*Characterization and Application of Nanomaterials* 2024, 7(2), 6401.
https://doi.org/10.24294/can.v7i2.6401

Article

# QCA-based design of polar encoder circuit

**Sravan Kumar Vittapu, Ravichand Sankuru, Chepuri Rakshana, Beeradhar Mahesh*, Amudha Naga Teja**

Department of ECE, Nalla Narasimha Reddy Education Society's Group of Institutions, Chowdariguda, Hyderabad 500088, Telangana, India
*** Corresponding author:** Beeradhar Mahesh, beeradharmahesh@gmail.com

**CITATION**

Vittapu SK, Sankuru R, Rakshana C, et al. QCA-based design of polar encoder circuit. Characterization and Application of Nanomaterials. 2024; 7(2): 6401. https://doi.org/10.24294/can.v7i2.6401

**ARTICLE INFO**

Received: 14 May 2024
Accepted: 8 July 2024
Available online: 26 July 2024



**Abstract:** In the last few decades, nano-electronic devices have been manufactured using VLSI technology. Over the past four decades, IC technology has been growing by using CMOS technology successfully, but this CMOS technology has a scaling limitation. To overcome this scaling limitation, QCA (quantum dot cellular automata) emerges as an alternative. This work is the implementation of the design of a polar encoder using QCA technology. This design is a single-layered and even bottom-up approach technique. The Polar code is more efficient and has less energy dissipation compared to the turbo code and conventional codes (CC). This design explores (8:4). A Polar encoder is designed to have fewer cells and area compared to the turbo encoder and conventional encoder. The proposed design is implemented using the QCA designer tool.

**Keywords:** ALU; CMOS; nano technology; polar encoder; QCA; VLSI

## 1. Introduction

The IC technology is growing faster and is using VLSI technology successfully. However, the scaling limitation of the CMOS technology is the main disadvantage. This leads to a switch to an alternative approach. QCA is a nano technology that has emerged as an alternative to CMOS technology. An encoder is an electronic digital circuit that compares the inputs and outputs. An encoder is used in error correction and communication networks. At the nanoscale level, quantum-dot cellular automata (QCA) is an emerging computing technology beyond the current paradigm. QCA uses charge to encode binary values. Information is processed with the aid of the intercellular coupling mechanism and the QCA cell's configuration or magnetization. In comparison to CMOS devices, QCA might provide high device density, high integration, and high switching speed with extremely low energy dissipation. Up until now, QCA has been used to design and implement a large number of sequential and combinational logic circuits. In addition, a simple processor design strategy and complex arithmetic circuits demonstrate the QCA's efficiency as a viable nano computer. Inherent shift-register capacity in QCA, as opposed to conventional characteristics of logic circuits, results in cyclic 4-phase clocking in QCA devices. Circuits have been investigated in a number of studies. The key design element for storage is "memory-in-motion" and for computation, it is "logic-in-wire", in order to achieve high performance and device density. The issue of "layout = timing" in QCA design is illustrated by using variables like wire length, clock zone width, wasted space, and physical feedback.

These factors are taken into account when designing QCA circuits in an effective and dependable manner. Additionally, a number of design guidelines are put forth by the researchers to accomplish dependable and effective nano-scale digital circuit

design in QCA. In QCA, nano communication has drawn interest from a broad spectrum of researchers. Numerous nano communication architectures based on QCA have previously been documented by scholars. However, there are still many different domains in which to construct a QCA-based architecture for nano communication. This study investigates the design.

## 2. Related work

To build the QCA architectures for nano communication, several works have been proposed [1–28]. To maximize channel utilization, a multiplexer (MUX) and demultiplexer (DEMUX)-based router architecture was created in QCA, as proposed by Tirthji Maharaja Jagadguru [1]. The computational fidelity for channels composed of QCA devices in nanocomputing has been calculated by Van Loan [2]. An array of QCA cells is used as the noisy channel for the estimation. The random flaw in the array has been taken into account during measurement. A nano-router circuit employing DEMUX, PISO converter, and crossbar architecture is suggested by Hashemi and Navi [3] in contrast to the method presented by Das and De [4]. A detailed description is given of how well this nano-router routes information. A 4-bit data processor circuit was implemented in QCA by Sardinha et al. [5]. That data processor can be used for various multifunctional tasks, including sigmoid function creation and information preprocessing. Sayedsalehi et al. [6] presented a strong QCA architecture with serial communication capabilities. This design includes a SIPO converter, a PISO converter, a parity checker, and a hamming code generator. The turbo encoder design process and its implementation in QCA are illustrated by Yao et al. [7]. The use of reversible logic in QCA and its application in the design of nano communication architecture are discussed.

