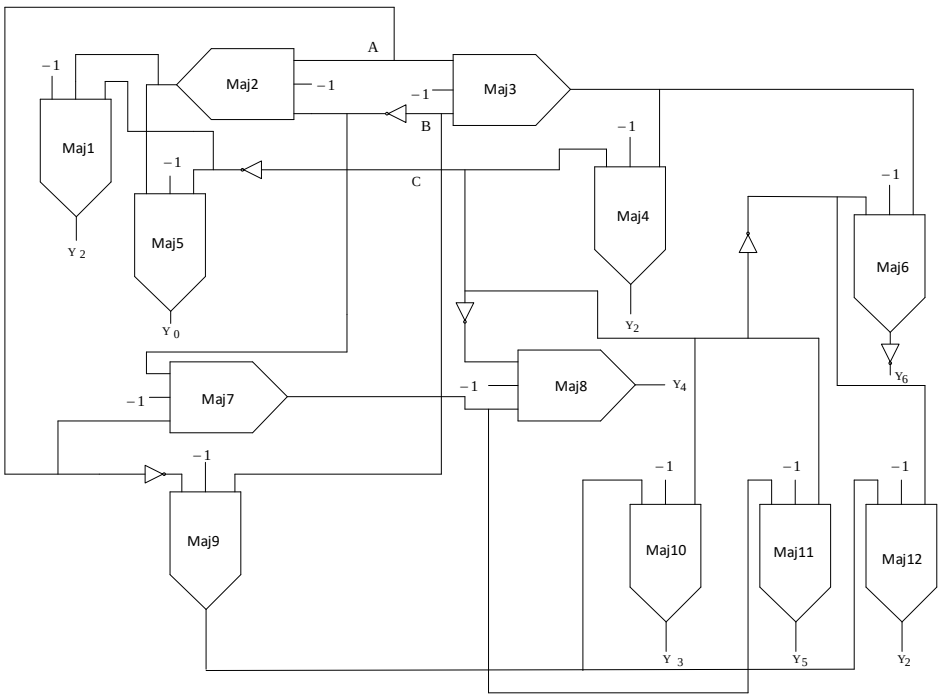
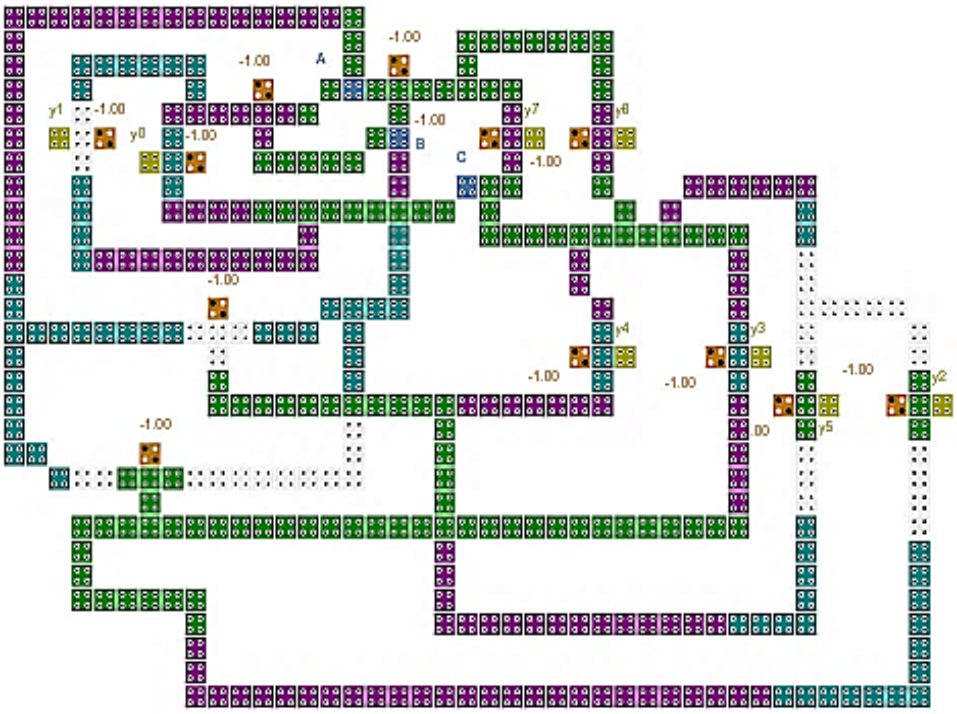
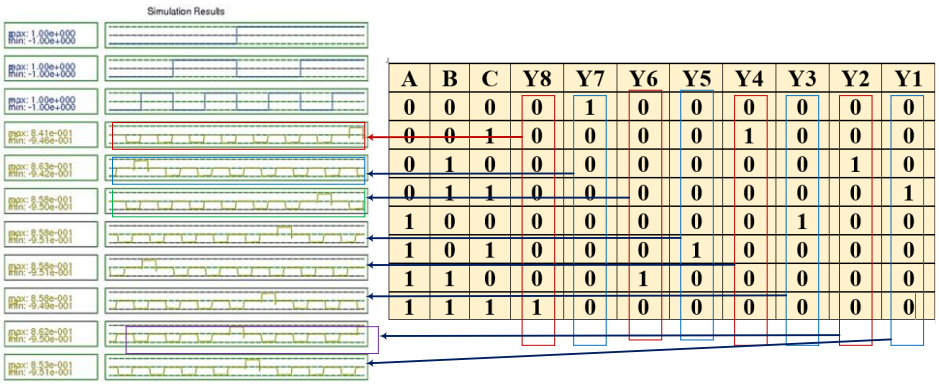


