Journal of Computational Electronics (2020) 19:419–434
https://doi.org/10.1007/s10825-019-01411-6

Nanocomputing channel fidelity of QCA code converter circuits under thermal randomness

Debashis De[1] · **Jadav Chandra Das**[1]

Published online: 23 October 2019

Abstract
The signal propagation and logical operations performed by the electrostatic interaction between the nanodots belonging to the quantum cellular automata (QCA) cells of different polarizations are influenced by the environmental noise like temperature fluctuations. The effect of thermal randomness on the computational fidelity of QCA-based 4-bit binary-to-Gray and binary-to-excess-3 code converters is studied in this article. The fidelity of computation of these digital circuits in the presence of noise is calculated by applying Shannon's information-theoretic measures, and thus, the robustness of the quantum cellular automata circuits to thermal noise is estimated. Finally, the temperature range over which the semiconductor quantum cellular automata circuits yield reliable computation is indicated. The proposed converters have minimum number of clock zones and high device density.

Keywords Binary-to-Gray code converter · Binary-to-excess-3 code converter · Information theory · Communication · QCA · Computational fidelity

1 Introduction

Nanoscale-based circuit design using quantum dot cellular automata (QCA) is focused by the researchers today because of the advantages of low energy consumption and high processing speed [1–7]. The fundamental QCA logic is based on the 3-input majority gate (MV), QCA wire and QCA inverter [8–15]. The architecture of QCA accommodates the above three cellular automata structures. Lent et al. [1] proposed a new paradigm for computation with cellular automata and also showed the construction and interconnection of basic digital logic gates. An adiabatic switching technique that permits the clocked control of the arrays of quantum dot cells performing useful computations was developed [6, 16–19]. Researchers already reported the experimental demonstration of a binary wire and logic gate using QCA [2–5, 8, 20, 21]. Lots of general-purpose combinational and sequential circuits have been designed using QCA [22–30]. The overview of logic redundancy schemes and the classical fault-tolerant approaches for circuit reliability was also discussed [7, 8, 14, 15, 31–34]. The computational efficiency of such circuits is, however, influenced by the environmental noise such as temperature variation. The efficacy of nanocomputing devices in the presence of noise can be estimated by Shannon's information-theoretic measures. In nanocomputing, two things are most important: (1) the efficacy of noisy channels to implement complex computational operations and (2) statistically how the efficacy varies in ensembles of channels having variable physical structures. Those aspects motivate the information-theoretic measures, which statistically confines the computational efficacy of noisy computation channels that assembled via processes. This induces the randomness in the structure of physical channel and is therefore tailor-made to characterize realistic nanocomputing channels. The stated confinement is generally appropriate for any artificial as well as natural nanocomputing channels, realized through a structured or in random nanonetwork, which can be modeled like discrete computation channel. Recently, Anderson et al. [35] have quantified the impact of structural randomness and temperature fluctuations on the efficiency of performance of the full-adder circuit from the information-theoretic point of view.

✉ Jadav Chandra Das
jadav2u@gmail.com

Debashis De
dr.debashis.de@wbut.ac.in

[1] Department of Computer Science and Engineering, West Bengal University of Technology, BF-142, Salt Lake, Sector-I, Kolkata 700064, India

The main aim of this work is to explore the robustness and trustworthiness of digital circuits in QCA through calculation of computational fidelity under thermal randomness and presence of noise. Thus, this work deals with the computational fidelity of QCA circuits, i.e., computation of channel fidelity in QCA channel routing when they will be implemented as noiseless and noisy nanocomputing channels. This work has the contributions as follows.

1. Computation of channel fidelity in QCA channel routing. The computation is performed for both noiseless and noisy QCA channels.
2. To perform the computation of channel fidelity, in this work, QCA-based 4-bit binary-to-Gray code and binary-to-excess-3 code converters have been considered as a routing channels.
3. The computational fidelity has been estimated applying Shannon's information-theoretic measures to confirm the robustness of the QCA channels.
4. It has been observed that the computational fidelity of QCA channel fluctuates by rising the temperatures. Thus, range of temperatures over which the QCA channels yield reliable computation is proposed.

This article has five sections as follows. The QCA-based design of both the converters is outlined in Sect. 2. Estimation of channel fidelity in QCA channel routing for noiseless and noisy QCA channels is given in Sect. 3. Comparative analysis is given in Sect. 4. Finally, the conclusion is made in Sect. 5.

2 QCA-based binary-to-Gray and excess-3 code converters

For an n-bit converter, the output bits ($Y_{i,n}$) of the Gray code related to the input bits ($X_{i,n}$) can be written as

$$Y_{i,n} = X_{i,n},\ Y_{i,n-1} = X_{i,n} \oplus X_{i,n-1}, \ldots, Y_{i,1} = X_{i,2} \oplus X_{i,1}$$

The truth table for conversion of 4-bit binary code into Gray code is given in Table 1 [12]. It is seen from the truth table that the most significant bit (MSB) of the Gray code is same with that of the input binary code. The logic expressions of the 4-bit binary code-to-Gray code converter circuit can be expressed as $W = A,\quad X = A \oplus B,\quad Y = B \oplus C,\quad Z = C \oplus D$. QCA-based majority function representations are given by $X = M(M(\overline{A}, B, 0), M(A, \overline{B}, 0), 1)$, $Y = M(M(\overline{B}, C, 0), M(B, \overline{C}, 0), 1)$, $Z = M(M(\overline{C}, D, 0), M(C, \overline{D}, 0), 1)$.

The QCA layout of this code converter circuit is outlined in Fig. 1. The circuit is developed using QCA Designer tool. The design is executed as follows.

1. Each XOR layout consists of 2 inverters and 3 MVs.
2. The inputs are in first clock zone.
3. The AND and OR operations necessary for the XOR operation are performed in the second and third clock zones, respectively.
4. The circuit layout consists of total 194 QCA cells over the area 920 nm × 240 nm, and three clock zones are required.
5. The simulation is achieved by the coherence vector method and it requires 5 iterations to converge into the initial steady-state polarization.