To further achieve security during nano communication, a number of cryptographic and steganographic architectures have been developed by Kamaraj et al. [8], Kianpour et al. [9], and Angizi et al. [10]. An enhanced arithmetic logic unit (ALU) design in QCA has been demonstrated in Zhang et al. [11]. The ALU is made up of a 1-bit full adder, a 2:1 MUX, and a two-input XOR gate. Additionally, it has been explained in Sheikhfaal et al. [12] how those I/O interfaces can be used to solve the issue with conventional tri-state gates. Sayedsalehi et al. [13], an effective. QCA SRAM cell implementation has been investigated. One 3-input MV, one 5-input MV, and a 2:1 MUX are included in this design. Accurate descriptions are also provided for energy efficiency and structural strength. A brand-new programmable QCA circuit is shown in Ahmad et al. [14]. Crossbar architecture is used in the design of the circuit. You can use this proposed circuit to build any type of Boolean logic. The architecture helps create and model area-efficient, stable, and consistent QCA circuits. A synchronous counter with a reliable and effective architecture is described in Kalogeiton et al. [15], Das and De [16,17]. Dflip-flip and edge-to-level converter circuits make up the design. A proposed efficient Fredkin gate with QCA implementation can be found in Das and De [18] and Chandra Das and De [20]. The Fredkin gate is then used to create an authenticator circuit that uses the user's password to identify the authorized user. Compared to previous QCA-Fredkin gates, this one has a lower cell count, latency, and device area. In order to investigate circuit stability, the

computational functionality under thermal randomization is also assessed. By Zhang et al. [21], the QCA design of a block cipher employing an electronic code book (ECB) is demonstrated, and QCA technology is used to implement it. A proposed encoder circuit with dual functionality as a decoder circuit is able to construct block ciphers.

### Polar code

A polar code is an error-correcting code. Polar code uses recursive cascading, which converts the physical medium into a virtual medium. The main advantage of the polar code is that there is less polarization. A polar code has 4 tuples (i.e., N, R, A, UA). Here, N represents the block length; R represents the code rate; A is the bit position; and UA is the fixed rate.

This polar code has K inputs and N outputs. This design explorer (8:4) In the polar encoder, the code rate is calculated by (K/N). The code rate of this proposed design is (1/2), which is the code rate. A polar code, often known as an error-correcting code, is a kind of linear block code that ranges from 23 to 25. Cascaded recursively over the short kernel code is employed in the creation of the polar code, which transforms the tangible medium into a virtual one by Azimi et al. [22] and Zhang et al. [23]. For a high quantity of repeat. The information bits are allocated to the most dependable virtual medium based on whether the virtual medium has low or high polarization during the simulation process. With polynomial dependence by Premananda et al. [24] on the gap to capacity, it was built to achieve the capacity of symmetric binary-input discrete memory-less channels (B-DMC). In formal terms, a polar code is defined as a 4-tuple (N, R, A, UA), in which N is the block length (i.e., the length of information bit travelled via the communication channel) by Teen et al. [25].

The code is represented by $R \in [0,1]$. A polar code, often known as error-correcting code, is a kind of linear block code that ranges from 23 to 25 by Dehbozorgi et al. [26] and Siddaiah et al. [27]. It cascaded recursively over the short kernel code.

## 3. Proposed work

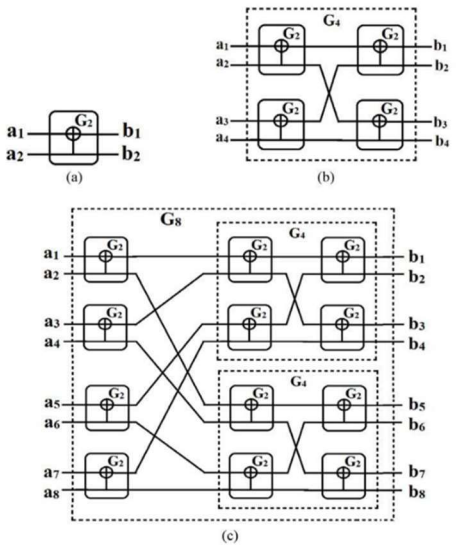


**Figure 1.** Block diagrams of **(a)** G2; **(b)** G4; **(c);** G8 [9].

The proposed polar encoder has 4 inputs and 8 outputs, which represents the (8:4) polar encoder. This encoder has G8 and 'a' be the input (ai = a1, a2, a3, a4, …, an) and the 'b' the output (bi = b1, b2, b3, b4, …, bn).

Actually, this encoder is given 8 inputs, of which 4 are frozen or fixed with values, and the remaining 4 are for giving inputs. So, here, a1, a2, a3, and a5 are frozen inputs, and a4, a6, a7, and a8 are given inputs, as shown in **Figure 1**.