Fig. 8 Block diagram of 3-to-8 decoder



(a)



A	B	C	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1
0	0	0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0
0	1	0	0	0	0	0	0	0	1	0
0	1	1	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	1	0	0
1	0	1	0	0	0	1	0	0	0	0
1	1	0	0	0	1	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

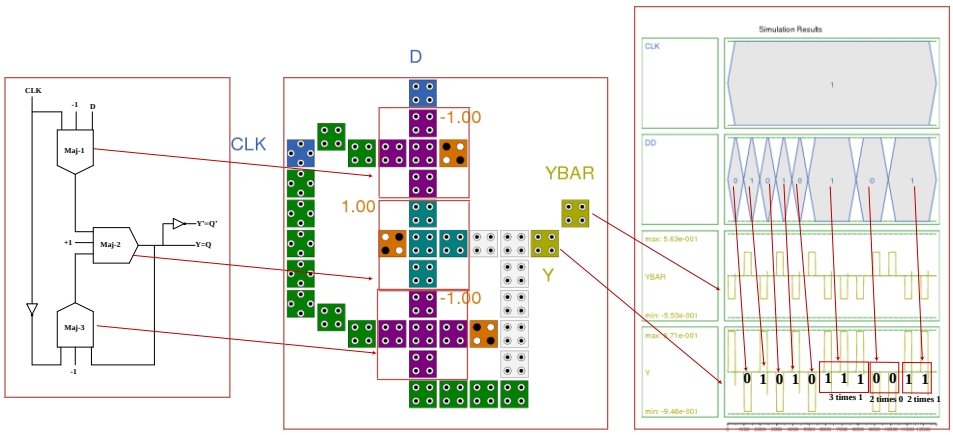
(b)

Fig. 9 Design 3-to-8 decoder (a) layout (b) outputs

Algorithm 1: Synthesis method of n-to-2^n decoder
Inputs: I=(I_{n-1},I_{n-2},…,I_0) where I_{n-1} denote MSB and I_0 denote LSB
Line Outputs: Y=(Y_0,Y_1,…,Y_{2^n-1})
Initialize current state: Reset all output lines (Yp=0 for p=0 to 2^{n-1})
1. **for** loop
Step-1
2. **Measure Q**=$I_{n-1}\cdot 2^{n-1}+I_{n-2}\cdot 2^{n-2}+\cdots+I_1.2^1+ I_0.2^0$
3. Convert **Q** into decimal
Step-2
4. Measure Y_Q=1 (Only one line active)
Step-3
5. Output line Y at index Q
6. **end for**
7. **return** Y

4.5. Subcomponent D flip-flop

The flipflop is used as a memory element, which will store information in the registers that can be used for storing the information. In other words, all flip-flops are sequential circuits. A sequential circuit is highly dependent on the present inputs and past outputs. In the case of flip flops, there are different styles such as the S-R flip flop, J-K flip flop, and D flip flop. This section deals with the design of a D-flip flop using QCA technology. The block diagram as shown in Fig 10a illustrates the basic operation of the D flip flop using the majority gate. In D-FF design, we have used two AND gates at level one and then at the next level we have used an OR gate and a NOT gate in layout design in QCA technology as shown in the block diagram of Fig 10b.



