International Journal of Basic and Applied Sciences, 13 (2) (2024) 18-24

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS

Research paper

QCA-BASED design of reversible hamming code encoding, decoding and correcting circuits

Bahijja Yahaya Galadima [1] *, Garba Shehu Musa Galadanci [1]

[1] Department of Physics, Bayero University, Kano, Nigeria
**Corresponding author E-mail: bygaladima.elt@gmail.com*

Abstract

This work proposes a model of a low-power hamming code generator (HCG) circuit for single-bit data with 11 cells, based on a reversible Feynman gate. A circuit is implemented for error detector parity bit (EDP) in hamming coding for message signals of three bits with 21 cells. To ensure optimum functionality, the suggested circuits and their theoretical values are verified using the QCA Designer simulator version 2.0.3, and the energy dissipation of the circuits is estimated using the QCA Designer-E. The results for the simulation show that the proposed circuits improve the occupied area by 82.5% and the cell counts by 66.6% for the HCG circuits. Additionally, the EDP circuit improves the occupied area by 55% and the cell counts by 25%. The error-correcting circuit with coplanar crossover achieves 30.5% in the occupied area and 16.7% cell reduction. The result also proves that energy dissipation by the proposed circuit increases as the cell count increases. As such, the cell count, area, and energy dissipation performance are all greatly improved by the proposed Hamming coding circuit. This shows that QCA-based reversible circuits are more energy-efficient, quicker, and denser than other types, which makes them a good option for usage in upcoming nanoscale integrated circuit applications.

Keywords: *Error Detector Parity Bit (EDP); Feynman Gate; Hamming Code Generator (HCG); Qcadesigner-E; Reversible Logic.*

1. Introduction

The quest for technology that is both energy- and space-efficient has been motivated by the need for scalable transistor sizes and lower power consumption in electronic circuits. Even with ballistic transistors, power dissipation at the contacts is unavoidable in today's CMOS silicon technology [1], [2]. The size of CMOS devices has gradually decreased by very high-scale integration, but this technique has already hit its limit [3], [4]. In addition to increased leakage and power consumption, many short-channel effects (SCEs) must be addressed throughout the ongoing shrinking of CMOS devices. CMOS substitutes are essential for maintaining circuit shrinkage and enhancing microprocessor performance [5 - 7]. A highly promising alternative that tackles problems with size, power, and speed is quantum-dot cellular automata (QCA). QCA exploits its cells' magnetization or charge configuration to encode binary values, and intercellular coupling techniques are used to process the data [8], [9]. Transistor-free Quantum-dot Cellular Automata (QCA) technology is distinguished by its elevated device density, rapid clocking rate, and incredibly low power consumption [10], [11], [6]. Information flows and other calculations in QCA are dependent on the mutual interaction of electrons within the QCA cells that affect their magnetization states or charge configurations, which in turn contain binary information [12], [13].

To overcome the problems with size, speed, and power dissipation, Landauer [14] proposed using quantum cellular automata and reversible computing. Reversible logic, which is extremely energy efficient, releasing no heat energy for each information byte, is another design strategy [15]. In an irreversible circuit, the energy loss for every bit of information translation is quantitatively equal to kTln 2 J, where T is the absolute computing temperature and k is Boltzmann's constant [16]. Bennett (1973) demonstrated that reversible circuits are necessary to avoid the energy dissipation of kTln2 joules in a circuit. Reversible gates are required for reversible computing, and they can be implemented using QCA [6].

Errors occurs during data transmission from source to destination in digital communication due to noise and other environmental disturbances [17], [18]. When a single bit changes while being transferred from the sender to the destination an error occurs. To reduce these problems, all digital circuits have incorporated error detection and repair circuits [18]. A circuit for a Hamming code encoder and decoder that utilizes reversible logic gates to detect and fix errors when they happen is introduced in [17]. The simulation results are obtained by employing the Xilinx ISE 14.4 design suite. The simulation results provide a new way to incorporate reversible logic gates into traditional Hamming code circuits to reduce power consumption. With 27 gates, a 39 quantum cost, 18 garbage outputs, and a 1ns delay. In [19], the Hamming Code converter, decoder, and corrector unit was introduced in an improved reversible version.A Toffoli Netlist was used in the implementation of the designs with 18 gates, and a quantum cost of 18. The result showed that the suggested designs could end up being crucial parts of future quantum architectures. A circuit based on QCA that generates Hamming codes, detects errors, and correct messages that is extremely reliable, reversible, and area-efficient is presented in [15]. Feynman reversible logic and coplanar crossings with 180° clock zones are used to develop the circuit. Comparatively small number of cells, 2.75 periods of clock demand, 0.99 quantum cost, and zero garbage values for both the generator and corrector circuits, and excellent area efficiency of 29.5% are all attained. A different work, [20] showcases the creation and application of a new 3-to-8 decoder using basic cells, as well as a 3-input XOR gate These components are used to build an electronic circuit that integrates the QCA-based methodology and is used to implement Hamming codes. The proposed hamming communication network has shown a significant improvement in occupied area (10.14%) and a significant reduction in consumed cells (54.27%).