## 3.1. Construction of G8

The G8 structure can be denoted by GN. To achieve the construction of G8, firstly, the construction of G2 is made. By using G2, the G4 is made, and lastly, by using G4, the construction of the G8 is made. One way to define it is by using a simple recursive rule. This section deals with the construction of G8. To attain the structure of G8, first the building of G2 has been completed. After that, G4 was constructed using the structure of G2. Lastly, employ G2 and G4, the building blocks of G8.

### 3.1.1. Design of G2

The outputs of G2 are two (b1, b2) and two inputs (a1, a2). **Figure 2** (where $\oplus$ indicates the XOR operation and the input-output mapping is displayed) and **Figure 3** demonstrate that the output b2 is equal to the input a2, and the output b1 is the XOR value of the inputs a1 and a2. Therefore, building the structure of G2 just requires a single XOR operation.

$b1 = a1 \oplus a2$

$b2 = a2$

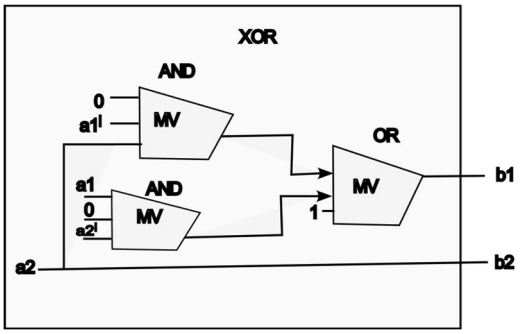


**Figure 2.** Design of G2.

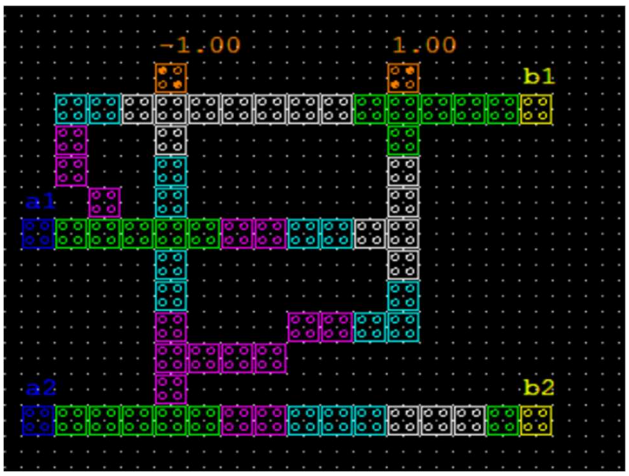


**Figure 3.** Implementation of G2 design.

### 3.1.2. Design of G4

The structure of G4 can be easily created by concatenating four copies of G2, as FIG-4 illustrates. It maps inputs (a1, a2, a3, and a4) to outputs (b1, b2, b3, and b4).

b1 = a1 ⊕ a2 ⊕ a3 ⊕ a4

b2 = a3 ⊕ a4

b3 = a2 ⊕ a4

b4 = a4

As said, one XOR operation is needed for the production of G2. As a result, to complete the QCA implementation of G2, corresponding to **Figure 3**, as illustrated in **Figure 4**, one QCA XOR circuit is sufficient. **Figure 5** displays the QCA layout.

