International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August, 2019

Adiabatic Hamming Code Generator-Checker using 4-Dot-2-Electron Quantum Dot Cellular Automata

Ayan Chaudhuri, Diganta Sengupta, Chitrita Chaudhuri

Abstract*: Technological advancements have witnessed rapid regression of Moore's Law within the past few years. With rising demand for higher clocking speeds, CMOS has already started exhibiting threshold limitations. Reversible Logic has emerged as a suitable alternative with near zero heat dissipation attribute. Quantum Dot Cellular Automata (QCA) has adopted the concept of reversibility and emerged as a primitive tool for quantum architecture deigns with clocking near Terra-Hertz range. A plethora of quantum architectures based on QCA cells have been proposed till date. With rise of research on digital designs based on QCA, multiple literary proposals exist which realize digital designs incorporating QCA cells. This communication proposes a Hamming Code Generator-Checker architecture design using 4-dot-2-electron QCA cells. We employ an existing QCA based XOR gate literary proposal for designing the proposed architecture. Peer comparison with literary counterparts has proven our design to fare better with a gain of 60.6% in area.

Keywords: Quantum Dot Cellular Automata, Hamming Code, Reversible Logic, Quantum Architecture, 4-dot-2-electron, QCA, Moore's Law.

I. INTRODUCTION

Quantum Dot Cellular Automata (QCA) has emerged as a viable alternative to CMOS in digital circuit designs for the last decade with escalating research in the search of alternatives for CMOS. CMOS has reached near-threshold limitations due to primarily two factors – its heat dissipation issue and the clocking speed of GHz. QCA poses as a sustainable counterpart in quantum architecture proposals for its alignment to concepts of reversibility [1] [2] and clock speeds in THz. Literary proposals witness numerous implementations for digital designs using QCA [3], some are arithmetic and some are logical; both resembling the digital domain. One of the reasons for the popularity of QCA can be attributed to the ease of transformation of the digital designs into valid QCA realizations. The primitive gate for QCA is the Majority Voter, and this gate can be used to realize the primitive gates of Boolean algebra [4] [5]; viz. AND, OR and NOT gate. Since these three can be achieved using QCA, hence any digital architecture is possible on the QCA platform since digital designs are profoundly Boolean dependent. Although multiple architectures have been proposed on arithmetic and logical designs, few have observed the cryptographic domain also. This communication explores the viability of QCA cells in designing cryptographic architectures, explicitly the Hamming Code Generator and the Checker. Boolean XOR operation forms the foundation for cryptographic designs; hence, we explore numerous XOR proposals using QCA cells in literature [6] [7] and concentrate on a state-of-the-art design [8] for designing the novel Hamming Code Generator-Checker in this work. The design exhibits better results in peer comparison and reflects optimized performance when implemented using [9].

The rest of the paper is organized as follows. The next section presents the related work followed by the section on our proposal. The subsequent section provides the design analysis followed by conclusion.

II. RELATED WORK

Quantum Dot Cellular Automata (QCA) [10] [11] [12] [13] has gained mileage in the past two decades due to its alignment to the attributes of reversible logic. Zero heat dissipating architectures have been proposed using the QCA technology [5] [14] [15]. To establish viability, experimental results for realizing the basic Boolean gates, viz. AND, OR and NOT gate, have been provided in [15]. The Boolean XOR gate forms an important parameter in multiple digital circuits, primarily cryptographic circuits. Although XOR gate can been realized by the primitive Boolean gates [17], research has been tremendous in search for standalone XOR implementations [6] [7] [8], since realization of XOR using AND, OR and NOT consumes huge resources. The XOR gate is of typical interest in this work because it forms the basis for the design of Hamming Code Generator-Checker. Fundamentally research on QCA has concentrated on these parameters – a. physical implementation of QCA cells, arithmetic architectures using QCA cells, logical realizations using QCA cells, and realization of digital circuits using the technology.