The work of [21] implements a VHDL-based channel coding and decoding system that has a channel coding component and a channel decoding component in the system design. It utilizes the Hamming Code for encoding and decoding. According to the results, in high-bit rate scenarios, the system's capacity for error correction can be enhanced, and its bit error rate can be significantly decreased during transmission. However, it is noted that the circuitry for the hamming error control has been designed differently using reversible logic in QCA. The most important factor in circuit design is energy consumption and occupied area, so it is imperative to create an energy-efficient design with low power, high speed, and minimal occupied area.

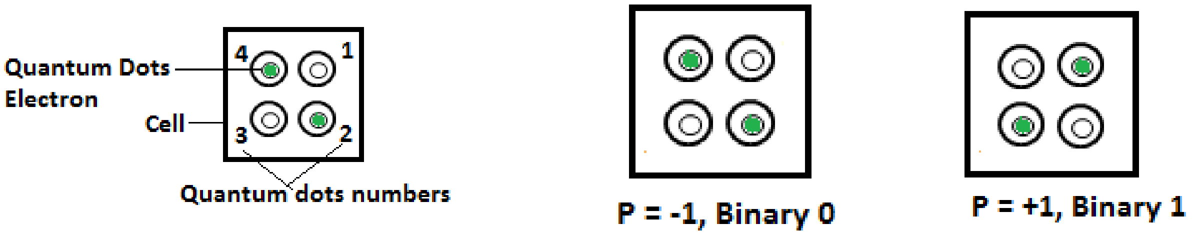
This work aims at designing a hamming encoder code circuit for single-bit data. Additionally, a circuit for detecting errors in Hamming coding is suggested for message signals with three bits. To demonstrate the viability, a one-bit error-correcting circuit employing reversible QCA logic has been successfully designed. QCADesigner version 2.0.3 validates the designs and QCADesigner-E detects energy dissipation, Cell count, delay, and occupied area are the performance measures taken into consideration.

1.1. Reversible logic

A reversible logic gate is a logic device with the same quantity of inputs and outputs in addition to a distinct mapping between them [22], [23]. An equal number of inputs and outputs are generated by a reversible logic gate. A distinct pattern of output vectors is produced for each configuration of input vectors [24]. This feature eliminates information loss, which lowers power consumption. Feedback loops and fan-outs are not possible with reversible logic. A reversible logic circuit has minimal garbage outputs, minimum reversible gates, and minimum input constants. Conversely, Quantum-dot Cellular Automata (QCA) is a cell array where quantum dots make up each cell, which can alternatively be considered sites in the corners of the square cell [25]. The dots constitute the majority of the charge. Additionally, two mobile electrons in the cell can tunnel between the dots. Potential barriers between cells prevent electrons from tunneling out of a cell. In the corners of the cell, two free electrons are always positioned diagonally away from one another due to Coulombic repulsion. Figure 1(a) displays a QCA cell with four quantum dots and a distinct number (site) labeled on each. The cell's polarization (P) is calculated using equation (1) [26].

$$P = \frac{(\rho_1+\rho_3)-(\rho_2+\rho_4)}{\rho_1+\rho_2+\rho_3+\rho_4} \qquad (1)$$

Equation (2) provides the expectation value of the number operator on site (dot) for the ground state eigen function (ρi), where I denote the numbers 1, 2, 3, and 4 of the quantum dot, as illustrated in Figure 1(a).