(a) (b) (c)

Fig. 10 D flip-flop QCA Designer (a) Majority design (b) layout (c) outputs

In the block diagram input to the OR gate has been feedback that serves as a memory as is shown in Fig. 10a. The complete layout of a D flip flop using the field coupled QCA

technology as seen in Fig. 10b. We have used a standard characteristic table of the D flip flop to compare the simulation outputs. We know from the characteristic equation of D flip-flop as Qn = D. Therefore, the output will be the same as that of the input. The Qbar has been the inverted output of Qn. The simulation output is presented in Fig. 10c.

In algorithm 2, we present the synthesis method for D-Flip Flop. This circuit synthesis uses three D-FFs with clocks. If the clock is high in the characteristic table, then data input (D) will go to the output node (Q). Otherwise, if CLK=0, the current value is retained in the next stage. In the case of clock transitions, the output Q will change, and the output is measured by the output of the node.

Algorithm 2: Synthesis method of D Flip-flop
Inputs: Data input as D and Clock as CLK
Outputs: Qt+1 and Q't+1
Initialize Clock signal transitions (Positive or Negative Triggered)
Step-1
1. Determine the next stage using Q_{t+1}=Q.CLK'+D.CLK
2. **Step-2**
3. if CLK=0
4. Q_{t+1}=Q #retains its current value
5. else
6. Q_{t+1}=D #Output takes the value of D

Step-3
7. Observe Q_{t+1}
8. end if
9. **end for**
10. **return** Qt+1 and Q't+1

4.6. Subcomponent sequential counter

By using a single and common clock signal, multiple flip flops are triggered simultaneously in sequential designs. The flip flops are triggered simultaneously because a single and common clock signal is used to trigger them all. This type of counter counts binary values between 0 and 7. Counting sequences in this program ranges from 0 to 7. Counting seven times resets this counter's logic and it returns to the first state after one more count. In Fig 11, we have shown a schematic-level design of a sequence counter using D flip-flops with various logic gates. At the layout level, design a sequence counter as shown in Fig. 12. To simulate in QCADesigner, we have used a coplanar cell type and the default settings. Fig 13 illustrates the outcomes of the sequential counter. In Table 2 the simulation is used to verify the effectiveness of the sequential counter based on the simulation results. The layout design uses only coplanar cells since they provide less delay and more rapid signal processing to the output node. Simulation results in Fig 12 show that the signal does not degrade at the output node when the polarization output is high. In algorithm 3, we present the sequence counter-synthesis approach. For the sequence counter, we used D-FF with each state of the counter having outputs Qa, Qb, and Qc. As part of the first step, we reset all three states to zero (Qa=0, Qb=0, and Qc=0). In this case, it represents state 000. Then based on the truth table as shown in Table 2 of

the sequence counter set the logic as Qc toggle on each pulse, Q_b toggle when Q_b=1, Q_a toggle when both Qb and Qc=1. It is necessary to apply the clock and measure all the states. As a binary representation of the state, we will count from 0 to 7 in terms of binary, as follows: 000,001,010,011,100,101,110,111.

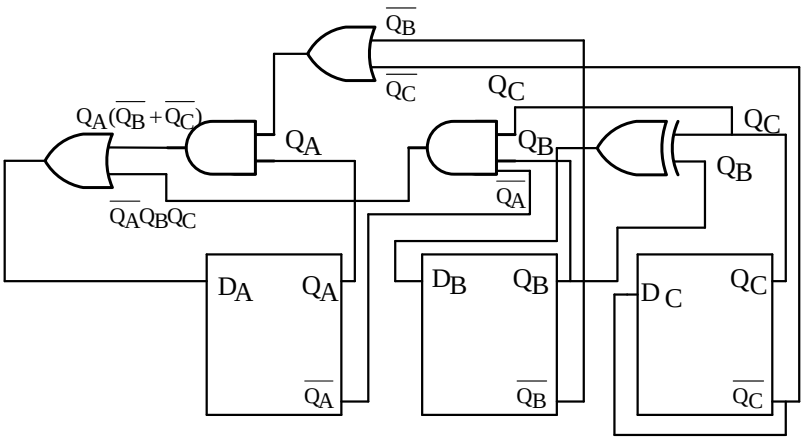


Fig. 11 Block diagram of sequential counter

Table 2 Truth Table of sequential counter

Outputs						Inputs		
Qa	Qb	Qc	Qa+1	Qb+1	Qc+1	D_A	D_B	D_C
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	0
0	1	0	0	1	1	0	1	1
0	1	1	1	0	0	1	0	0
1	0	0	1	0	1	1	0	1
1	0	1	1	1	0	1	1	0
1	1	0	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0