Revised Manuscript Received on October 30, 2019.
* Correspondence Author
Ayan Chaudhuri, Research Scholar, Department of Computer Science and Engineering, Jadavpur University, Kolkata, (West Bengal), India,
Diganta Sengupta*, Dept. of Computer Science and Engineering, Techno International Batanagar, Kolkata, West Bengal 700141, India,
Chitrita Chudhuri, Department of Computer Science and Engineering, Jadavpur University, Kolkata, (West Bengal), India,

We concentrate on the realization of the digital counterpart for Hamming Code Generator/Checker using QCA since it finds wide acceptance in cryptographic systems. Although a few implementations have been proposed till date [18] [19] [20], our proposal fares better, as has been discussed in the analysis section in this communication. Since, XOR forms the basis of Hamming Code operation; hence Figure 1, Figure 2 and Figure 3 present the existing proposals in literature. We have shortlisted Figure 3 for our realization since it excels than the other two designs for XOR formulation.

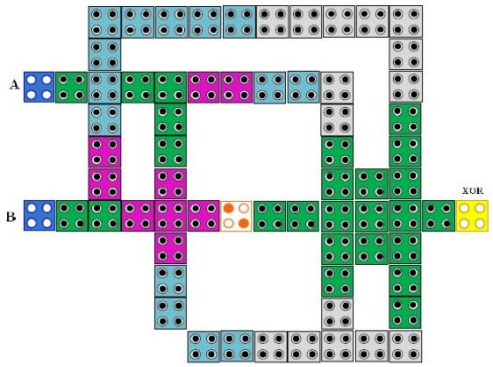

Fig. 1.XOR Gate proposal in [6]

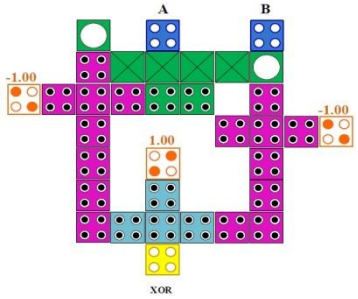

Fig. 2.XOR Gate proposal in [7]

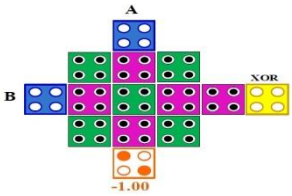

Fig. 3.XOR Gate proposal in [8]

It is interesting to note that Figure 2 implements QCA cross-overs [21]. In this communication, we concentrate on single layer QCA designs, hence consideration of XOR proposal in Figure 2 is eliminated thereof.

Multiple Hamming Code circuit realizations have been witnessed in literature. NAND/NOR gate realizations using QCA cells have been proposed in [18]. These gates form the basis for the XOR gate that has been implemented for realizing the Hamming Code Generator-Checker. The complete design has been divided into $3 \times 3, 3 \times 2$ and 2×2 zones and the Quantum Dot cells are then routed among them. The proposal in [20] is similar to the proposal in [18] using QCA NAND equivalents.

The next section details the Hamming Code Generator/Checker logic

III. HAMMING CODE GENERATOR/CHECKER

Hamming Codes form the member of Forward Error Correction (FEC) codes for single bit errors. The logic of error detection and correction are presented in the Equation 1 through Equation 12. Illustration 1 provides a working example for the equations.

Let $Message\ P = p_7p_6p_5p_4p_3p_2p_1p_0$ (1)

$Coded\ Message\ C = p_7p_6p_5p_4h_8p_3p_2p_1h_4p_0h_2h_1$ (2)

Where $h_1, h_2, h_4, and\ h_8$ are the hamming code bits for P.

Let, X denote the coded message C. Therefore,

$X = \sum_{i=0}^{11} x_{i+1}2^i$

$\Rightarrow D = x_{12}x_{11}x_{10}x_9x_8x_7x_6x_5x_4x_3x_2x_1$ (3)

Where

$x_1 = h_1\ ;\ x_2 = h_2\ ;\ x_3 = p_0\ ;\ x_4 = h_4\ ;\ x_5 = p_1$

$x_6 = p_2\ ;\ x_7 = p_3\ ;\ x_8 = h_8\ ;\ x_9 = p_4\ ;\ x_{10} = p_5$

$x_{11} = p_6\ ;\ x_{12} = p_7$

The Hamming codes are generated according to the following equations.