(a) (b) (c)

Fig. 1: QCA Cell (a) Representation (b) with Polarization P = "-1" (c) with Polarization P = "+1". [25].

$$\rho_i = \langle \Psi_o | \hat{\eta}_i | \Psi_o \rangle \qquad (2)$$

The cell's ground state is denoted by $|\Psi_o\rangle$ and is defined by equation (3) as;

$$|\Psi_o\rangle = \sum_j \Psi_j^0 |\phi_j\rangle \qquad (3)$$

Where $\Psi_j^o$is the basis vector's coefficient, as obtained by Hamiltonian diagonalization in direct form and $|\phi_j\rangle$ is the jth vector. However, the diagonal placement of electrons determines the cell's polarization. When electrons are arranged as they are in Figure 1(b), cell polarization P = -1 and they are represented by equation (1) as binary 0 (Logic 0). Similar to this, the cell polarization P = +1 and its encoding are binary 1 (Logic 1), given the placement of the electrons in Figure 1(c). The coulombic attraction between the cells in the QCA array is what causes the data flow. The cell-to-cell response can be obtained by computing the Schrodinger equation for two-particle. When a cell's polarization mirrors that of its neighbor, cell i in the two-cell system i and j, is said to be a driver. For a single cell i, the two-state models in the N-cell system are calculated using the Hamiltonian in equation (4) [27].

$$\hat{H} = \begin{bmatrix} -\frac{1}{2}P_j E_{i,j}^k & -\gamma_j \\ -\gamma_j & \frac{1}{2}P_j E_{i,j}^k \end{bmatrix} \qquad (4)$$

Equation (5) defines the Kink energy for cells i and j, where γ_i is its tunneling energy and $E_{i,j}^{k}$is the Kink energy. The polarization of cell j is given by P_j. Kink energy can be used to characterize the coulombic contact between two cellular entities, E_kink. The variation in electrostatic energies between two cells with the same polarization and two with opposing polarization is known as kink energy [28].

$$E_{kink}^{i,j} = E_{opposite}^{i,j} - E_{same}^{i,j} \tag{5}$$

Where $E_{opposite}^{i,j}$ is the amount of energy within cells i and j with opposing polarization, and $E_{same}^{i,j}$ is the energy between cells i and j with similar polarization. The state energy is determined by evaluating the electrostatic energy of two cells. The electrostatic energy across cells i and j is presented by (6).

$$E^{i,j} = \frac{1}{4\pi\varepsilon_o\varepsilon_r}\sum_{r=1}^{4}\sum_{m=1}^{4}\frac{q_n^i q_m^j}{|r_n^i - r_m^j|} \tag{6}$$

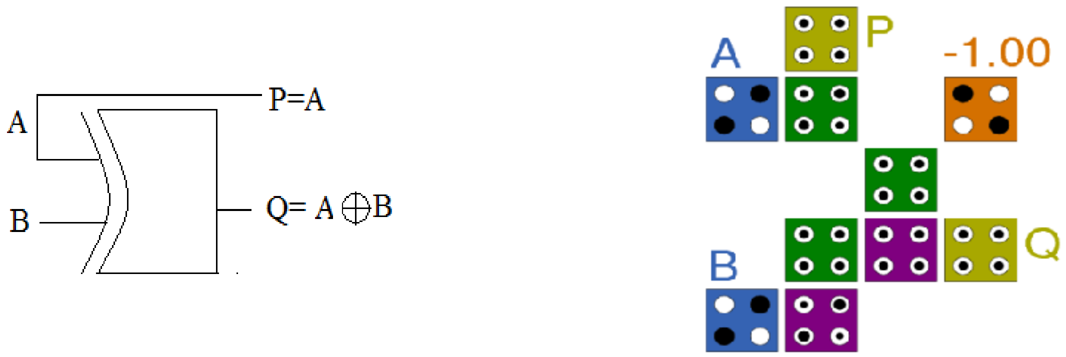
Where, ε_r is the relative permittivity of a substance and ε_o is the permittivity of free space. q_i^n is the charge in dot n of cell i, q_m^j is the charge in dot m of cell j, r_n^i is the location of the nth dot in cell i, r_m^j is the location of the mth dot in cell j, and the distance between the nth dot in cell i and the mth dot in cell j is denoted by $|r_n^i - r_m^j|$

2. Methodology

The Hamming coding technique is used in this section to generate, detect, and correct one-bit data. The Feynman gate is used to calculate the required parity bit number, determine the message signal's parity bit location, calculate the parity bit, and generate a message signal with the correct parity and data bit arrangement. Based on the developed Hamming code, the circuits for the error bit position detector and the bit correction circuit are further explained. The quantity of parity bits needed for a specific data length in order to produce the Hamming code is calculated using Equation (7).

$$2^P \geq D + P + 1 \tag{7}$$

Where P is the number of parity bits and D is the number of data bits.
The circuit that produces parity bits makes use of the Feynman reversible gate proposed in [29], which has two inputs along with two outputs. The first input is mirrored in one output, and the XOR (exclusive OR) value of the two inputs is shown in the other output. Figures 2a and b display the Feynman gate's QCA configuration and block diagram.