Table 1 Truth table for the conversion of 4-bit binary code into Gray code

X_i	A	B	C	D	Y_i	W	X	Y	Z
X_0	0	0	0	0	Y_0	0	0	0	0
X_1	0	0	0	1	Y_1	0	0	0	1
X_2	0	0	1	0	Y_2	0	0	1	1
X_3	0	0	1	1	Y_3	0	0	1	0
X_4	0	1	0	0	Y_4	0	1	1	0
X_5	0	1	0	1	Y_5	0	1	1	1
X_6	0	1	1	0	Y_6	0	1	0	1
X_7	0	1	1	1	Y_7	0	1	0	0
X_8	1	0	0	0	Y_8	1	1	0	0
X_9	1	0	0	1	Y_9	1	1	0	1
X_{10}	1	0	1	0	Y_{10}	1	1	1	1
X_{11}	1	0	1	1	Y_{11}	1	1	1	0
X_{12}	1	1	0	0	Y_{12}	1	0	1	0
X_{13}	1	1	0	1	Y_{13}	1	0	1	1
X_{14}	1	1	1	0	Y_{14}	1	0	0	1
X_{15}	1	1	1	1	Y_{15}	1	0	0	0

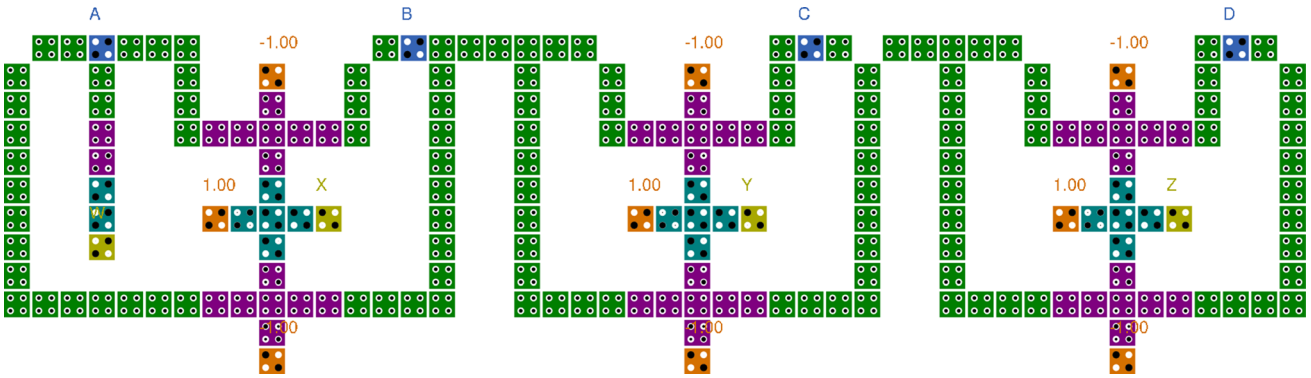

Fig. 1 QCA layout of 4-bit binary code-to-Gray code converter

6. The required clock zone is three. Thus, after ¾ clock cycles the correct output will be appeared as displayed in Fig. 2.

The computational efficacy of both the code converter circuits designed in this article can be analyzed by considering them as the computational channel where the output(s) can arise from the combination of the different inputs unlike the noiseless communication channel having one-to-one mapping between the input(s) and output(s) as shown in Fig. 3.

The truth table for conversion of 4-bit binary code into excess-3 code is shown in Table 2. Here, the circuit is designed using four serial adders as presented in Fig. 4. The combinations of the input bit sequence range from "0000" to "1111" (i.e., 0–15). Since any output of the excess-3 code converter is increased by three with respect to a specific binary input, four bits cannot represent the correct output of the converter for the input combinations "1101," "1110" and "1111" (i.e., 13–15). The most significant bit (*E*) denotes the error in the output of the excess-3 code converter as

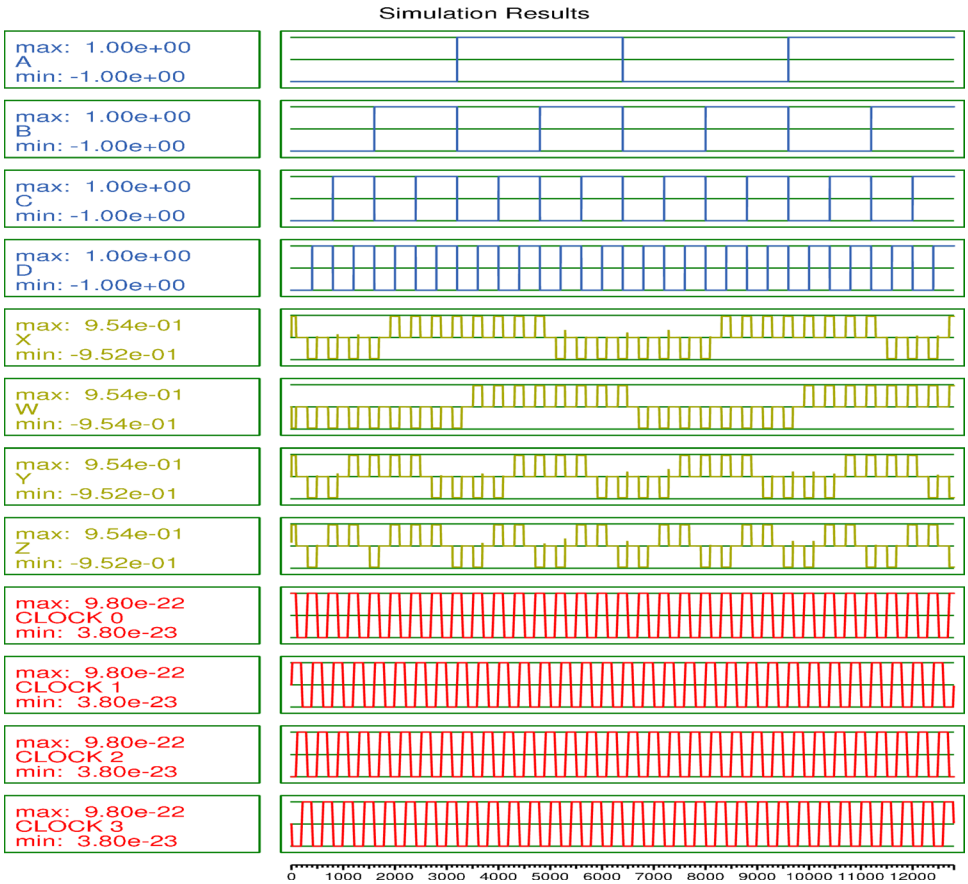

Fig. 2 Simulation result for 4-bit binary code-to-Gray code converter

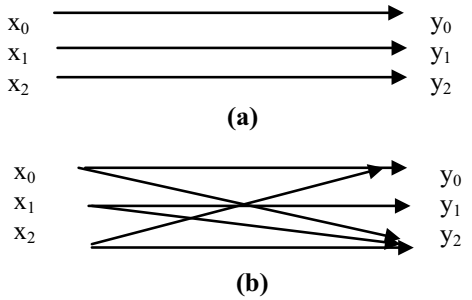

Fig. 3 **a** Noiseless communication channel, **b** computation channel

shown in the truth table for the last three combinations of the binary input.

The circuit layout of the binary-to-excess-3 code converter shown in Fig. 5 is designed using QCA Designer simulation tool. The design is carried out as follows.

1. The inputs are in first clock zone.
2. Each of the full adders requires four clock zones to execute.
3. The layout consists of 686 cells including four inputs and five outputs.
4. The simulation is calculated by coherence vector method and it requires 10 iterations to converge into the initial steady-state polarization, and the output is shown in Fig. 6.