$h_1 = x_1 \oplus x_3 \oplus x_5 \oplus x_7 \oplus x_9 \oplus x_{11}$ (4)

$h_2 = x_2 \oplus x_3 \oplus x_6 \oplus x_7 \oplus x_{10} \oplus x_{11}$ (5)

$h_4 = x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_{12}$ (6)

$h_8 = x_8 \oplus x_9 \oplus x_{10} \oplus x_{11} \oplus x_{12}$ (7)

Where, $\oplus$ denotes the Binary XOR operation.

The Hamming Code checker equations are generated according to Equation 8 through Equation 11.

$c_1 = x_1 \oplus x_3 \oplus x_5 \oplus x_7 \oplus x_9 \oplus x_{11} \oplus h_1$ (8)

$c_2 = x_2 \oplus x_3 \oplus x_6 \oplus x_7 \oplus x_{10} \oplus x_{11} \oplus h_2$ (9)

$c_4 = x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_{12} \oplus h_4$ (10)

$c_8 = x_8 \oplus x_9 \oplus x_{10} \oplus x_{11} \oplus x_{12} \oplus h_8$ (11)

Let

$H = h_8h_4h_2h_1\ and\ C' = c_8c_4c_2c_1$

Therefore, the Error message $'E'$ is obtained by bit-wise XORing of $H\ and\ C'$.

Therefore,

$E = \sum_{i=0}^{3} e_i 2^i\ ;where\ e_i = h_j \oplus c_j \forall j \in \{8,4,2,1\}$ (12)

The Error message E provides the bit position of error in the transmitted message D. Illustration 1 presents the process in details.

Illustration 1:

Let $M = 10100111$

Therefore,

$C = 1010h_8011h_41h_2h_1$ and

$D = 1010h_8011h_41h_2h_1$

Where

$h_1 = 0, h_2 = 1, h_4 = 1,$ and $h_8 = 0$ respectively.

Therefore the transmitted message D is

$D = 101000111110$

Now, if the message bit

p_1 gets corrupted at bit position x_6 of D, then $D = 101000\underline{\mathbf{0}}11110$. The bold and underlined bit represents the corruption.

Therefore, at the receiver end,
$c_1 = 0, c_2 = 0, c_4 = 0, and\ c_8 = 0$ respectively.
Therefore error E equals
$E = 0101$, where
$e_4 = h_8 \oplus c_8 = 0$
$e_3 = h_4 \oplus c_4 = 1$
$e_2 = h_2 \oplus c_2 = 1$
$e_1 = h_1 \oplus c_1 = 0$
Hence, bit position for corruption is given by E, i.e. 6^{th} position (x_6) of the message has been corrupted in transmitted message D. Therefore inversion of that bit (x_6) fetches the rectified original message at the receiving end. Hence, this coding scheme is termed as the Forward Error Correcting (FEC) scheme.

IV. HAMMING CODE GENERATOR/CHECKER USING QCA CELLS

This section discusses the QCA implementation for Equation (1) through (12). It can be observed that the equations reflect XOR operation only. Hence we concentrate the design provided in Figure 2 to implement the equations as the stated design has been perceived to fare better than the counterparts.

The clocking sequence for the XOR gate of [8] is provided in Table I.

Table- I: Clocking sequence of XOR implementation of [8]

		$Clock_x$			
	$Clock_x$	$Clock_{x+1}$	$Clock_x$		
$Clock_x$	$Clock_{x+1}$	$Clock_x$	$Clock_{x+1}$	$Clock_{x+1}$	$Clock_{x+1}$
	$Clock_x$	$Clock_{x+1}$	$Clock_x$		
		$Clock_x$			

Figure 4 through Figure 7 present the QCA realization for Equation (4) through (7). The designs have been implemented using QCA Designer [9] using Bistable Approximation as the Simulation Engine. The design parameters are as follows:

Layer properties:
- Cell Width : 18 nm
- Cell Height : 18 nm
- Dot Diameter : 5 nm

Simulation Engine Parameters
- No. of samples : 12800
- Radius of effect : 65 nm
- Relative Permittivity : 12.9
- Clock High : $9.8e^{-022}$
- Clock Low : $3.8e^{-023}$
- Clock Amplitude Factor : 2
- Layer Separation : 11.5
- Max. Iterations/sample : 100