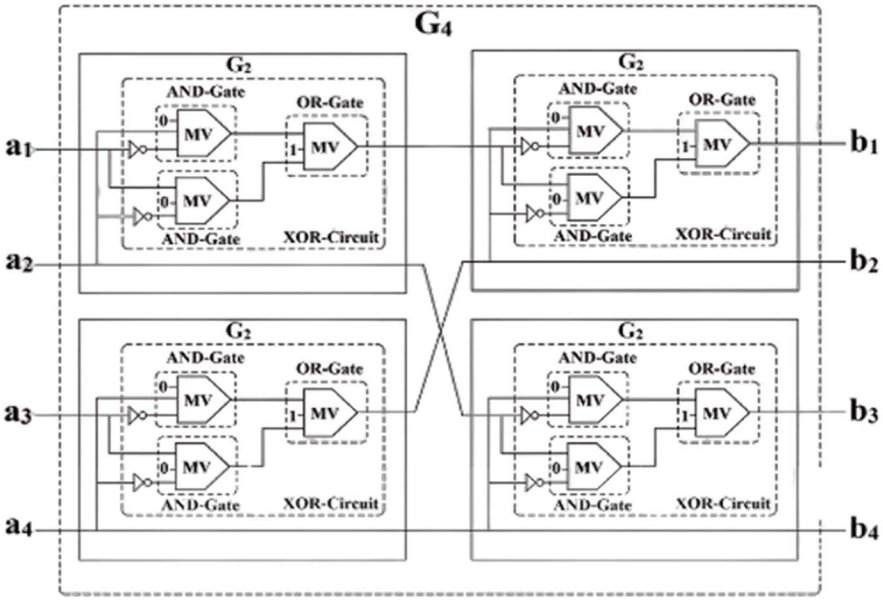


**Figure 4.** Design of G4 [9].

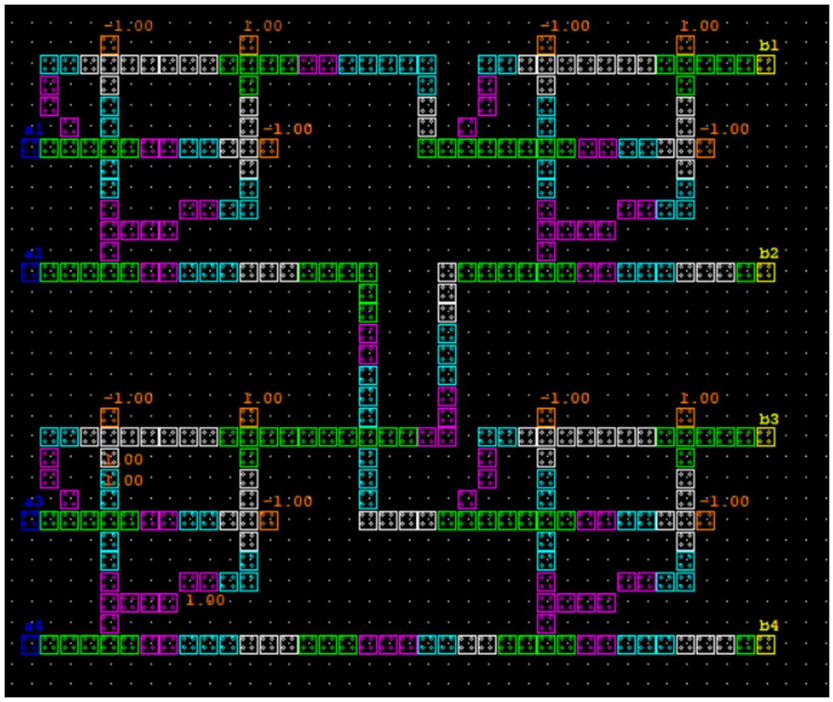


**Figure 5.** Implementation of G4.

### 3.1.3. Design of G8

The construction of G8 includes 4 blocks of G2 design and two blocks of G4 design, as shown in **Figure 6**. It has 8 inputs (a1, a2, a3, a4, a5, a6, a7, a8) and 8 outputs (b1, b2, b3, b4, b5, b6, b7, b8). **Figure 7** illustrates the implementation of G8.