(a) **(b)**

Fig. 2: (a) Proposed Feynman Gate Logic Diagram (b) QCA Layout.

2.1. Encoder circuit

The encoder circuit accepts a data word and performs XOR operations to process it. As a result, the parity generator generates the required parity bits. A single data bit and two parity bits are needed to generate a Hamming (3, 1) code, according to the HCG relation in equation (7). The bit locations and their configuration are shown in Table 1. Parity bit 1 (P1) checks bit positions 1 and 3, and parity bit 2 (P2) checks bit positions 2 and 3. P1 and P2 are precisely the same as D1 in HCG. The block diagram and QCA layout of the Hamming (3, 1) code generator circuit are displayed in Figures 3a and b.

Table 1: Hamming Code Bit Positions

Bit Position	1	2	3
Parity/ Data bit	P1	P2	D1

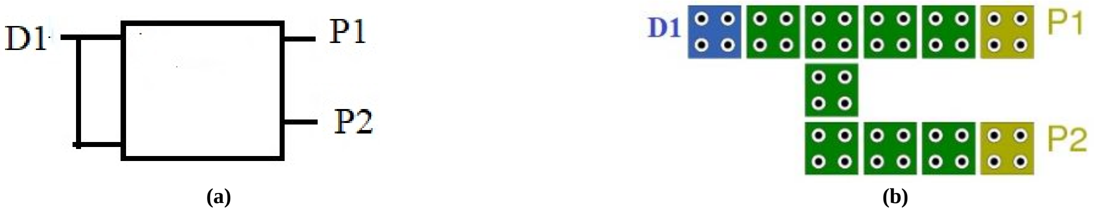


Fig. 3: (a) Proposed Reversible Hamming (3, 1) Code Generator Circuit Logic Diagram (b) QCA Layout.

2.2. Decoder circuit

The code word is fed into the decoder circuit as input. After that, check bits are generated by the checker bit generator to confirm the parity bits. Through the decoder circuit, check bits are computed at the receiver side to confirm whether the bit stream has reached its destination safely. To generate EDP1, the code detector circuit at the receiving end requires P1 and D1, and to generate EDP2, it requires P2 and D1. The relationships needed to create the three-bit message signals EDP1 and EDP2 are shown by equations (8) and (9).

$$EDP1 = P1 \oplus\ D1 \qquad (8)$$

$$EDP2 = P2\ \oplus D1 \qquad (9)$$

The block diagram for the three-bit message error-detecting circuit is displayed in Figures 4a and b.

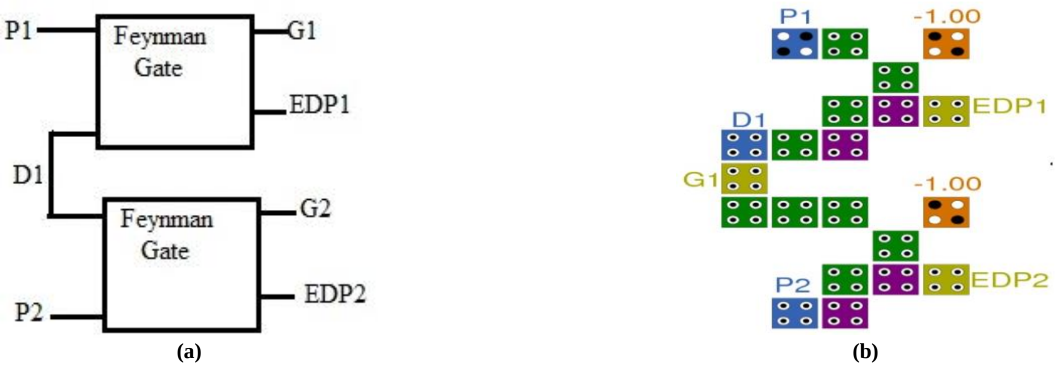


Fig. 4: (a) Reversible Hamming (3, 1) Code Error Detector Circuit Logic Diagram (b) QCA Layout.