Table 2 Truth table for the binary-to-excess-3 code converter

X_i	A	B	C	D	Y_i	W	X	Y	Z
X_0	0	0	0	0	Y_0	0	0	1	1
X_1	0	0	0	1	Y_1	0	1	0	0
X_2	0	0	1	0	Y_2	0	1	0	1
X_3	0	0	1	1	Y_3	0	1	1	0
X_4	0	1	0	0	Y_4	0	1	1	1
X_5	0	1	0	1	Y_5	1	0	0	0
X_6	0	1	1	0	Y_6	1	0	0	1
X_7	0	1	1	1	Y_7	1	0	1	0
X_8	1	0	0	0	Y_8	1	0	1	1
X_9	1	0	0	1	Y_9	1	1	0	0
X_{10}	1	0	1	0	Y_{10}	1	1	0	1
X_{11}	1	0	1	1	Y_{11}	1	1	1	0
X_{12}	1	1	0	0	Y_{12}	1	1	1	0
X_{13}	1	1	0	1	Y_{13}	0	0	0	0
X_{14}	1	1	1	0	Y_{14}	0	0	0	1
X_{15}	1	1	1	1	Y_{15}	0	0	1	0

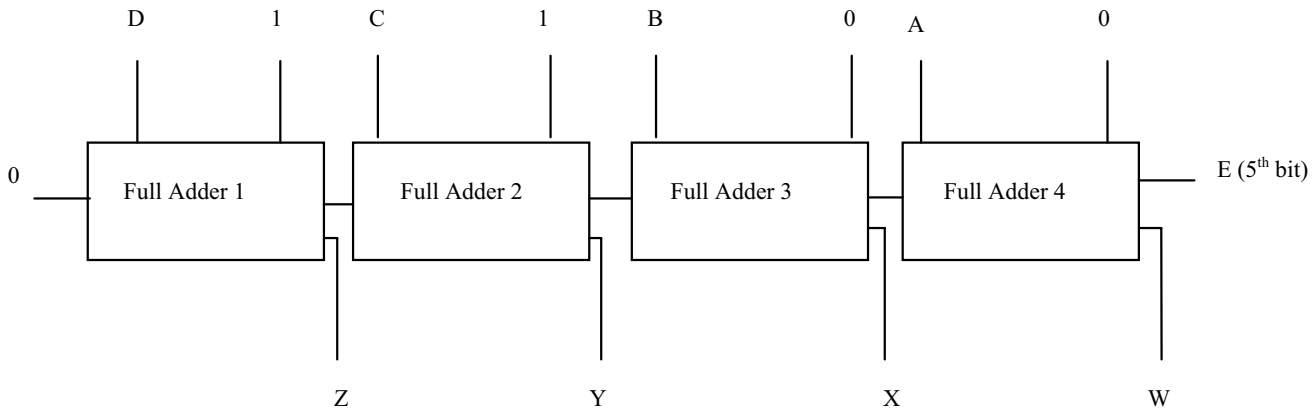

Fig. 4 Design of binary-to-excess-3 code converter

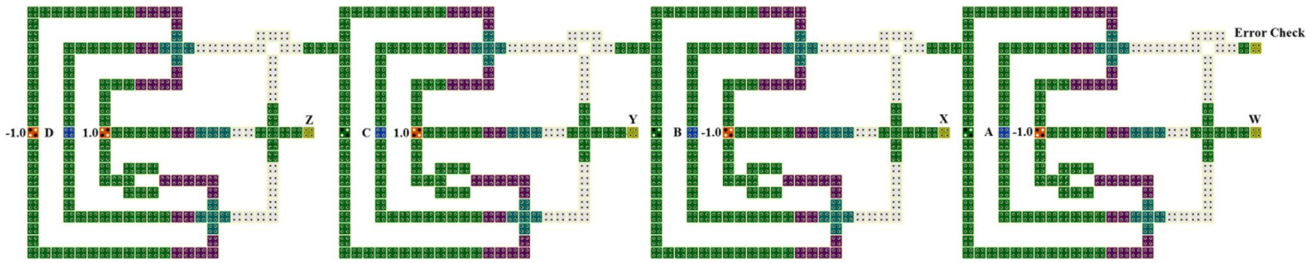

Fig. 5 QCA layout of binary code-to-excess-3 code converter

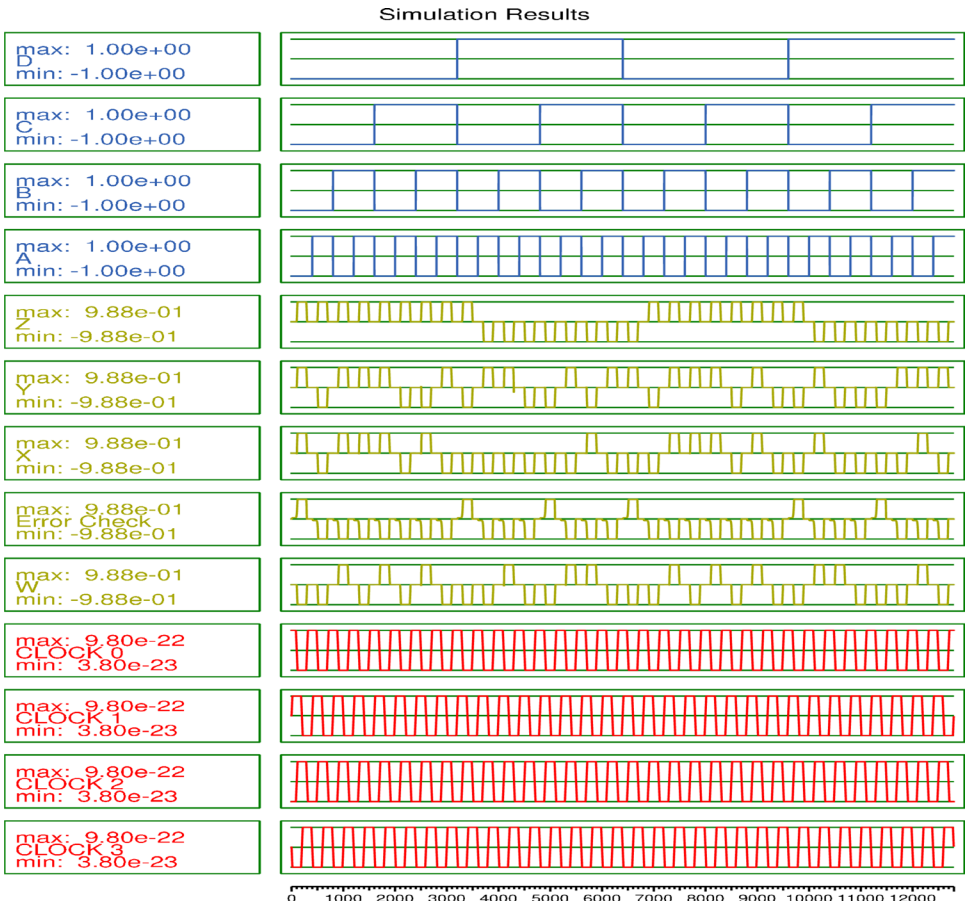

Fig. 6 Simulation result for binary-to-excess-3 code converter

The simulation result of proposed converters as shown in Figs. 2 and 6 is compared with their theoretic values. The evaluation agreed the circuit's functional efficiency. The parameters used for coherence vector simulation of proposed QCA layout is shown in Fig. 7. The height as well as breadth of QCA cell used to achieve the circuit is 18 nm.