$b1 = a1 \oplus a2 \oplus a3 \oplus a4 \oplus a5 \oplus a6 \oplus a7 \oplus a8$

$b2 = a5 \oplus a6 \oplus a7 \oplus a8$

$b3 = a3 \oplus a4 \oplus a7 \oplus a8$

$b4 = a7 \oplus a8$

$b5 = a2 \oplus a4 \oplus a6 \oplus a8$

$b6 = a6 \oplus a8$

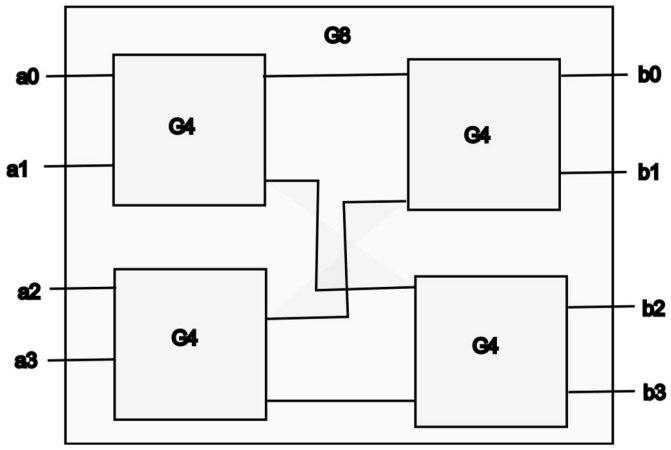


**Figure 6.** Design of G8.

**Figure 7.** Implementation of G8.

## 4. Experimental results

The design of G8 has no fixed cells and has (8:4). The G8 design has fixed polarized input cells, and 36 are fixed polarized QCA cells. Totally, the G8 design consists of 40 fixed polarized cells. The design is simulated using the QCA Designer program. The QCA implementation of G2 is accomplished by using the wire crossing technique suggested by Abedi et al. To implement single-layer wire crossing in QCA, the four-phase clocking approach of QCA Designer, with its benefit of two clock zones, is helpful. It is demonstrated by Angizi et al. [10] that clock zones 0 and 2 can be combined to build a wire cross. The same thing Plotting the simulation result corresponds to **Figure 6**. The legitimate outputs are displayed in a rectangle box. **Figure 6** illustrates that when a1 = 0 and a2 = 0, the resulting values are b1 = 0 and b2 = 0. The results are b1 = 1 and b2 = 1 for a1 = 0 and a2 = 1. In a similar vein, every output appears in accordance with every input. This outcome validates the theoretical values, demonstrating the accuracy of the design. Another interesting finding from **Figure 6** is that the output appears during the second clock cycle, meaning that it is delayed by one clock cycle after the input. In the work of Vangala et al. [28], algorithms for polar encoders were proposed, and in the work of Babar et al. [29], polar encoder types were designed using QCA.

Two G2 copies are needed for the production of G4. As seen in **Figure 4,** the QCA implementation of G4 will thus be done utilizing two QCA circuits in G2. **Figure 8** displays the QCA layout from **Figure 9**. The design is simulated using QCA.