The error detector circuit is designed with the inclusion of two Feynman reversible gates. These gates accept one data bit and one parity bit as inputs, and they produce one garbage output and one EDP bit as outputs for each gate. The location of the error bit is found using data from EDP1 and EDP2. The corrector circuit then finds this bit and reverses its value. Consequently, the correct message bits (CM1, CM2, and CM3) are produced at the corrector circuit's output. It has a decoder responsible for decoding the EDPs to ascertain the erroneous position. Three 2-to-1 multiplexers and three inverters are used to flip each bit in the message signal. Each multiplexer's select line is connected to the decoder's outputs (O1, O2, and O3). One message bit and its inverse equivalent make up each multiplexer's input. Now, depending on the decoder's output, the multiplexer selects either the genuine message bit or the flispped bit. The erroneous bit's position will be indicated by a value of 1 in the decoder's output. Since the connected decoder's output is zero, the message's flipped bit will be chosen by the multiplexer connected to that output bit, while the original message bits will be chosen by the remaining multiplexers. Figures 5a and b depict the basic circuit block diagram and QCA layout for the corrector circuit.

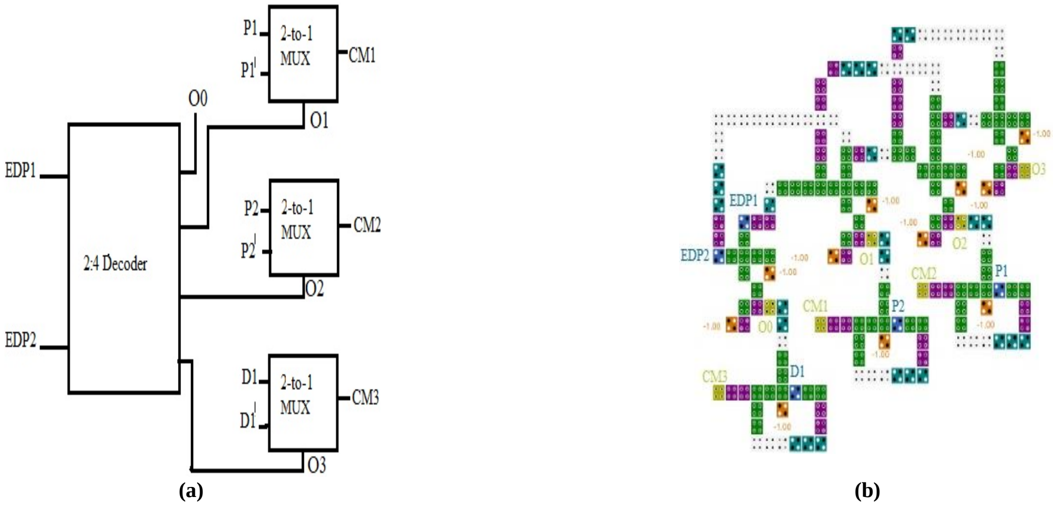


Fig. 5: (A) Reversible Hamming (3, 1) Code Error Corrector Circuit Logic Diagram (B) QCA Layout.

3. Results and discussion

This section presents and discusses the simulation results obtained by the Hamming (3, 1) code. It also addresses the hamming code error detector circuits involving three-bit messages, along with the corrector circuit involving three-bit messages. The proposed circuit is designed and validated by the QCA –Designer tool version 2.0.3[27]. The simulation's default settings are: QCA cell size = 18 nm, diameter of quantum dots = 5 nm, number of samples = 50,000, convergence tolerance = 0.001, radius of effect = 65 nm, relative permittivity = 12.9, $clocklow\ =\ 3.8E^{-23}J$, $clockhigh$ $9.8E^{-22}J$, clock amplitude factor = 2.000, layer separation = 11.5 nm, and maximum iterations per sample = 100.

3.1. Simulation result of the hamming encoder

The simulation result of the proposed reversible Hamming (3, 1) code generator circuit is shown in Figure 6. From the result, both P1 and P2 are equivalent to D1. This structure has a complexity of 11 cells, which is 67% less than the structure of [15]. Moreover, the proposed circuit's occupied area is 0.007µm2, leading to an improvement of 82% compared to the [15] structure.

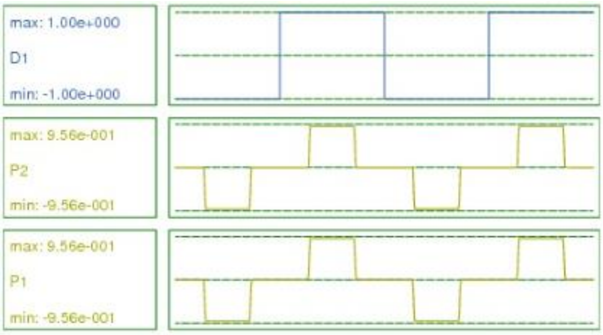


Fig. 6: Input/Output Waveforms of the Reversible Hamming (3, 1) Code Generator Circuit.