3 Computational fidelity in QCA channel routing

The logical 4-bit binary code-to-Gray code as well as binary-to-excess-3 code converter has no information loss as because the number of inputs and that of the corresponding outputs are equal. Besides, each input line and

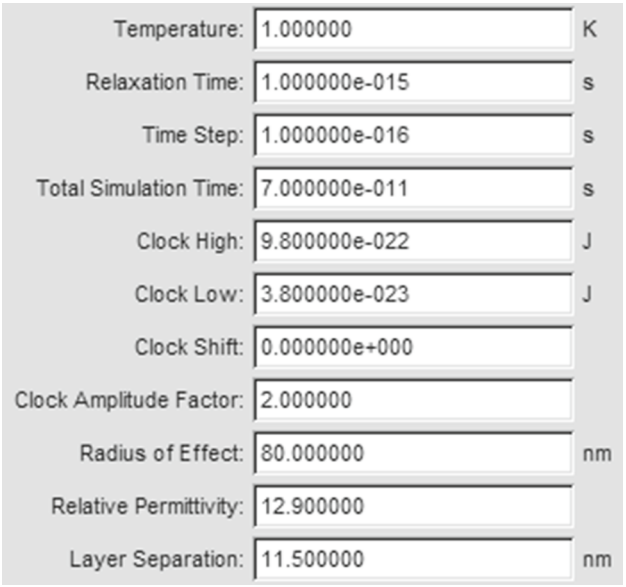

Fig. 7 Coherence vector simulation parameter list

output line are distinct in nature. Thus, this computational channel can behave almost like a communicational channel. In particular, when both the converters having equal number of input and output bits perform ideal logical transformation, then they will behave as an ideal noiseless communication channel.

The 4-bit binary code-to-Gray code converter has 4-bit output for each 4-bit input. But binary code-to-excess-3 code converter has 5-bit output for each 4-bit input. So, there must be 32 ($2^5 = 32$) distinct output combinations. But, out of these 32 output combinations, logically only 16 output combinations are valid. As there is only 16 input combinations, the number of output combinations must be ≤ 16. Thus, binary code-to-Gray code as well as binary-to-excess-3 code converter is 16-input and 16-output channel.

Though both the converters have equal number of inputs as well as outputs, the QCA circuit of binary-to-excess-3 code converter as shown in Fig. 5 is more complex than that of the QCA circuit of binary code-to-Gray code converter circuit as shown in Fig. 1. Binary-to-excess-3 code converter required more QCA cells, MVs and clocking zones. Thus, it is better to study how computational fidelity varies in QCA circuits, which have same number of input and output combinations. This motivates the selection of the proposed converters as a QCA routing channels.

The mean or average information per message (considering equal probability $\{p_i\}$ of input bits of message (m) defined in terms of M-array input alphabet $\{x_i\}$ and N-array output alphabet $\{y_j\}$with probability $\{q_j\}$) transmitted by the discrete memory less source is given by the entropy [35, 36],

$$H(m) = \sum_{i=0}^{M-1} p_i I_i \tag{1}$$

where $I_i = \log_2\left(\frac{1}{p_i}\right)$ bits represent the information content in the message m. Thus, Eq. (1) can be written as

$$\begin{aligned} H(m) &= \sum_{i=0}^{M-1} p_i \log_2\left(\frac{1}{p_i}\right) \\ &= -\sum_{i=0}^{M-1} p_i \log_2 p_i \text{bits} \end{aligned} \tag{2}$$

Since the nanodots are arranged in the QCA cells, form the channel for carrying signal and circuit for performing computation, Shannon's information theory-based measures can be applied to evaluate the amount of uncertainty in obtaining the correct output as specified by the logical transformation of the inputs given to the circuits in the presence of noise. The 4-bit binary number contains total 16 combinations of the input (x_i) to both the binary code-to-Gray code and binary-to-excess-3 code converters and the output (y_j) of the circuit also contains 16 combinations as the input. So, $p_i = q_i = 1/16$. Using Eq. (2), the Shannon's entropy for the input of both the converters is given by,

$$\begin{aligned} H(X_{abcd}) &= -\sum_{i=0}^{M-1} p_i \log_2 p_i \\ &= -\sum_{i=0}^{15} \left(\frac{1}{16}\right) \log_2\left(\frac{1}{16}\right) \\ &= -16\left[\left(\frac{1}{16}\right) \log_2\left(\frac{1}{16}\right)\right] \\ &= 4 \end{aligned}$$

The average uncertainty about the input bits (x_i) with respect to the received output bits (y_j) is calculated by the conditional entropy as follows [35, 36]

$$H(X|Y) = \sum_{j=0}^{N-1} q_j H(X|y_j) = \sum_{j=0}^{N-1} q_j \left(-\sum_{i=0}^{M-1} p_{i|j} \log_2 p_{i|j}\right) \tag{3}$$

where $H(X|y_j) = -\sum_{i=0}^{M-1} p_{i|j} \log_2 p_{i|j}$ is the average entropy for all output bits y_j and $p_{i|j} = (q_{i|j} p_i / q_j)$ stands for conditional

probability of output bit (y_j) with respect to input bits (x_i) [35]. Similarly, $q_{i|j}$ is the transition probability. $q_{i|j}$ will be 1 if $i \in \{i\}_j$; otherwise, it will be zero.

The mutual information received at the output of the circuits, i.e., the information present in output Y about input X is formulated by,

$$I(X;Y) = H(X) - H(X|Y) \tag{4}$$

In practical cases, the array of QCA cells can behave as noisy channel in the presence of thermal randomness because of the change in maximum output polarization (MOP) of the cells with temperature. Consequently, the logical transformation effectively done by the circuits may deviate from the predicted output.

The computational fidelity (F_L) measure provides the amount of information about how far the noisy channel output resembles the correct output expected from the logical transformation of the same input distribution, i.e.,$\{x_i\}$ with probability mass function $\{p_i\}$ performed by a noiseless channel [35]. Basically, computational fidelity measure provides a more widespread description of computational efficacy. It is an information theory-based measure that reveals the statistical information of correlations between inputs and outputs in a noisy computing channel, yielding quantitative data about computational abilities of the channel that transcend a specific cluster of the M channel outputs ($\{z_k\}$) into N abstract outputs y_j (logical) and assignment of N sequence of digits to this y_j. Besides, the computational fidelity measure is more versatile due to their information-theoretic nature and the entropic scenario of information-theoretic approach enables connections from computational efficacy to the physical costs for noisy computation. In more general, the computational fidelity measure statistically computes the distinguishability of outputs resulting from the inputs which belongs to the poles apart logical outputs. Note that the computational fidelity measure is mainly used in quantum computation to quantify the closeness behavior of physical outputs to desire states.