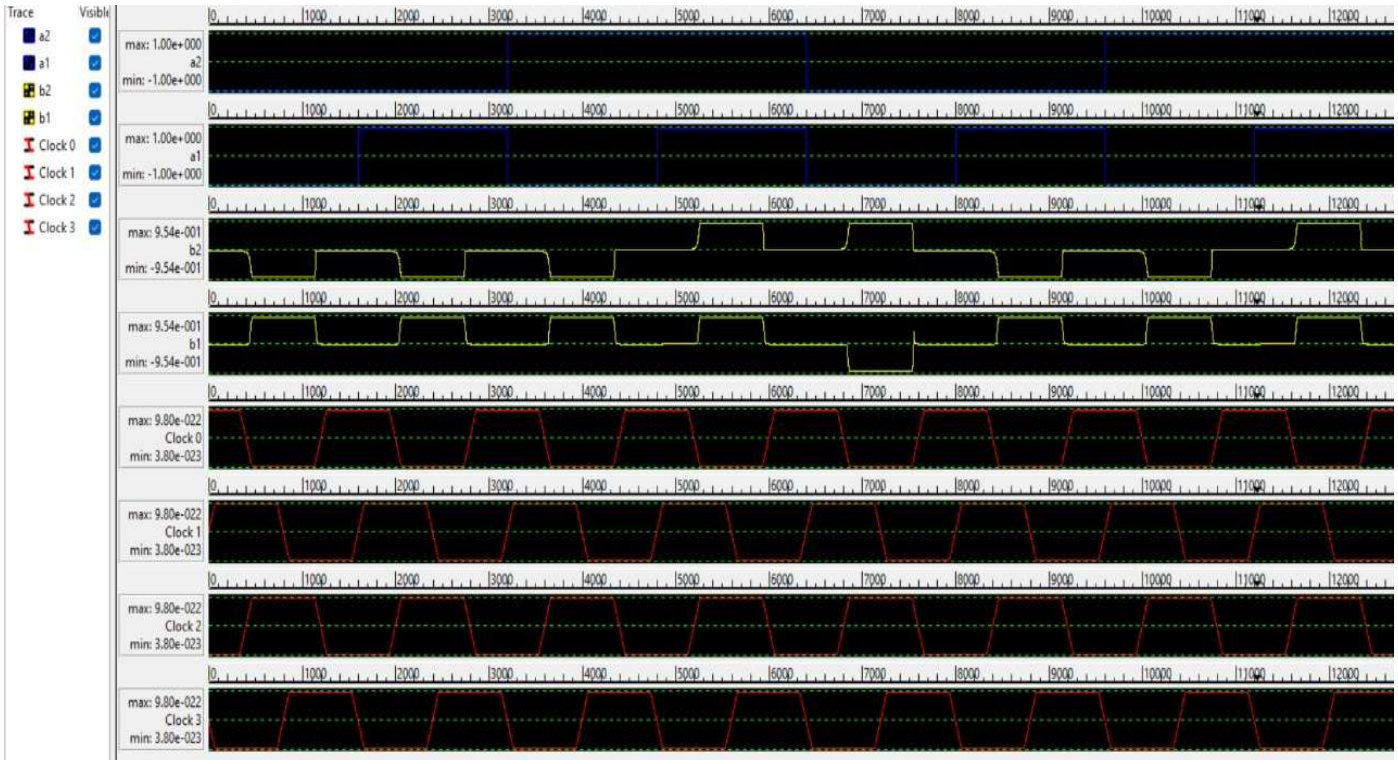


**Figure 8.** Output waveform of G2.


**Figure 9.** Output waveform of G4.

**Figure 10** displays the simulation results that correlate. The legitimate outputs are displayed in a rectangle box. As can be seen from **Figure 6c**, the output is b1 = 0, b2 = 0, b3 = 0, and b4 = 0 when a1 = 0, a2 = 0, a3 = 0, and a4 = 0. The result is b1 = 1, b2 = 1, b3 = 1, and b4 = 1 when a1 = 0, a2 = 0, a3 = 0, and a4 = 1. In a similar manner, the input combinations determine how each output appears. Therefore, the outcome validates the theory.

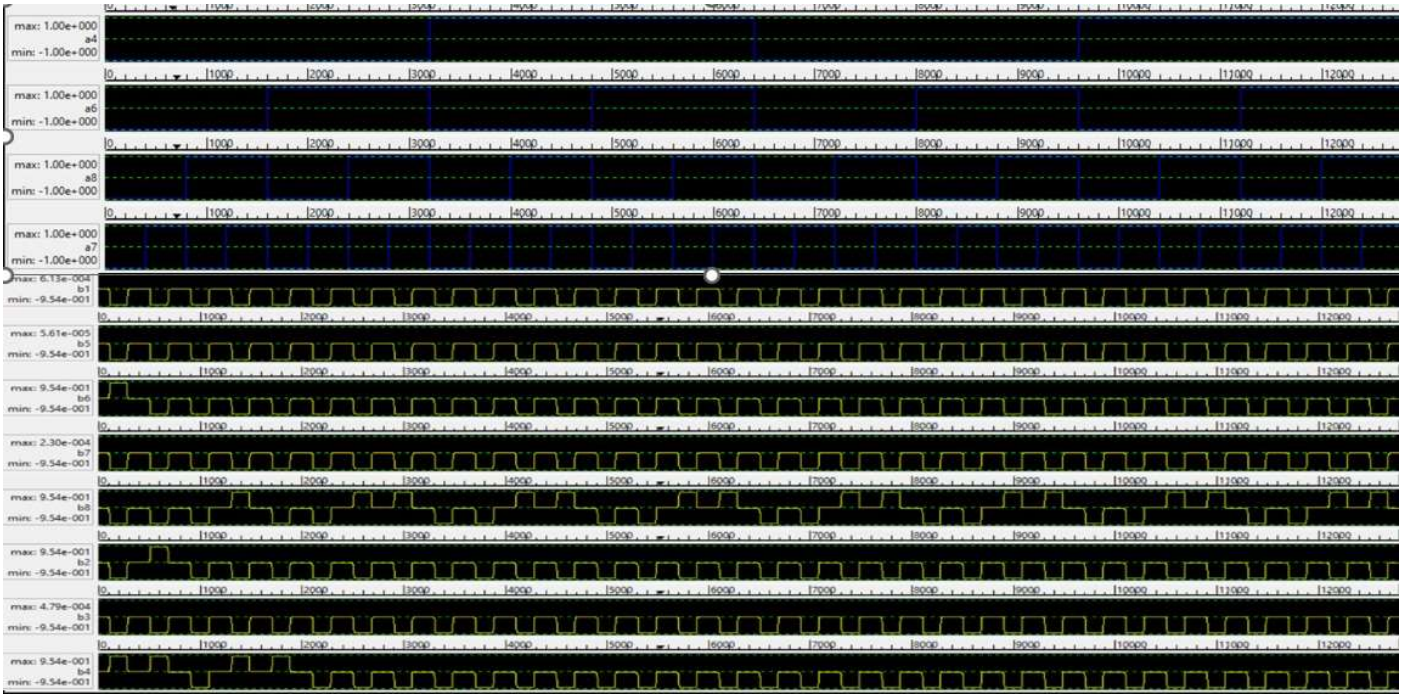

**Figure 10.** Output waveform of G8.

**Figure 10** describes the simulation result corresponding to the (8:4) polar encoder as it is displayed in **Figure 7**. The valid results are displayed using the rectangle box. According to **Figure 8b**, the outputs for data bits a4 = 0, a6 = 0, a7 = 0, and a8 = 0 are b1 = 0, b2 = 0, b3 = 0, b4 = 0, b5 = 0, b6 = 0, b7 = 0, and b8 = 0. This is because all of the frozen bits (i.e., a1, a2, a3, and a5) are set to zero. The outputs are b1 = 1, b2 = 1, b3 = 1, b4 = 1, b5 = 1, b6 = 1, b7 = 1, and b8 = 1 for the data bits a4 = 0, a6 = 0, a7 = 0, and a8 = 1. In the same vein, every output appears in accordance with the data bits. As a result, the outcome validates the theoretical values, proving the design's accuracy. **Figure 10** also shows that the output is six clock cycles behind the input; that is, the legitimate output appears at the eleventh clock cycle. The performance comparison of the proposed schematic with the existing one is shown in **Table 1**. From **Table 1**, it is clear that the proposed schematic outperforms the existing [11] in terms of cell count, area, and complexity.