3.2. Simulation result of the hamming decoder

The input and output waveforms corresponding to the (3,1) hamming code-based error detector circuit layout are depicted in Figure 7. It is observed from the simulation findings that the moment the inputs are D1 = 0, P1 = 0, and P2 = 0, the output becomes EDP = 0 and EDP = 0. Likewise, if the inputs are D1 = 0, P1 = 0, and P2 = 1, the output becomes EDP = 0, and the entire process persists for other values of the input data. These results are in line with equations (8) and (9). The structure has a complexity of 21 cells covering an area of 0.018 m2, which is improved by 25% and 55%, respectively, compared to the [15] structure.

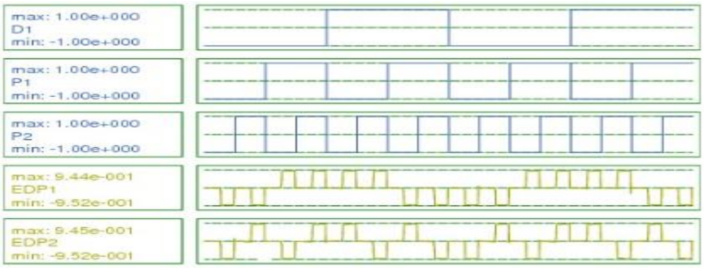


Fig. 7: Input/output waveforms of the reversible Hamming (3,1) code error detector circuit

Regarding the complete process of error correction, figure 8 shows the input and output waveforms for the (3,1) Hamming code corrector circuit. The corrector circuit has a complexity of 199 cells, occupying an area of 0.25µm^{2}. This is more improved result by 16.7% and 30.5% compared to the one obtained by [15].

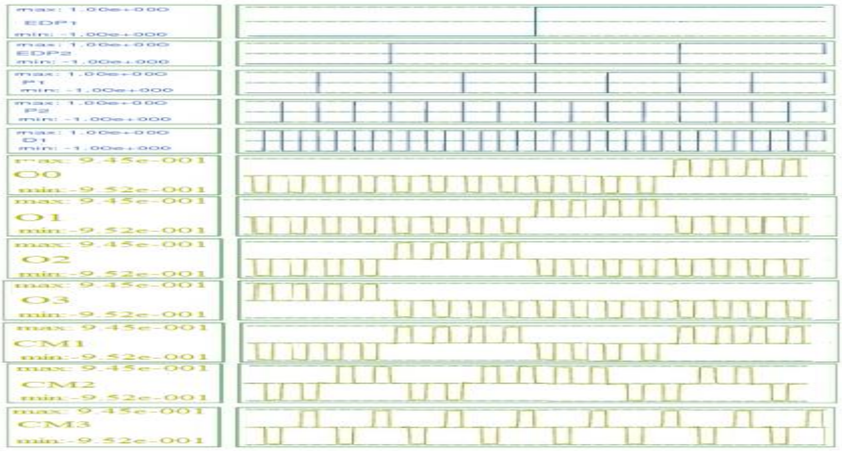


Fig. 8: Input/output waveforms of the reversible Hamming (3,1) code error corrector circuit

In Table 2, the summary of the main findings of this work in comparison with previous work according to cell count, occupied area, and latency is presented.

Table 2: Summary of the Main Findings of this Work in Comparison with Previous Work

QCA circuit	Cell count	Total area (μm^2)	Latency (ns)
Hamming (3,1) code generator [15]	33	0.04	1
Hamming (3,1) code generator [This work]	11	0.007	0.25
Hamming(3,1) code error detector [15]	28	0.04	0.5
Hamming (3,1) code error detector [This work]	21	0.018	0.5
Hamming (3,1) code error corrector circuit [15]	239	0.36	2.75
Hamming (3,1) code error corrector circuit [This work]	199	0.25	2.75

3.3. Energy dissipation

QCA Designer-E version 2.2, an updated version of QCA Designer, is applied to calculate the proposed circuits' energy dissipation. For each of the suggested circuits, the average and total energy are computed, presented, and compared with other work in Table 5. According to the table, the dissipation of energy increases with an increase in the number of cells.