So, we consider an M-input ($\{x_i\}$), K-output ($\{z_k\}$) noisy discrete channel as binary-to-Gray and binary-to-excess-3 code converters and the channels are characterized by a channel matrix ($\{\pi_{k|i}\}$) that represents the conditional probability of the occurrence of a particular output ($\{z_k\}$) for a particular input x_i.

For both the circuits, the amount of information about the extent to which the noisy channel generates correct logical output which is analogous to that of the ideal channel is given by the mutual information between the noise-free and noisy channel outputs as [35] as shown in Eq. (5).

$$I(Z;Y) = H(Z) - H(Z|Y) \tag{5}$$

where

$$H(Z) = -\sum_{k=0}^{k-1} \pi_k \log_2 \pi_k \tag{6}$$

and

$$\pi_k = \sum_i p_i \pi_{k|i} \tag{7}$$

In Eq. (5), Y stands for output of ideal channel and Z denotes output for noise channel. $H(Z)$ is the self-entropy for noise channel output Z and $H(Z|Y)$ denotes average entropy for noisy channel output.

The conditional entropy in the noisy channel output, i.e., average entropy for both the converters can be calculated from the expression [35] as shown in Eq. (8).

$$H(Z|Y) = \sum_{j=0}^{N-1} q_j H(Z|y_j) = \sum_{j=0}^{N-1} q_j \left(-\sum_{k=0}^{k-1} \pi_k^{(j)} \log_2 \pi_k^{(j)} \right) \tag{8}$$

where

$$\pi_k^{(j)} = \frac{1}{q_j} \sum_{i \in \{i\}_j} p_i \pi_{k|i} \tag{9}$$

In Eq. (8), $H(Z|y_j)$ is the entropy given that $Y = y_j$ and $\pi_k^{(j)}$ is the probability for $Z = z_k$ when $Y = y_j$. For a noisy computing channel $N = \{\{x_i\}, \{z_k\}, \{\pi_{k|i}\}\}$ with input probability mass function$\{p_i\}$, the computational fidelity can be written as

$$F_L = \frac{I(Z;Y)}{H_L(Y)} \tag{10}$$

where

$$H_L(Y) = -\sum_j q_j \log_2 q_j \tag{11}$$

In Eq. (10), $H_L(Y)$ is the output entropy associated with the logical transformation of the ideal channel, i.e., the output entropy for logical transformation L with input probability mass function $\{p_i\}$ [35]. The computational fidelity of the proposed circuits can be calculated using Eq. (10).

3.1 Computational fidelity in noiseless QCA channel routing

In this section, the measure of computational fidelity of 4-bit binary code-to-Gray code converter and binary code-to-excess-3 code converter is performed when they behaves like a noiseless computing channels.

The conditional probabilities comprising the channel matrix ($\{\pi_{k|i}\}$) for the binary-to-Gray code converter (for the binary inputs 0–15) are presented in Eq. (12).

$$\pi_{0|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 - P_i^X\right)\left(1 - P_i^Y\right)\left(1 - P_i^Z\right) \tag{12a}$$

$$\pi_{1|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 - P_i^X\right)\left(1 - P_i^Y\right)\left(1 + P_i^Z\right) \quad (12b)$$

$$\pi_{2|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 - P_i^X\right)\left(1 + P_i^Y\right)\left(1 + P_i^Z\right) \quad (12c)$$

$$\pi_{3|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 - P_i^X\right)\left(1 + P_i^Y\right)\left(1 - P_i^Z\right) \quad (12d)$$

$$\pi_{4|i} = \left(\frac{1}{16}1 - P_i^W\right)\left(1 + P_i^X\right)\left(1 + P_i^Y\right)\left(1 - P_i^Z\right) \quad (12e)$$

$$\pi_{5|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 + P_i^X\right)\left(1 + P_i^Y\right)\left(1 + P_i^Z\right) \quad (12f)$$

$$\pi_{6|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 + P_i^X\right)\left(1 - P_i^Y\right)\left(1 + P_i^Z\right) \quad (12g)$$

$$\pi_{7|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 + P_i^X\right)\left(1 - P_i^Y\right)\left(1 - P_i^Z\right) \quad (12h)$$

$$\pi_{8|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 + P_i^X\right)\left(1 - P_i^Y\right)\left(1 - P_i^Z\right) \quad (12i)$$

$$\pi_{9|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 + P_i^X\right)\left(1 - P_i^Y\right)\left(1 + P_i^Z\right) \quad (12j)$$

$$\pi_{10|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 + P_i^X\right)\left(1 + P_i^Y\right)\left(1 + P_i^Z\right) \quad (12k)$$

$$\pi_{11|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 + P_i^X\right)\left(1 + P_i^Y\right)\left(1 - P_i^Z\right) \quad (12l)$$

$$\pi_{12|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^X\right)\left(1 + P_i^Y\right)\left(1 - P_i^Z\right) \quad (12m)$$

$$\pi_{13|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^X\right)\left(1 + P_i^Y\right)\left(1 + P_i^Z\right) \quad (12n)$$

$$\pi_{14|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^X\right)\left(1 - P_i^Y\right)\left(1 + P_i^Z\right) \quad (12o)$$

$$\pi_{15|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^X\right)\left(1 - P_i^Y\right)\left(1 - P_i^Z\right) \quad (12p)$$

where P_i^W,P_i^X,P_i^Y and P_i^Z denote the polarization of output cell W, X, Y and Z, respectively, for ith input $\{x_i\}$. If it is assumed that the proposed channels have the logical outputs, i.e., all the outputs which are logically true provide perfect 1 and all the outputs which are logically false provide perfect 0. Then, the channel matrix ($\{\pi_{k|i}\}$) for the binary-to-Gray code converter can be estimated (Table 3) using Eqs. (12a–12p).

The conditional probabilities comprising the channel matrix ($\{\pi_{k|i}\}$) for the binary-to-excess-3 code converter (for the binary inputs 0–15) are presented in Eq. (13).

$$\pi_{0|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 - P_i^Y\right)\left(1 + P_i^X\right)\left(1 + P_i^Z\right)\left(1 - P_i^E\right) \quad (13a)$$

$$\pi_{1|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 + P_i^Y\right)\left(1 - P_i^X\right)\left(1 - P_i^Z\right)\left(1 - P_i^E\right) \quad (13b)$$

$$\pi_{2|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 + P_i^Y\right)\left(1 - P_i^X\right)\left(1 + P_i^Z\right)\left(1 - P_i^E\right) \quad (13c)$$