**Table 1.** Comparison of proposed vs. existing [11].

| Parameters | Existing [11] | Proposed |
|---|---|---|
| Cells count | 2275 | 1188 |
| Area ($um^2$) | 5.4320 | 1.915 |
| Complexity | More | Less |

## 5. Conclusion

In this work, the (8:4) polar encoder circuit is designed for nano scale communication and implemented using the QCA platform. Here, the bottom-up approach is used to reduce the complexity, and it is performed by a single layer, which reduces the complexity even during the physical device fabrication. The code proposed has a lower cell count and less area compared to other encoders that use turbo and cc codes [11]. The top-down method has been taken into consideration to simplify the design. The implementation is done in a single layer, which helps lower complexity while fabricating devices. The communication procedure with the Polar encoder is depicted in the communication architecture. It is helpful to give fault-free design with the stuck-at-fault effect analysis. The suggested test vectors are robust enough to enable 100% fault coverage. The simulation result validates the encoder circuit's design accuracy. Device area and circuit latency demonstrate the quicker speed at which the Polar Encoder circuit may work at the nanoscale. The low dissipated energy of the proposed Polar encoder circuit is verified through energy dissipation estimation.

**Author contributions:** Conceptualization, SKV and CR; methodology, CR, BM and ANT; software, CR; validation, SKV and RS; formal analysis, CR; investigation, SKV, RS; resources, CR; data curation, CR; writing—original draft preparation, SKV and CR; writing—review and editing, SKV and CR; visualization, CR, BM and ANT; supervision, SKV; project administration, CR, BM and ANT. All authors have read and agreed to the published version of the manuscript.