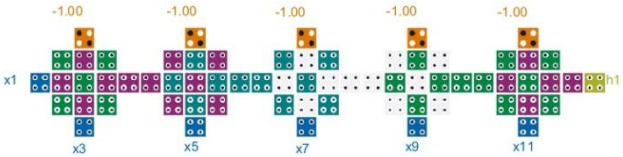

Fig. 4.QCA implementation for Equation (4)

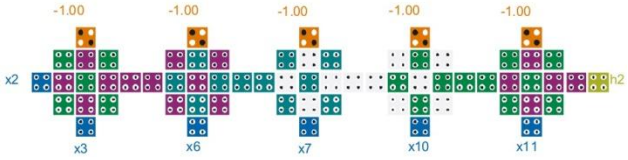

Fig. 5.QCA implementation for Equation (5)

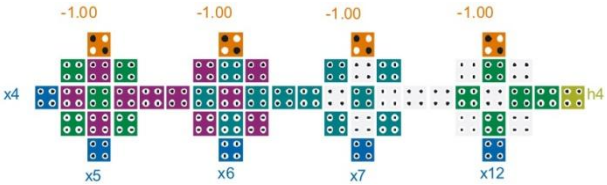

Fig. 6.QCA implementation for Equation (6)

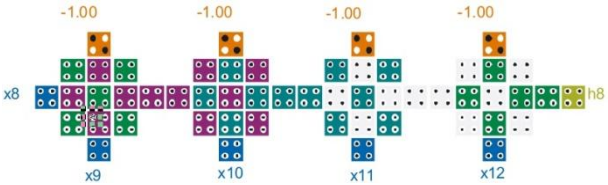

Fig. 7.QCA implementation for Equation (7)

Table II through Table VI present the clock sequence for the designs presented in Figure 4.

Table- II: Clocking sequence of XOR gate - 1 in Figure 4

		$Clock_0$			
	$Clock_0$	$Clock_1$	$Clock_0$		
$Clock_0$	$Clock_1$	$Clock_0$	$Clock_1$	$Clock_1$	$Clock_1$
	$Clock_0$	$Clock_1$	$Clock_0$		
		$Clock_0$			

Table- III: Clocking sequence of XOR gate - 2 in Figure 4

		$Clock_1$			
	$Clock_1$	$Clock_2$	$Clock_1$		
$Clock_1$	$Clock_2$	$Clock_1$	$Clock_2$	$Clock_2$	$Clock_2$
	$Clock_1$	$Clock_2$	$Clock_1$		
		$Clock_1$			

Table- IV: Clocking sequence of XOR gate - 3 in Figure 4

		$Clock_2$			
	$Clock_2$	$Clock_3$	$Clock_2$		
$Clock_2$	$Clock_3$	$Clock_2$	$Clock_3$	$Clock_3$	$Clock_3$
	$Clock_2$	$Clock_3$	$Clock_2$		
		$Clock_2$			

Table- V: Clocking sequence of XOR gate - 4 in Figure 4

		$Clock_3$			
	$Clock_3$	$Clock_4$	$Clock_3$		
$Clock_3$	$Clock_4$	$Clock_3$	$Clock_4$	$Clock_4$	$Clock_4$
	$Clock_3$	$Clock_4$	$Clock_3$		
		$Clock_3$			

Table- VI: Clocking sequence of XOR gate - 5 in Figure 4

		$Clock_4$			
	$Clock_4$	$Clock_0$	$Clock_4$		
$Clock_4$	$Clock_0$	$Clock_4$	$Clock_0$	$Clock_0$	$Clock_0$
	$Clock_4$	$Clock_0$	$Clock_4$		
		$Clock_4$			

Table I reflects that a single XOR operation requires a latency of 0.5. The complete realization of Equation 4 observed a latency of 1.25 as can be seen in Table II through Table V, since the output cloak for one XOR is mapped with the input for another XOR gate.