Table 5: Energy Dissipation Comparison of this Work and Other Work

QCA circuits	Total energy dissipation (eV)	Average energy dissipation (eV)
Hamming (3,1) code generator [15]	6.72×10^{-3}	6.11×10^{-4}
Hamming (3,1) code generator [This work]	3.75×10^{-3}	3.41×10^{-4}
Hamming (3,1) code error detector [15]	2.36×10^{-2}	2.15×10^{-3}
Hamming (3,1) code error detector [This work]	1.68×10^{-2}	1.52×10^{-3}
Hamming (3,1) code error corrector circuit [15]	8.64×10^{-2}	7.85×10^{-3}
Hamming (3,1) code error corrector circuit [This work]	5.62×10^{-2}	5.11×10^{-3}

4. Conclusion

We developed circuits for generating error-detecting codes for single-bit data. It also covers designing a circuit for an error detector for three-bit messages. We employ the coplanar crossover approach and 2 × 2 Feynman reversible gates to develop these circuits. Here, we use the Feynman gate with the fewest cells to minimize the area needed for the circuits. These revealed that the circuits are more viable because just one layer is needed to create them, courtesy of the coplanar crossover. The suggested circuit for the Hamming code shows notable improvements in terms of area utilization, cell count efficiency, and clock delay performance. The designed HCG circuits have achieved a minimum reduction of 82.5% in area utilization, and a 66.6% decrease in the number of cells required. An impressive 55% reduction in occupied area, with a 25% decrease in the number of cells for the EDP circuit compared to the most recently proposed circuit. Additionally, the circuits show excellent energy efficiency, having an exceptionally low energy dissipation of 3.75×10^{-3} eV with a maximum energy dissipation of 5.62×10^{-2} eV for the suggested designs. Subsequent investigations may delve into more refinements concerninsg latency, energy consumption, and fault tolerance to augment the feasibility and expandability of QCA-based circuits for advanced computing needs. Researchers can proceed to develop and create new approaches to address the constantly rising needs for contemporary communication networks by expanding upon the basis of this study.

References

[1] J. Yang, G. Li, and H. Liu, "Edge direct tunnelling current in nano-scale MOSFET with high-K dielectrics," vol. 1, pp. 30–33, 2008, https://doi.org/10.1108/13565360810846626.

[2] I. Gassoumi, L. Touil, B. Ouni, and A. Mtibaa, "An Ultra-Low Power Parity Generator Circuit Based on QCA Technology," *J. Electr. Comput. Eng.*, vol. 2019, 2019, https://doi.org/10.1155/2019/1675169.

[3] J. Huang, G. Xie, R. Kuang, F. Deng, and Y. Zhang, "Microprocessors and Microsystems QCA-based Hamming code circuit for nano communication network," *Microprocess. Microsyst.*, vol. 84, no. November 2020, p. 104237, 2021, https://doi.org/10.1016/j.micpro.2021.104237.

[4] K. Kalpana, K. Sivakami, N. Revathi, S. M. Deepa, and V. V. Teresa, "Efficient Nano-Scale Design of TIEO Based Reversible Logic Toffoli Gate Priority Encoder in Quantum-Dot Cellular Automata," *E3S Web Conf.*, vol. 472, 2024, https://doi.org/10.1051/e3sconf/202447203014.

[5] 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, https://doi.org/10.1016/j.egypro.2017.05.170.

[6] J. C. Das and D. De, "Nanocommunication Network Design Using QCA Reversible Crossbar Switch," *Nano Commun. Netw.*, 2017, https://doi.org/10.1016/j.nancom.2017.06.003.

[7] V. K. Sharma, "Optimal design for digital comparator using QCA nanotechnology with energy estimation," *Int. J. Numer. Model. Electron. Networks, Devices Fields*, vol. 34, no. 2, pp. 2–11, 2021, doi: 10.1002/jnm.2822. https://doi.org/10.1002/jnm.2822.

[8] A. Norouzi and S. R. Heikalabad, "Design of reversible parity generator and checker for the implementation of nano-communication systems in quantum-dot cellular automata," *Photonic Netw. Commun.*, vol. 38, no. 2, pp. 231–243, 2019, https://doi.org/10.1007/s11107-019-00850-2.

[9] L. Lu, W. Liu, M. O'Neill, and E. E. Swartzlander, "QCA Systolic array design," *IEEE Trans. Comput.*, vol. 62, no. 3, pp. 548–560, 2013, https://doi.org/10.1109/TC.2011.234.

[10] D. Tougaw and M. Khatun, "A scalable signal distribution network for quantum-dot cellular automata," *IEEE Trans. Nanotechnol.*, vol. 12, no. 2, pp. 215–224, 2013, https://doi.org/10.1109/TNANO.2013.2243162.

[11] J. C. Das and D. De, "Novel low power reversible binary incrementer design using quantum-dot cellular automata," *Microprocess. Microsyst.*, vol. 42, pp. 10–23, 2016, https://doi.org/10.1016/j.micpro.2015.12.004.