Table 3 Channel matrix for the binary code-to-Gray code converter

	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	Y_{10}	Y_{11}	Y_{12}	Y_{13}	Y_{14}	Y_{15}
X_0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X_1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X_2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
X_3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
X_4	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
X_5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
X_6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
X_7	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
X_8	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
X_9	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
X_{10}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
X_{11}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
X_{12}	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
X_{13}	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
X_{14}	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
X_{15}	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

$$\pi_{3|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 + P_i^Y\right)\left(1 + P_i^X\right)\left(1 - P_i^Z\right)\left(1 - P_i^E\right) \tag{13d}$$

$$\pi_{4|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 + P_i^Y\right)\left(1 + P_i^X\right)\left(1 + P_i^Z\right)\left(1 - P_i^E\right) \tag{13e}$$

$$\pi_{5|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^Y\right)\left(1 - P_i^X\right)\left(1 - P_i^Z\right)\left(1 - P_i^E\right) \tag{13f}$$

$$\pi_{6|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^Y\right)\left(1 - P_i^X\right)\left(1 + P_i^Z\right)\left(1 - P_i^E\right) \tag{13g}$$

$$\pi_{7|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^Y\right)\left(1 + P_i^X\right)\left(1 - P_i^Z\right)\left(1 - P_i^E\right) \tag{13h}$$

$$\pi_{8|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^Y\right)\left(1 + P_i^X\right)\left(1 + P_i^Z\right)\left(1 - P_i^E\right) \tag{13i}$$

$$\pi_{9|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 + P_i^Y\right)\left(1 - P_i^X\right)\left(1 - P_i^Z\right)\left(1 - P_i^E\right) \tag{13j}$$

$$\pi_{10|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 + P_i^Y\right)\left(1 - P_i^X\right)\left(1 + P_i^Z\right)\left(1 - P_i^E\right) \tag{13k}$$

$$\pi_{11|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 + P_i^Y\right)\left(1 + P_i^X\right)\left(1 - P_i^Z\right)\left(1 - P_i^E\right) \tag{13l}$$

$$\pi_{12|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 + P_i^Y\right)\left(1 + P_i^X\right)\left(1 + P_i^Z\right)\left(1 + P_i^E\right) \tag{13m}$$

$$\pi_{13|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 - P_i^Y\right)\left(1 - P_i^X\right)\left(1 - P_i^Z\right)\left(1 + P_i^E\right) \tag{13n}$$

$$\pi_{14|i} = \frac{1}{16}\left(1 - P_i^W\right)\left(1 - P_i^Y\right)\left(1 - P_i^X\right)\left(1 + P_i^Z\right)\left(1 + P_i^E\right) \tag{13o}$$

$$\pi_{15|i} = \frac{1}{16}\left(1 + P_i^W\right)\left(1 - P_i^Y\right)\left(1 + P_i^X\right)\left(1 - P_i^Z\right)\left(1 + P_i^E\right) \tag{13p}$$

where P_i^W,P_i^X,P_i^Y,P_i^Z and P_i^E denote the polarization of output cell W, X, Y, Z, and E respectively, for ith input $\{x_i\}$. If it is assumed that the proposed channels have the logical outputs, i.e., all the outputs which are logically true provide perfect 1 and all the outputs which are logically false provide perfect 0. Then, the channel matrix ($\{\pi_k|i\}$) for binary-to-excess-3 code converter can be estimated (Table 4) using Eqs. (13a–13p).

Now, in case of binary-to-Gray code converter, for noiseless communication, $P_i^W = P_i^X = P_i^Y = P_i^Z = 1$ and then from Eq. (12), we can write

$$\begin{aligned}\pi_{k|i} &= \frac{1}{16}(1+1)(1+1)(1+1)(1+1)\\ &= \left(\frac{1}{16} \times 16\right)\\ &= 1\end{aligned}$$

Table 4 Channel matrix for the binary-to-excess-3 code converter

	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	Y_{10}	Y_{11}	Y_{12}	Y_{13}	Y_{14}	Y_{15}
X_0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
X_1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
X_2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
X_3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
X_4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
X_5	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
X_6	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
X_7	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
X_8	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
X_9	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
X_{10}	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
X_{11}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
X_{12}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
X_{13}	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X_{14}	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X_{15}	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Using Eq. (7), for every value of k ($k=0, 1, \ldots, 15$), π_k can be calculated as

$$\begin{aligned}\pi_k &= \sum_i p_i \pi_{k|i} \\ &= \left(\frac{1}{16} \times 1\right)\left(\text{For 4 - bit binary code-to-Gray code converter } p_i = \frac{1}{16}\right) \\ &= \frac{1}{16}\end{aligned}$$

Therefore,

$$\begin{aligned}H(Z) &= -\sum_{k=0}^{k-1} \pi_k \log_2 \pi_k \\ &= -\sum_{k=0}^{15} \pi_k \log_2 \pi_k \\ &= -16\left(\frac{1}{16} \log_2 \frac{1}{16}\right) \\ &= -\log_2 \frac{1}{16} \\ &= 4\end{aligned}$$

Again from Eq. (9), $\pi_k^{(j)}$ can be calculated as

$$\begin{aligned}\pi_k^{(j)} &= \frac{1}{q_j} \sum_{i \in \{i\}_j} p_i \pi_{k|i} \left(\text{For 4 - bit binary-to-Gray code converter } q_i = \frac{1}{16}\right) \\ &= 16 \times \frac{1}{16} \\ &= 1\end{aligned}$$

Thus,

$$\begin{aligned}H(Z|y_j) &= -\sum_{k=0}^{k-1} \pi_k^{(j)} \log_2 \pi_k^{(j)} \\ &= -\sum_{k=0}^{15} \pi_k^{(j)} \log_2 \pi_k^{(j)} \\ &= -\frac{1}{16}(1 \log_2 1) \\ &= 0\end{aligned}$$

$$\begin{aligned}H(Z|Y) &= \sum_{j=0}^{15} q_j H(Z|y_j) \\ &= \frac{1}{16} \times 0 \\ &= 0\end{aligned}$$

$$\begin{aligned}I(Z;Y) &= H(Z) - H(Z|Y) \\ &= 4 - 0 \\ &= 4\end{aligned}$$

$$\begin{aligned}H_L(Y) &= -\sum_j q_j \log_2 q_j \\ &= -16\left(\frac{1}{16} \log_2 \frac{1}{16}\right) \\ &= 4\end{aligned}$$

Placing the value of $I(Z;Y)$ and $H_L(Y)$ in Eq. (10), the computational fidelity can be calculated as

$$\begin{aligned}F_{\mathrm{L}} &= \frac{I(Z;Y)}{H_L(Y)} \\ &= \frac{4}{4} \\ &= 1\end{aligned}$$

So, fidelity (F_{L}) will be 1 when the circuit behaves like a noiseless computing channel.

Similarly, in case of binary-to-excess-3 converters, it can be shown that the fidelity (F_{L}) will be 1 when it behaves like a noiseless computing channel.

It can be noted that the computational fidelity is within the range $0 \le F_{\mathrm{L}} \le 1$. The equality in the lower bound occurs when the noisy channel output, i.e., Z has no knowledge about the logical output Y. On the other hand, the equality in the upper bound occurs when Y can be inferred from Z without ambiguity, i.e., $I(Z;Y)$ has maximum value of $H_{\mathrm{L}}(Y)$ [35]. Thus, it can be noted that if the output of any circuit is correspondent to logical one then the circuit will achieve the maximum fidelity, i.e., $F_{\mathrm{L}}=1$ and it indicates no induced noise in the outputs. But, less than one means the noise is present in the outputs, i.e., the output is produced with induced noise.