For Figure 5, Table II through Tables VI provides the clocking sequences whereas for Figure 6 and Figure 7, Table II through Table V provides the clocking sequences. Therefore, it can be claimed that the complete Hamming Generator can be executed in 1.25 latencies as that is the max of all the latencies in Figure 4 through Figure 7. Figure 8 through Figure 11 provide the QCA designs for the Hamming Code checker part (Equation 8 through Equation 11).

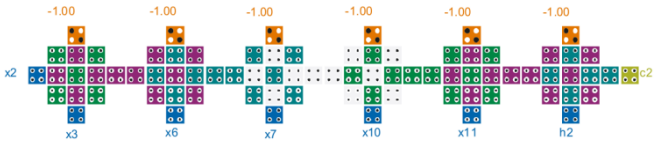

Fig. 8.QCA implementation for Equation (8)

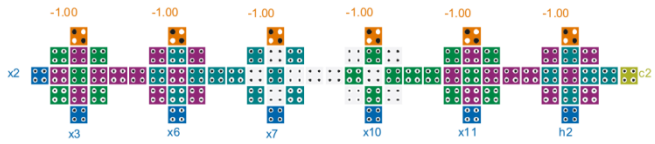

Fig. 9.QCA implementation for Equation (9)

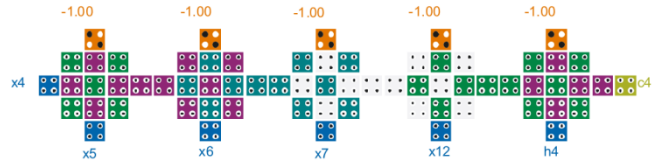

Fig. 10. QCA implementation for Equation (10)

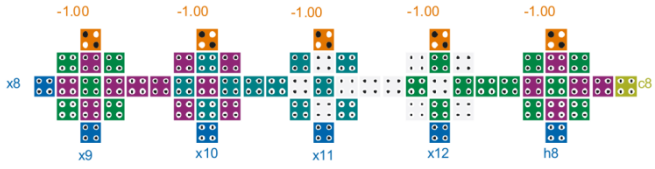

Fig. 11. QCA implementation for Equation (11)

It can be observed from Figure 8 through Figure 11 that the QCA implementations require an additional 0.25 latency to complete the checking operation with respect to their generator counterparts. Figure 12 presents the QCA design for the Error Signal 'E' given by Equation 12.

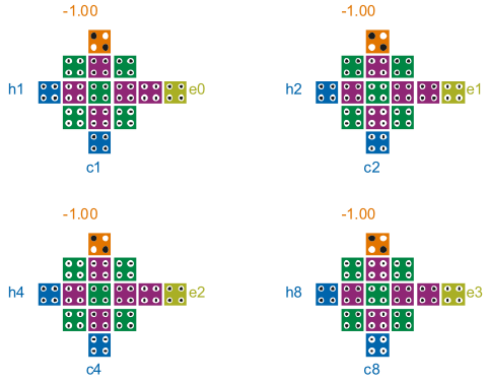

Fig. 12. QCA implementation for Equation (12 – Error 'E')

The four outputs of Figure 12 present the four bits of equation 12 representing the bit number in Equation 3 which has been corrupted during transmission.

V. DESIGN ANALYSIS

This section presents the design analysis of our designs. Table VII provides the design analysis according to the QCA structures for the Equations provided in Equation 4 through Equation 11. The analyses have been done with respect to design area and latency, the primary two factors in deciding architectures using QCA cells.

Table- VII: Design Analysis

Design for	Area (nm^2)	Latency
h_1	21384	1.25
h_2	21384	1.25
h_4	17172	1
h_8	17172	1
c_1	25596	1.5
c_2	25596	1.5
c_4	21384	1.25
c_8	21384	1.25
Error 'E'	4536	0.5
Total	175608	---

Table VII reflects that the maximum area for the comprehensive design for Hamming Code Generator/Checker using QCA cells is 175608 nm^2 and the maximum latency for generator is 1.25 clocks and that of checker is 1.5 clocks.