[12] A. Roohi, H. Khademolhosseini, S. Sayedsalehi, and K. Navi, "A symmetric quantum-dot cellular automata design for 5-input majority gate," *J. Comput. Electron.*, vol. 13, no. 3, pp. 701–708, 2014, https://doi.org/10.1007/s10825-014-0589-5.

[13] G. Singh, R. K. Sarin, and B. Raj, "A novel robust exclusive-OR function implementation in QCA nanotechnology with energy dissipation analysis," *J. Comput. Electron.*, vol. 15, no. 2, pp. 455–465, 2016, https://doi.org/10.1007/s10825-016-0804-7.

[14] Rolf Landauer, "Irreversibility and Heat Generation in the Computing Process," *IBM J. Res. Dev.*, no. July, pp. 183–191, 1961. https://doi.org/10.1147/rd.53.0183.

[15] A. Kaity and S. Singh, "An area-efficient, robust, and reversible QCA-based Hamming code generator, error detector, and corrector: design and performance estimation," *J. Comput. Electron.*, vol. 20, no. 6, pp. 2622–2647, 2021, https://doi.org/10.1007/s10825-021-01802-8.

[16] C. H. Bennett, "Logical Reversibility of Computation.," *IBM J. Res. Dev.*, vol. 17, no. 6, pp. 525–532, 1973, https://doi.org/10.1147/rd.176.0525.

[17] D. K. Kavitha, "ISSN NO : 2236-6124 Design and analysis of Hamming Code Encoding , Decoding and Correcting Circuits usingReversible Logic Page No : 36 ISSN NO : 2236-6124 Page No : 37," vol. 7, no. 2236, pp. 36–41, 2018.

[18] M. A. Muneeb and S. Namratha, "Verilog Implementation of Hamming Code for Error Control Coding," vol. 10, no. 1, pp. 69–73, 2022.

[19] D. Sengupta, M. Sultana, and A. Chaudhuri, "Hamming code converter using reversible toffoli netlist," *Int. J. Recent Technol. Eng.*, vol. 8, no. 3, pp. 1814–1818, 2019, https://doi.org/10.35940/ijrte.C4617.098319.

[20] H. Xie, Y. Qi, and F. Q. A. Alyousuf, "Designing an ultra-efficient Hamming code generator circuit for a secure nano-telecommunication network," *Microprocess. Microsyst.*, vol. 103, no. May, p. 104961, 2023, https://doi.org/10.1016/j.micpro.2023.104961.

[21] J. Shan, J. Zhoe, "'Design of encoding and decoding of Hamming code based on VHDL," pp. 241–244, 2020, https://doi.org/10.1109/ICCSMT51754.2020.00056.

[22] N. Abdessaied and R. Drechsler, *Reversible and Quantum Circuits*. 2016. https://doi.org/10.1007/978-3-319-31937-7 .

[23] M. Soeken, R. Wille, O. Keszocze, D. Michael Miller, and R. Drechsler, "Embedding of large boolean functions for reversible logic," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 12, no. 4, 2015, https://doi.org/10.1145/2786982.

[24] P. Biswas, N. Gupta, and N. Patidar, "Basic Reversible Logic Gates and It's Qca Implementation," *J. Eng. Res. Appl. www.ijera.com*, vol. 4, no. 6, pp. 12–16, 2014, [Online]. Available: www.ijera.com

[25] U. Mehta and V. Dhare, "Quantum-dot cellular automata (QCA): A survey," *arXiv*, no. November, 2017.

[26] C. S. Lent and P. D. Tougaw, "Lines of interacting quantum-dot cells: A binary wire," *J. Appl. Phys.*, vol. 74, no. 10, pp. 6227–6233, 1993, https://doi.org/10.1063/1.355196.

[27] K. Walus, T. J. Dysart, G. A. Jullien, and R. A. Budiman, "QCADesigner: A Rapid Design and Simulation Tool for Quantum-Dot Cellular Automata," *IEEE Trans. Nanotechnol.*, vol. 3, no. 1 SPEC. ISS., pp. 26–31, 2004, https://doi.org/10.1109/TNANO.2003.820815.

[28] G. L. Snider *et al.*, "Quantum-dot cellular automata: Review and recent experiments (invited)," *J. Appl. Phys.*, vol. 85, no. 8 II A, pp. 4283–4285, 1999, https://doi.org/10.1063/1.370344.

[29] B. Y. Galadima, G. S. M. Galadanci, S. M. Gana, A. Tijjani, and M. Ibrahim, "QCA Based Design of Reversible Parity Generator and Parity Checker Circuits for Telecommunication," vol. 5, no. 2, pp. 331–343, 2023.