3.2 Computational fidelity in noisy QCA channel routing

Fidelity may vary based on induced noise, which may present due to dissipated power, structural randomness and thermal

randomness. In this section, the estimation of fidelity for both the circuits is performed considering thermal randomness.

To perform the estimation of fidelity, proposed binary code-to-Gray code converter circuit as shown in Fig. 1 has been simulated on QCA Designer tool at different temperatures such as 1 K and 2 K. From the simulation outcome, the polarization of each output cell at specific temperature is observed and utilized in calculation of fidelity. For example, if the circuit is simulated at 1 K temperature, then each output cell has the maximum output polarization (MOP) for logical true as follows.

$$P_i^W = P_i^X = P_i^Y = P_i^Z = 0.954$$

From Eq. (12), we can write

$$\begin{aligned}\pi_{k|i} &= \frac{1}{16}(1+0.954)(1+0.954)(1+0.954)(1+0.954)\\ &= \frac{14.578}{16}\\ &= 0.9111\end{aligned}$$

Using Eq. (7), for every value of k (k=0, 1, ….., 15), π_k can be calculated as

$$\begin{aligned}\pi_k &= \sum_i p_i \pi_{k|i}\\ &= \left(\frac{1}{16}\times 0.9111\right)\left(\text{For 4 - bit binary code-to-Gray code converter } p_i = \frac{1}{16}\right)\\ &= \frac{0.9111}{16}\\ &= 0.05694\end{aligned}$$

Therefore, $$\begin{aligned}H(Z) &= -\sum_{k=0}^{k-1} \pi_k \log_2 \pi_k\\ &= -16\left[\left(\frac{0.9111}{16}\right)\log_2\left(\frac{0.9111}{16}\right)\right]\\ &= -(0.9111)\log_2\left(\frac{0.9111}{16}\right)\\ &= 3.7670\end{aligned}$$

Again from Eq. (9), $\pi_k^{(j)}$ can be estimated as

$$\begin{aligned}\pi_k^{(j)} &= \frac{1}{q_j}\sum_{i\in\{i\}_j} p_i \pi_{k|i}\\ &= 16\times\frac{1}{16}\times(0.9111)\\ &= 0.9111\end{aligned}$$

Thus, $$\begin{aligned}H(Z|y_j) &= -\sum_{k=0}^{k-1} \pi_k^{(j)} \log_2 \pi_k^{(j)}\\ &= -16[(0.9111)\log_2(0.9111)]\\ &= 1.9575\end{aligned}$$

$$\begin{aligned}H(Z|Y) &= \sum_{j=0}^{15} q_j H(Z|y_j)\\ &= \frac{1}{16}(16\times 1.9575)\\ &= 1.9575\end{aligned}$$

$$\begin{aligned}I(Z;Y) &= H(Z) - H(Z|Y)\\ &= 3.7670 - 1.9575\\ &= 1.8094\end{aligned}$$

Table 5 Fidelity calculation of binary code-to-Gray code converter

Temperature (K)	MOP	π_{kli}	H(z)	$\pi_k^{(j)}$	$H(Z\|y_j)$	$H(Z\|Y)$	$I(Z;Y)$	F_L
1	0.954	0.9111	3.7670	0.9111	1.9575	1.9575	1.8094	0.4567
2	0.954	0.9111	3.7670	0.9111	1.9575	1.9575	1.8094	0.4567
3	0.954	0.9111	3.7670	0.9111	1.9575	1.9575	1.8094	0.4567
4	0.954	0.9111	3.7670	0.9111	1.9575	1.9575	1.8094	0.4567
5	0.953	0.9093	3.7621	0.9093	1.9957	1.9957	1.7664	0.4416
6	0.950	0.9037	3.7469	0.9037	2.1126	2.1126	1.6343	0.4086
7	0.940	0.8853	3.6960	0.8853	2.4899	2.4899	1.2062	0.3015
8	0.935	0.8762	3.6720	0.8762	2.6731	2.6731	0.9989	0.2497
9	0.922	0.8529	3.6074	0.8529	3.1328	3.1328	0.4746	0.1187

$$H_{\mathrm{L}}(Y) = -\sum_j q_j \log_2 q_j$$
$$= -16\left(\frac{1}{16}\log_2\frac{1}{16}\right)$$
$$= 4$$

Placing the value of $I(Z;Y)$ and $H_{\mathrm{L}}(Y)$ in Eq. (10), the computational fidelity at 1 K temperature can be calculated as

$$F_{\mathrm{L(For1\,K)}} = \frac{I(Z;Y)}{H_{\mathrm{L}}(Y)}$$
$$= \frac{1.8094}{4}$$
$$= 0.4567$$

Similarly, at different temperatures, the computational fidelity of the binary code-to-Gray code converter is estimated and tabulated in Table 5.

In similar approach, the estimation of computational fidelity of proposed binary-to-excess-3 code converter for different temperatures such as 1 K and 2 K has been performed. For example, if the circuit is simulated at 1 K temperature, then each output cell has the maximum output polarization (MOP) for logical true as follows.

$$P_i^W = P_i^X = P_i^Y = P_i^Z = 0.988$$

From Eq. (13), we can write

$$\pi_{k|i} = \frac{1}{16}(1+0.988)(1+0.988)(1+0.988)(1+0.988)$$
$$= \frac{15.6194}{16}$$
$$= 0.9762$$

Using Eq. (7), for every value of k (k=0, 1, ….., 15), π_k can be calculated as

$$\pi_k = \sum_i p_i \pi_{k|i}$$
$$= \left(\frac{1}{16}\times 0.9762\right)\left(\text{For 4 - bit binary code-to-Gray code converter } p_i = \frac{1}{16}\right)$$
$$= \frac{0.9762}{16}$$
$$= 0.06101$$

Therefore, $$H(Z) = -\sum_{k=0}^{k-1} \pi_k \log_2 \pi_k$$
$$= -16\left[\left(\frac{0.9762}{16}\right)\log_2\left(\frac{0.9762}{16}\right)\right]$$
$$= 3.9388$$

Again from Eq. (9), $\pi_k^{(j)}$ can be calculated as

$$\pi_k^{(j)} = \frac{1}{q_j}\sum_{i\in\{i\}_j} p_i \pi_{k|i}$$
$$= 16\times\frac{1}{16}\times(0.9762)$$
$$= 0.9762$$

Thus, $$H(Z|y_j) = -\sum_{k=0}^{k-1} \pi_k^{(j)} \log_2 \pi_k^{(j)}$$
$$= -16[(0.9762)\log_2(0.9762)]$$
$$= 0.5425$$

$$H(Z|Y) = \sum_{j=0}^{15} q_j H(Z|y_j)$$
$$= \frac{1}{16}(16\times 0.5425)$$
$$= 0.5425$$

$$I(Z;Y) = H(Z) - H(Z|Y)$$
$$= 3.9388 - 0.5425$$
$$= 3.3963$$

$$H_L(Y) = -\sum_j q_j \log_2 q_j$$
$$= -16\left(\frac{1}{16}\log_2\frac{1}{16}\right)$$
$$= 4$$