Since, the single XOR gate in Figure 4a of [20] requires 33 QCA cells in contrast to 13 cells in the design of XOR gate in [8], hence we can conclude that our proposal claims a gain of 60.6% in area over the proposal in [20]. A single XOR Gate in [20] consumes one latency, hence, the modular chip design in [20] will consume 5 to 6 latencies depending upon Equation 4 through Equation 12. Hence, our proposal gains by 75% - 79.19% in latency over the proposal in [20].

VI. CONCLUSION

This communication presents the realization of Hamming Code Generator/Checker using 4-Dot-2-Electron Quantum Dot Cellular Automata (QCA) Cells. A state-of-the-art XOR Gate using QCA cells has been utilized to generate the architectures. The architectures fare better than a literary counterpart in terms of area as well as latency.

REFERENCES

1. R Landauer, "Irreversibility and Heat Generation in the Computing Process," IBM Journal of Research and Development, vol. 5, no. 3, pp. 183 - 191, July 1961.
2. C H Bennett, "Logical Reversibility of Computation," IBM Journal of Research and Development, vol. 17, no. 6, pp. 525-532, Nov 1973.
3. J C Das and D De, "Circuit switching with Quantum-Dot Cellular Automata," Nano communication networks, vol. 14, pp. 16-28, December 2017.
4. A Chaudhuri, M Sultana, D Sengupta, C Chauhduri, and A Chaudhuri, "A Comparative Analysis of CNF and XOR-AND Representations for QCA Majority Gate Estimation," in Annual Convention of Computer Society of India (CSI 2017), Springer, Kolkata, 2018, pp. 402-415.
5. A Chaudhuri, M Sultana, D Sengupta, C Chaudhuri, and A Chaudhuri, "A Reversible Approach to Two's Complement Addition using a Novel Reversible TCG Gate and its 4 Dot 2 Electron QCA Architecture," Microsystem Technologies, Springer, vol. 25, pp. 1965-1975, July 2018.
6. S Angizi, E Alkaldy, N Bagherzadeh, and K Navi, "Novel Robust Single Layer Wire Crossing Approach for Exclusive OR Sum of Products Logic Design with Quantum-Dot Cellular Automata," Journal of Low Power Electronics, vol. 10, pp. 256–271, 2014.
7. M Mohammadi, S Gorgin, and M Mohammadi, "Design of non-restoring divider in quantum-dot cellular automata technology," IET Circuits, Devices & Systems, vol. 11, no. 2, pp. 135–141, 2017.
8. NURIDDINSAFOEV, J S LEE, and J C JEON, "QCA XOR GATE FOR ARITHMETIC AND LOGIC CIRCUIT DESIGN," International Journal of Advanced Computational Engineering and Networking, vol. 5, no. 7, pp. 19-22, July 2017.
9. K Walus, T J Dysart, and G A Jullien, "QCADesigner: a rapid design and Simulation tool for quantum-dot cellular automata," IEEE Transactions on Nanotechnology, vol. 3, no. 1, pp. 26-31, March 2004.
10. C S Lent, P D Tougaw, W Porod, and G H Bernstein, "Quantum cellular automata," Nanotechnology, vol. 4, no. 1, pp. 49-57, 1993.
11. C S Lent, P D Tougaw, and W Porod, "Bistable saturation in coupled quantum dots for quantum cellular automata," Appl. Phys. Lett., vol. 62, p. 714, 1993.
12. C S Lent and P D Tougaw, "A device architecture for computing with quantum dots," Proc. IEEE, vol. 85, pp. 541-557, Apr 1997.
13. Craig S Lent and Beth Isaksen, "Clocked Molecular Quantum-Dot Cellular Automata," IEEE Transactions on Electron Devices, vol. 50, no. 9, pp. 1890-1896, September 2003.
14. D De, S Bhattacharaya , and K P Ghatak , Quantum Dots and Quantum Cellular Automata: Recent Trends and Applications.: Nova Science Publishers, 2013.
15. A Chaudhuri, M Sultana, D Sengupta, and A Chaudhuri, "A novel reversible two's complement gate (TCG) and its quantum mapping," in Devices for Integrated Circuit (DevIC), IEEE, Kalyani, 2017, pp. 252-256.
16. G L Snider et al., "Quantum-dot cellular automata: Review and recent experiments," Quantum-dot cellular automata: Review and recent experiments, vol. 85, no. 8, pp. 4283-4285, April 1999.
17. S S Kavitha and N Kaulgud, "Quantum dot cellular automata (QCA) design for the realization of basic logic gates," in 2017 IEEE International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques, Mysuru, 2017.
18. M Ziabari, A M Kassai, A Ziabari , and S E Maklavani, "Designing a Hamming Coder/Decoder Using QCAs," Journal of Applied Sciences, vol. 8, pp. 2569-2576, 2008.
19. G Xie, Y Xiang, Y Zhang, S Liu, and H LV, "Design and Implementation of Encoding and Check Code Circuit with Hamming Code on QCA," International Journal of Unconventional Computing, vol. 10, no. 5/6, pp. 391-404, 2014.
20. A Ahmadpour, A A Sha, and M Ziabari, "A Novel formulation of Hamming Code," in IEEE 6th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Pattaya, Thailand, 2009.
21. R Devadoss, K Paul, and M Balakrishnan, "Coplanar QCA crossovers," IET Electronics Letters, vol. 45, no. 24, pp. 1234-1235, November 2009.