The modelling equations of the sequential counter are presented by equations 13, 14, and 15

$$D_A = Qa*(\overline{Qb}+\overline{Qc})+\overline{Qa}*Qb*Qc \tag{13}$$

$$D_B = \overline{Qb}*Qc+Qb*\overline{Qc} \tag{14}$$

$$D_C = \overline{Qc} \tag{15}$$

In this section, we discussed how Boolean equations have been used to design 3-bit sequence counters. With inputs from a truth table as shown in Table 2, the k-map yields the equations given above. As can be seen from Fig 13, which is a layout implementation of a 3-bit sequential counter. The connections are inputs to each D flip-flop. The

equations are given as inputs to flip flops through majority gates. This 3-bit sequence counter is implemented with flip-flops, ANDs, ORs, and Ex-ORs. Coplanar cells are used in layout designs to demonstrate manufacturing feasibility. A layout design consisted of three-D flip flops, two OR gates, two AND gates, and one Ex-OR gate. This layout design uses a three-input majority gate, which is compact and optimizes the sequence counter. A vector set of inputs is used for simulation in the QCA Designer using Bistable approximation. We can conclude that the layout is working properly based on the functional match of the Truth Table 2. The complete simulation results are presented in Fig 13.

Algorithm 3: Synthesis method of sequence counter
Inputs: D-input (The next state of the system is determined by D input)
Outputs: Qa, Qb, and Qc (Current state), Qa+1, Qb+1, and Qc+1 (Next state)
Initialize current state: Initialize current state 000 to 111, A flip-flop's next state is determined by its D input.
Qa, Qb and Qc change state from 000 to 111
Start: State declaration of D-input to all three Flip-flops
for n=0 to 7 loop
Step-1
Connect the output of ($\sim Q_c$) ⟶ Dc (Right most FF)
Connect the output of (Q_b XOR Q_c) ⟶ Db (Second right most FF)
Connect the output of $Q_a(Q'_b+Q'_c) + Q'_a*Q_b*Q_c$ ⟶ Da (Left most FF)
Step-2
Apply clock and stage change to next state
Qc toggle on each pulse
Qb toggle when Qb=1
Qa toggle when both Qb and Qc=1
Step-3
Update all states Qa to Qc based on current state and D inputs
Step-4
Observe all states 000, 001, 010, 011, 100, 101, 110, and 111
Step-5
If state 111 the next clock will reset the state and back to the original state (000)
end for
return states

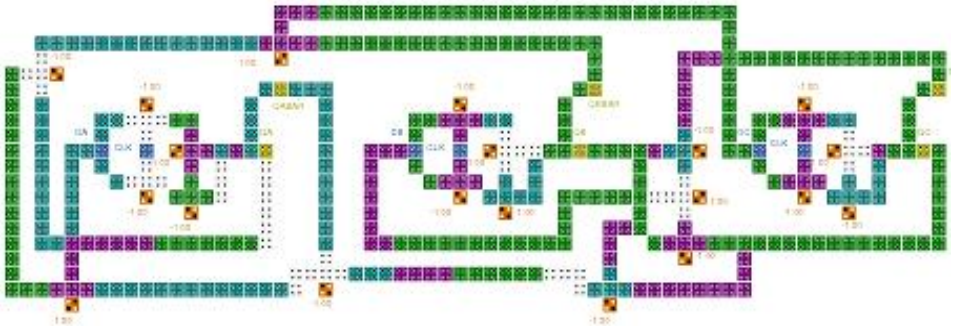
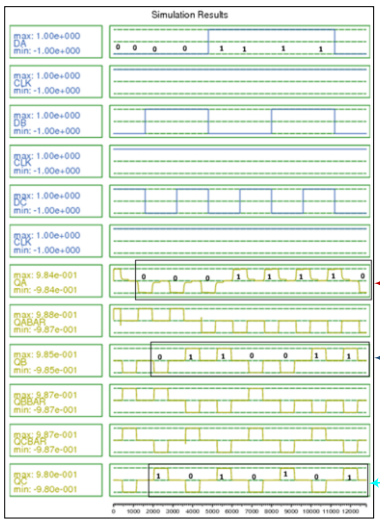