**Conflict of interest:** The authors declare no conflict of interest.

## References


1. Tirthji Maharaja Jagadguru SSBK. Vedic Mathematics. Motilal Banarsidas; 1986.
2. Van Loan CF. The ubiquitous Kronecker product. Journal of computational and applied mathematics. 2000; 123(1-2): 85-100. doi: 10.1016/S0377-0427(00)00393-9
3. Hashemi S, Navi K. New robust QCA D flip flop and memory structures. Microelectronics Journal. 2012; 43(12): 929-940. doi: 10.1016/j.mejo.2012.10.007
4. Das S, De D. Nanocommunication using QCA: A data path selector cum router for efficient channel utilization. In: Proceedings of the 2012 International Conference on Radar, Communication and Computing (ICRCC); 21-22 December 2012; Tiruvannamalai, India. pp. 43-47. doi: 10.1109/icrcc.2012.6450545
5. Sardinha LHB, Costa AMM, Neto OPV, et al. NanoRouter: A Quantum-dot Cellular Automata Design. IEEE Journal on Selected Areas in Communications. 2013; 31(12): 825-834. doi: 10.1109/jsac.2013.sup2.12130015
6. Sayedsalehi, Samira, Moaiyeri MH, and Navi K. Design of efficient and testable n-input logic gates in quantum-dot cellular automata. Journal of Computational and Theoretical Nanoscience. 2013; 10(10): 2347-2353.
7. Yao F, Zein-Sabatto MS, Shao G, et al. Nanosensor Data Processor in Quantum-Dot Cellular Automata. Journal of Nanotechnology. 2014; 2014: 1-14. doi: 10.1155/2014/259869
8. Kamaraj A, Abinaya, Ramya S. Design of router using Reversible Logic in Quantum Cellular Automata. In: Proceedings of the 2014 International Conference on Communication and Network Technologies; 18-19 December 2014; Sivakasi, India. pp. 249-253. doi: 10.1109/cnt.2014.7062764
9. Kianpour M, Sabbaghi-Nadooshan R, Navi K. A novel design of 8-bit adder/subtractor by quantum-dot cellular automata. Journal of Computer and System Sciences. 2014; 80(7): 1404-1414. doi: 10.1016/j.jcss.2014.04.012
10. Angizi S, Moaiyeri MH, Farrokhi S, et al. Designing quantum-dot cellular automata counters with energy consumption analysis. Microprocessors and Microsystems. 2015; 39(7): 512-520. doi: 10.1016/j.micpro.2015.07.011
11. Zhang M, Cai L, Yang X, et al. Design and Simulation of Turbo Encoder in Quantum-Dot Cellular Automata. IEEE Transactions on Nanotechnology. 2015; 14(5): 820-828. doi: 10.1109/tnano.2015.2449663
12. Sheikhfaal S, Angizi S, Sarmadi S, et al. Designing efficient QCA logical circuits with power dissipation analysis. Microelectronics Journal. 2015; 46(6): 462-471. doi: 10.1016/j.mejo.2015.03.016
13. Sayedsalehi S, Rahimi Azghadi M, Angizi S, et al. Restoring and non-restoring array divider designs in Quantum-dot Cellular Automata. Information Sciences. 2015; 311: 86-101. doi: 10.1016/j.ins.2015.03.030
14. Ahmad F, Bhat GM, Khademolhosseini H, et al. Towards single layer quantum-dot cellular automata adders based on explicit interaction of cells. Journal of Computational Science. 2016; 16: 8-15. doi: 10.1016/j.jocs.2016.02.005
15. Kalogeiton VS, Papadopoulos DP, Liolis O, et al. Programmable Crossbar Quantum-Dot Cellular Automata Circuits. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2017; 36(8): 1367-1380. doi: 10.1109/tcad.2016.2618869
16. Das JC, De D. User Authentication Based on Quantum-Dot Cellular Automata Using Reversible Logic for Secure Nanocommunication. Arabian Journal for Science and Engineering. 2015; 41(3): 773-784. doi: 10.1007/s13369-015-1870-z
17. Das JC, De D. Circuit switching with Quantum-Dot Cellular Automata. Nano Communication Networks. 2017; 14: 16-28. doi: 10.1016/j.nancom.2017.09.002
18. Das JC, De D. Nanocommunication network design using QCA reversible crossbar switch. Nano Communication Networks. 2017; 13: 20-33. doi: 10.1016/j.nancom.2017.06.003
19. Abutaleb MM. Robust and efficient quantum-dot cellular automata synchronous counters. Microelectronics Journal. 2017; 61: 6-14. doi: 10.1016/j.mejo.2016.12.013
20. Chandra Das J, De D. QCA based secure nanocommunication block cipher design based on electronic code book. Malaysian Journal of Computer Science. 2018; 31(2): 130-142. doi: 10.22452/mjcs.vol31no2.3
21. Zhang Y, Xie G, Cheng X, et al. The Implementation of I/O Interface in Quantum-dot Cellular Automata. Optik. 2018; 166: 177-188. doi: 10.1016/j.ijleo.2018.04.020
22. Azimi S, Angizi S, Moaiyeri MH. Efficient and Robust SRAM Cell Design Based on Quantum-Dot Cellular Automata. ECS Journal of Solid State Science and Technology. 2018; 7(3): Q38-Q45. doi: 10.1149/2.0281803jss
23. Zhang Y, Xie G, Han J. Serial concatenated convolutional code encoder in quantum-dot cellular automata. Nano Communication Networks. 2019; 22: 100268. doi: 10.1016/j.nancom.2019.100268

24. Premananda BSC, Skanda, Srivatsa B. Area and energy efficient QCA based decoder. In: Proceedings of the 2021 6th International Conference on Communication and Electronics Systems (ICCES); 8-10 July 2021; Coimbatre, India. pp. 7-12. doi: 10.1109/ICCES51350.2021.9489011
25. Teen YPA, Subha M, Shabeer SH, et al. Programmable multiplier circuit designed for quantum-dot cellular automata devices. Materials Today: Proceedings. 2021; 37: 1295-1300. doi: 10.1016/j.matpr.2020.06.464
26. Dehbozorgi L, Sabbaghi-Nadooshan R, Kashaninia A. Realization of processing-in-memory using binary and ternary quantum-dot cellular automata. The Journal of Supercomputing. 2021; 78(5): 6846-6874. doi: 10.1007/s11227-021-04152-1
27. Siddaiah, Premananda B, Megha P, Nagavika K. Compact and Energy Efficient QCA Based Hamming Encoder for Error Detection and Correction. Advances in Electrical and Electronic Engineering. 2023; 21(2): 120-126.
28. Vangala H, Hong Y, Viterbo E. Efficient Algorithms for Systematic Polar Encoding. IEEE Communications Letters. 2016; 20(1): 17-20. doi: 10.1109/lcomm.2015.2497220
29. Babar Z, Kaykac Egilmez ZB, Xiang L, et al. Polar Codes and Their Quantum-Domain Counterparts. IEEE Communications Surveys & Tutorials. 2020; 22(1): 123-155. doi: 10.1109/comst.2019.2937923