AUTHORS PROFILE

Ayan Chaudhuri was born in Kolkata, WB, IN in 1987. He received his B.Tech degree in Computer Science and Engineering from Narula Institute of Technology, Kolkata, IN in 2009 and M.Tech degree in Distributed and Mobile Computing from Jadavpur University, Kolkata, WB, IN in 2013. Presently he is pursuing Ph.D. in Computer Science and Engineering from Jadavpur University, Kolkata, WB, IN. From June 2012 to May 2013 he worked as a Part Time Lecturer for Regent Education and Research Foundation, Barrackpore, WB, IN. From 2013 to 2016, he was an Associate System Engineer with IBM. Since August 2016, he has been a Management Trainee in the Dept. of Information Technology, Techno India, Kolkata. He has authored one Conference article, co-authored three Journals and four Conferences. His research interests include Addressing Nano-Technology through Synthesis of Reversible Logic using Quantum-Dot Cellular Automata.

Dr. Diganta Sengupta, B.Tech (2004) Electronics and Instrumentation Engineering from University of Kalyani, WB, IN. M.Tech (2010) CSE from Jadavpur University, WB, IN. Ph.D. (2016) from Jadavpur University, WB, IN. He is an IEEE member since 2016 and an ACM member in 2017. He is a life member of Computer Society of India (LM'16) and The Institution of Engineers (India) (M'16). He is also a member of IEEE Electron Device Society. Presently he is serving as the State Student Coordinator for West Bengal, India for Computer Society of India.

Dr. Diganta Sengupta is presently working in the capacity of Associate Professor in the Dept. of CSE, Techno International Batanagar, Kolkata, India. Formerly he was associated with the School of Computer Engineering (SCOPE), VIT University, Vellore, India.

His research interests include Decimal Numeric processors, Vedic Mathematics, Reversible Logic, Quantum Dot Cellular Automata, Taxonomy Generation Process and Hardware Accelerators for both classical as well as reversible computational engines. He has served as a reviewer for IEEE Access, JIKM (World Scientific), IJSAEM (Springer), IJBDCN (IGI-Global), ISSE (Springer) to name a few.

Dr. Chitrita Chaudhuri, B.ETCE (1980), M.ETCE (1982), Ph.D. Engg. (2017) from Jadavpur University, WB, IN 700032. She served the IT Industry between 1983 and 1995. Thereafter she joined the world of Academics as Visiting Teacher at Jadavpur University and Regional Computer Centre between 1995 and early 1998. She continued as full time Assistant Professor in the Kolkata Extension Centre of BIT (Mesra) from August, 1998. She also served as the Faculty-In-Charge of the said Institute for two years. Then she joined the department of Computer Science and Engineering at Jadavpur University as full time Faculty Member and is still continuing as Associate Professor there. Her current domain of research encompasses Data Mining, Artificial Intelligence, Reversible Logic, Quantum Dot Cellular Automata, Image Processing and Pattern Recognition.