Fig. 12 QCA Designer layout of 3-bit sequential counter



Outputs						Inputs		
Qa	Qb	Qc	Qa+1	Qb+1	Qc+1	D_A	D_B	D_C
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	0
0	1	0	0	1	1	0	1	1
0	1	1	1	0	0	1	0	0
1	0	0	1	0	1	1	0	1
1	0	1	1	1	0	1	1	0
1	1	0	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0

Fig. 13 Simulation results of Sequence counter

4.7. Subcomponent of Instruction Register

In the instruction register, instructions are stored that are being executed. In the CPU, it is part of the control unit. Memory has been loaded with instructions that are currently being executed or decoded. Control signals are produced as outputs that control various processing elements to execute instructions. The instruction register consists of registers for storing inputs and memory elements, and decoders for decoding instructions. After each iteration, the program counter is incremented. Mostly, CPUs are composed of instruction registers.

4.8. Energy dissipation results of the proposed 2-to-4 decoder layout

Power estimation is a useful parameter for analyzing power-aware designs in nanoscale computing [11]. QCA design defines switching power as a change in polarization value as a result of polarity change [12]. Tunneling caused unintended leakage of power. The heat generated during computation contributes to thermal power in the GHz frequency range [13]. The authors Timler and Lent [3] proposed a methodology for estimating power based on the size and dimension of cells. Array cells are represented by Hamiltonian matrices as shown in equation 16.

$$H = \begin{bmatrix} -\frac{E_k}{2}\sum_t c_i f_{ij} & -\gamma \\ -\gamma & \frac{E_k}{2}\sum_t c_i f_{ij} \end{bmatrix} = \begin{bmatrix} -\frac{E_k}{2}(C_{j,1} + C_{j+1}) & -\gamma \\ -\gamma & \frac{E_k}{2}(C_{j,1} + C_{j+1}) \end{bmatrix} \quad (16)$$

In the Hamiltonian matrix fi,j represents a geometrical factor with consideration of the interaction between cell number i and j. A cell is represented by Ci [3].

The E_{ij} symbol represents the electrostatic interaction between electrons situated in two cells i and j as shown in equation (17).

$$E_{ij} = \frac{1}{4\pi\varepsilon_0\varepsilon_r}\sum_{n=1}^{4}\sum_{m=1}^{4}\frac{q_{i,n}q_{j,m}}{\left|r_{i,n} - r_{j,m}\right|} \tag{17}$$

There is a measurement of energy at each clock cycle relating to the plank constant and the coherence vector as expressed in equation (18)

$$E = \langle H \rangle = \frac{\hbar}{2}\cdot\vec{\Gamma}\cdot\vec{\lambda} \tag{18}$$

Nanocomputing devices require low power dissipation, which increases their performance over long periods [14]. The battery life of a device will be prolonged if the power sources consume less power [17]- [19]. It is more important to pay attention to the energy factor in the nanocomputing layout to meet the current research trends [20]. When determining the amount of dissipated energy, it is necessary to use two tools, QCADesigner1.4.1 and QCAPro. For layout considerations, QCADesigner1.4.1 is being used to implement a 2-to-4 decoder layout. QCA pro uses several types of power estimation techniques, including average dissipation, maximum dissipation, minimum dissipation, and leakage dissipation, as shown in Table 3. Figure 14 shows the thermal layout and polarization map generated by the QCAPro to determine the energy dissipation for each cell in the QCA layout of the 2-to-4 decoder. Table 3 shows the different kinds of energy dissipation that can be generated for layout scenarios by simulating layout and setting kink energy. By using QCA-Pro, we have estimated the parameters for energy dissipation in 2-to-4 Decoders. QCA-Pro has been used to obtain the energy dissipation parameters and thermal map following a successful simulation of the QCA design. According to the thermal diagram below, the cell with the most power dissipates the most heat. On the colour scale, black indicates a higher dissipation of energy, while white indicates a lower dissipation of energy. As shown in Fig 14, a black cell and red cell indicate very high power dissipation, a yellow cell indicates moderate power dissipation and a white cell indicates the least power dissipation.

Table 3 Energy estimation results for 2-to-4 decoder

	Level $0.5E_k$ (meV) x 10^-2		Level $1E_k$ (meV) x 10^-2		Level $1.5E_k$ (meV) x 10^-2	
	[11]	New	[11]	New	[11]	New
Max Energy dissipation	65.99	45.3	74.43	67.443	86.99	67.9
Avg Energy dissipation	36.77	23.156	50.95	42.554	67.68	53.596
Min Energy dissipation	11.10	9.58	29.66	31.35	49.93	39.56
Avg leakage Energy dissipation	11.78	8.32	30.71	22.77	51.12	37.18
Avg Switching Energy dissipation	24.99	17.25	20.79	12.77	16.55	9.78

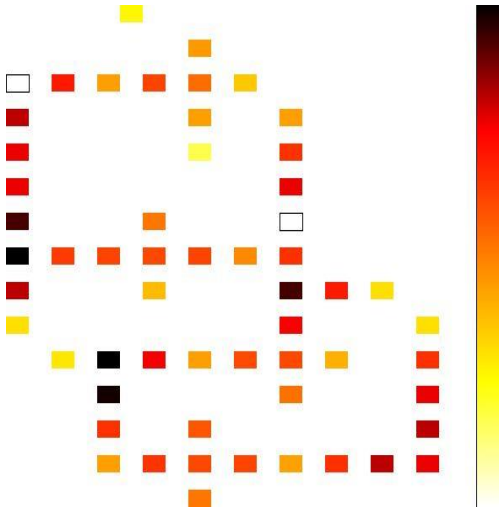


Fig. 14 Thermal layout of 2-to-4 decoder using QCAPro tool

5. Comparison of Performance Analysis with Existing Designs

An analysis of the performance of existing designs is provided in this section along with proposed designs of 2-to-4 and 3-to-8 decoders, D-FF, and sequence counter designs. Compared to state-of-the-art designs, the proposed design is more compact, has less latency, and occupies less space. The computation between inputs and outputs is fast due to the use of coplanar cells and the minimization of clock zones. An innovative architecture and a few minority gates enable the D-FF design to be compact. QCA cells are arranged in this way to result in a small area and no crossover within the sequence counter, which results in less latency and fast computations. Coplanar techniques are used in QCA to design decoders, D-FFs, and sequence counters using field coupling nanotechnology. As shown in Table 4, the proposed 2-to-4 decoder is more compact and optimizes parameters than the existing design, demonstrating superior performance.

Further, in comparison to the previous studies, the average savings for 2-to-4 decoders are 79.25%,81.57%, 85.71%, and 99.62% in terms of the number of cells, footprint area, latency, and area cost, respectively, when compared to the existing design, in [13] as shown in Table 4. Similarly, for a 3-to-8 decoder, the average savings are 64.68% and 76.78% in the number of cells, footprint area, and latency, respectively, as compared to existing work in [12] as depicted in Table 5. In comparison, the average savings for the proposed D Flip-flop are 72.41%, 68.75%, 66.66%, and 96.52 % savings in the number of cells, footprint area, latency, and cost, respectively as compared to prior work in [17] as shown in Table 6. There is no crossover in the proposed sequence counter, so it has a low latency and a high processing speed. The performance comparison of the sequence counter is presented in Table 7.

Table 4 Comparative analysis of 2-to-4 decoder with state-of-the-art work

Layout in QCA	Number of cells	Footprint area (μm^2)	Latency	Cost=Area x latency^2
[8]	212	0.25	6	9
[9]	296	0.43	11	52.03
[10]	361	0.44	9	35.64
[11]	200	0.22	4	3.52
[12]	270	0.38	7	18.62
[13]	268	0.30	7	14.7
[14]	318	0.50	7	24.5
[16]	87	0.10	2.5	0.625
Proposed	56	0.07	1	0.07
% improvement w.r.to [13]	79.2592593	81.57895	85.71429	99.62406
% improvement w.r.to [14]	82.3899371	86	85.71429	99.71429
% improvement w.r.to [16]	35.6321839	30	60	88.8

Table 5 Comparative analysis of 3-to-8 decoder with state-of-the-art work

Layout in QCA	Number of cells	Footprint area (μm^2)	Latency	Cost=Area x latency^2
[13]	1076	2.24	-	
[16]	476	0.57	9	46.17
Proposed	380	0.52	5.5	15.73
% improvement w.r.to [13]	64.6840149	76.78571	--	-
% improvement w.r.to [16]	20.1680672	8.77193	38.88888889	65.93026

Table 6 Comparative analysis of D Flip-flop with state of artwork

Layout in QCA	Number of cells	Footprint area (μm^2)	Latency	Cost=Area x latency^2
[17]	116	0.16	3	1.44
[18]	66	0.08	1.5	0.18
Design 1 [20]	48	0.05	1	0.05
Design 2 [20]	84	0.09	2.75	0.680625
Design 3 [20]	84	0.09	2.75	0.680625
Design 4 [20]	120	0.14	3.25	1.47875
Proposed	39	0.06	1	0.05
% improvement w.r.to [17]	66.3793103	62.5	66.66667	95.83333
% improvement w.r.to [18]	40.9090909	25	33.33333	66.66667

Table 7 Comparative analysis of sequence counter with state-of-the-art work

Layout in QCA	Number of cells	Footprint area (μm^2)	Latency	Cost=Area x latency^2
[20]	140	0.16	2	0.64
[21]	616	1.2	5	30
[22]	428	0.48	2	1.92
[23]	238	0.36	2.25	1.8225
[24]	196	0.22	4	3.52
[25]	174	0.20	3	1.8
Proposed	288	0.38	3	3.42

6. Conclusion and Future Work

In this article, we propose a subcomponent of a processor that optimizes overall efficiency in the field of QCA technology field coupling with energy dissipation proof, which is a first in state-of-art work. There are four sub-components that it is intended to target: 2-to-4 decoder, 3-to-8 decoder, D-FF, and sequence counters. Sequence counters are important parts of the processor because they count state. This article discusses how the layout is implemented in QCA using a coplanar approach with the use of novel algorithms. As per the literature, the coplanar technique can be used to manufacture a device. Therefore, keeping these criteria in mind, we have used this technology to implement the layout of a 2-to-4 decoder, 3-to-8 decoder, D-FF, and sequence counters. To verify its correctness, simulations performed using the QCADesigner tool with a nanotechnology approach have been performed concerning this new processor sub-component. Using QCADesigner the proposed design has been used to check the correctness of circuits. The proposed structure is compact, the number of cells is low, and the latency is low. IoT applications require low-power processors to increase the long run time of the sensor data for computation. A decoder functions as a building block for complex systems such as memory, ALU, PLA, and microcontrollers. Using nanotechnology, the proposed decoder will enable many digital designs to be compacted further.

References

[1] İ. Taştan, M. Karaca and A. Yurdakul, "Approximate CPU Design for IoT End-Devices with Learning Capabilities", *Electronics*, vol. 9, no. 1, p. 125, 2020.

[2] D. Thomas, R. McPherson, G. Paul and J. Irvine, "Optimizing Power Consumption of Wi-Fi for IoT Devices: An MSP430 Processor and an ESP-03 Chip Provide a Power-Efficient Solution", *IEEE Consum. Electron. Mag.*, vol. 5, no. 4, pp. 92-100, 2016.

[3] J. Timler and C. S. Lent, "Power Gain and Dissipation in Quantum-Dot Cellular Automata", *J. Appl. Phys.*, vol. 91, no. 2, pp. 823-831, 2002.

[4] S.-S. Ahmadpour, N. J. Navimipour, S. Kassa, N. K. Misra and S. Yalcin, "An Ultra-Efficient Design of Fault-Tolerant 3-Input Majority Gate (FTMG) with an Error Probability Model Based on Quantum-Dots", *Comput. Electr. Eng.*, vol. 110, p. 108865, 2023.

[5] M. Patidar, D. A. Kumar, P. William, G. B. Loganathan, A. M. Billah and G. Manikandan, "Optimized Design and Investigation of Novel Reversible Toffoli and Peres Gates Using QCA Techniques", *Measurement: Sensors*, vol. 32, p. 101036, 2024.

[6] S. R. Kassa, N. K. Misra and R. Nagaria, "Design, Synthesis and Assessment of QCA Primitives of 5-input Majority Gate in Field-Coupled QCA Nanotechnology", *Optik*, vol. 271, p. 170059, 2022.

[7] N. K. Misra, B. Sen and S. Wairya, "Novel Conservative Reversible Error Control Circuits Based on Molecular QCA", *Int. J. Comput. Appl. Technol.*, vol. 56, no. 1, pp. 1-17, 2017.

[8] M. Kumar and T. N. Sasamal, "An Optimal Design of 2-to-4 Decoder Circuit in Coplanar Quantum-Dot Cellular Automata", *Energy Procedia*, vol. 117, pp. 450-457, 2017.

[9] M. Kianpour and R. Sabbaghi-Nadooshan, "A Novel Quantum-Dot Cellular Automata CLB of FPGA", *J. Comput. Electron.*, vol. 13, pp. 709-725, 2014.

[10] R. Jayalakshmi and R. Amutha, "An Approach Towards Optimisation of 3 to 8 Decoder Using 5 Input Majority Gate with Coplanar Crossing in Quantum Dot Cellular Automata", *AIP Conference Proceedings*, vol. 1966, no. 1, p. 020039, May 2018.

[11] R. Sherizadeh and N. J. Navimipour, "Designing a 2-to-4 Decoder on Nanoscale Based on Quantum-Dot Cellular Automata for Energy Dissipation Improving", *Optik*, vol. 158, pp. 477-489, 2018.

[12] M. Kianpour and R. Sabbaghi-Nadooshan, "A Novel Modular Decoder Implementation in Quantum-Dot Cellular Automata (QCA)", In Proceedings of the 2011 International Conference on Nanoscience, Technology and Societal Implications, Dec. 2011, pp. 1-5.

[13] M. Kianpour and R. Sabbaghi-Nadooshan, "A Conventional Design and Simulation for CLB Implementation of an FPGA Quantum-Dot Cellular Automata", *Microprocess. Microsyst.*, vol. 38, no. 8, pp. 1046-1062, 2014.
[14] T. Lantz and E. Peskin, "A QCA Implementation of a Configurable Logic Block for an FPGA", In Proceedings of the 2006 IEEE International Conference on Reconfigurable Computing and FPGA's (ReConFig 2006), Sept. 2006, pp. 1-10.
[15] T. Lantz, *A QCA Implementation of a Look-Up Table for an FPGA*, Graduate Paper, Electrical Engineering Department, Rochester Institute of Technology, 2006.
[16] B. K. Bhoi, N. K. Misra and Prity Bharti, "Cost-Effective Quantum Dot Architecture of Decoder Circuit and its Futuristic Scope in the Nanoelectronics", *Scientia Iranica Articles*, in Press, 2022.
[17] X. Yang, L. Cai and X. Zhao, "Low Power Dual-Edge Triggered Flip-Flop Structure in Quantum Dot Cellular Automata", *Electron. Letters*, vol. 46, no. 12, pp. 825-826, 2010.
[18] A. Vetteth, K. Walus, V. S. Dimitrov and G. A. Jullien, "Quantum-Dot Cellular Automata of Flip-Flops", *ATIPS Laboratory*, vol. 2500, pp. 1-5, 2003.
[19] S. Hashemi and K. Navi, "New Robust QCA D Flip Flop and Memory Structures", *Microelectron. J.*, vol. 43, no. 12, pp. 929-940, 2012.
[20] A. H. Majeed, E. Alkaldy, M. S. bin Zainal and D. Bin MD Nor, "Synchronous Counter Design Using Novel Level Sensitive T-FF in QCA Technology", *J. Low Power Electron. Appl.*, vol. 9, no. 3, p. 27, 2019.
[21] X. Yang, L. Cai, X. Zhao and N. Zhang, "Design and Simulation of Sequential Circuits in Quantum-Dot Cellular Automata: Falling Edge-Triggered Flip-Flop and Counter Study", *Microelectron. J.*, vol. 41, no. 1, pp. 56-63, 2010.
[22] S. Sheikhfaal, K. Navi, S. Angizi and A. H. Navin, "Designing High Speed Sequential Circuits by Quantum-Dot Cellular Automata: Memory Cell and Counter Study", *Quantum Matter*, vol. 4, no. 2, pp. 190-197, 2015.
[23] S. Angizi, M. H. Moaiyeri, S. Farrokhi, K. Navi and N. Bagherzadeh, "Designing Quantum-Dot Cellular Automata Counters with Energy Consumption Analysis", *Microprocess. Microsyst.*, vol. 39, no. 7, pp. 512-520, 2015.
[24] M. M. Abutaleb, "Robust and Efficient Quantum-Dot Cellular Automata Synchronous Counters", *Microelectron. J.*, vol. 61, pp. 6-14, 2017.
[25] Z. Amirzadeh and M. Gholami, "Counters Designs with Minimum Number of Cells and Area in the Quantum-Dot Cellular Automata Technology," *Int. J. Theor. Phys.*, vol. 58, no. 6, pp. 1758-1775, 2019.
[26] K. Kaur, A. Kaur, Y. Gulzar and V. Gandhi, "Unveiling the Core of IoT: Comprehensive Review on Data Security Challenges and Mitigation Strategies", *Front. Comput. Sci.*, vol. 6, p. 1420680, 2024.