Placing the value of $I(Z;Y)$ and $H_{\mathrm{L}}(Y)$ in Eq. (10), the computational fidelity 1 K temperature can be calculated as

$$F_{L(\mathrm{For1}K)} = \frac{I(Z;Y)}{H_L(Y)}$$
$$= \frac{3.3988}{4}$$
$$= 0.8497$$

Table 6 Fidelity calculation of binary-to-excess-3 code converter

Temperature (K)	MOP	Π_{kli}	$H(z)$	$\Pi_k^{(j)}$	$H(Z\|y_j)$	$H(Z\|Y)$	$I(Z;Y)$	F_L
1	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
2	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
3	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
4	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
5	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
6	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
7	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
8	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
9	0.988	0.9762	3.9388	0.9762	0.5425	0.5425	3.3963	0.8497
10	0.987	0.9743	3.9334	0.9473	0.5860	0.5860	3.3474	0.8368
11	0.985	0.9703	3.9236	0.9703	0.6745	0.6745	3.2491	0.8123
12	0.983	0.9664	3.9135	0.9664	0.7617	0.7617	3.1517	0.7879
13	0.980	0.9604	3.8982	0.9604	0.8914	0.8914	3.0068	0.7517
14	0.977	0.9548	3.8830	0.9548	1.0197	1.0197	2.8633	0.7158
15	0.972	0.9451	3.6476	0.9451	1.2304	1.2304	2.4157	0.6039
16	0.966	0.9337	3.8274	0.9337	1.4783	1.4783	2.3491	0.5872
17	0.960	0.9224	3.7971	0.9224	1.7206	1.7206	2.0769	0.5191
18	0.952	0.9074	3.7569	0.9074	2.0354	2.0354	1.7216	0.4304
19	0.944	0.8926	3.7169	0.8926	2.3405	2.3405	1.3762	0.3440
20	0.935	0.8762	3.6720	0.8762	2.6731	2.6731	0.9989	0.2497

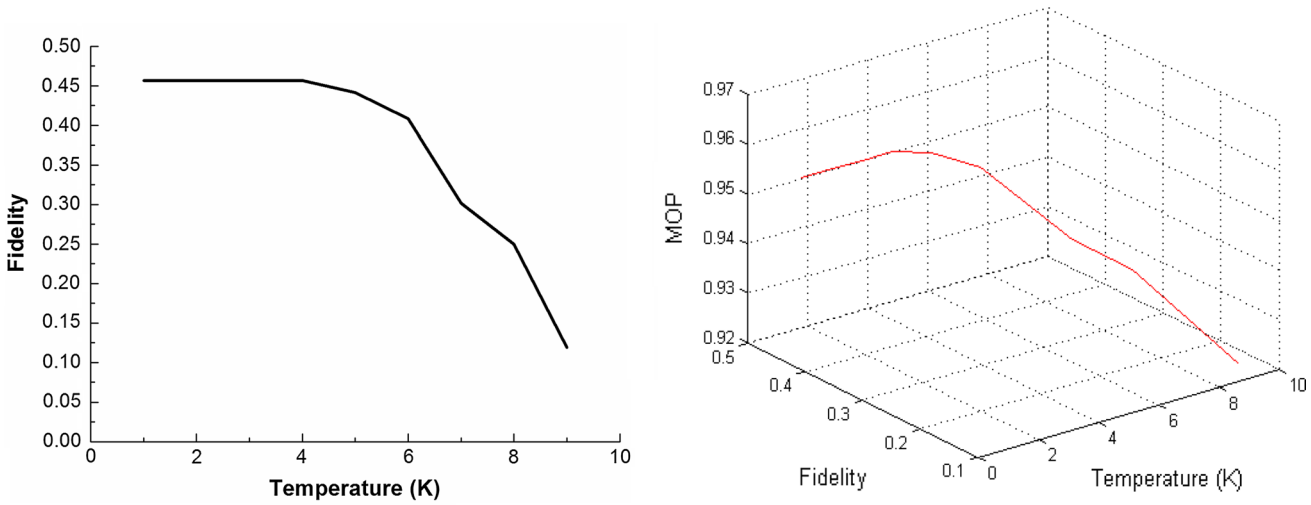

Fig. 8 Computational fidelity versus temperature characteristics of a 4-bit binary code-to-Gray code converter

Similarly, at different temperatures, the computational fidelity of the binary-to-excess-3 code converter is estimated and tabulated in Table 6.

4 Results and discussions

4.1 Fidelity versus temperature analysis

The variation of computational fidelity of the binary-to-Gray and binary-to-excess-3 code converter with

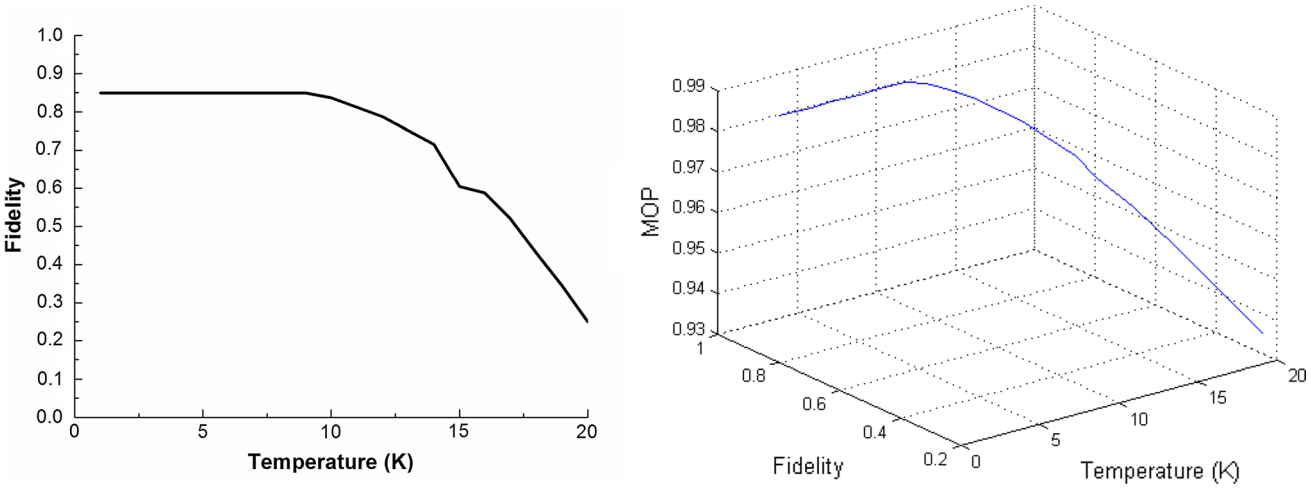

Fig. 9 Computational fidelity versus temperature characteristics of a 4-bit binary-to-excess-3 code converter

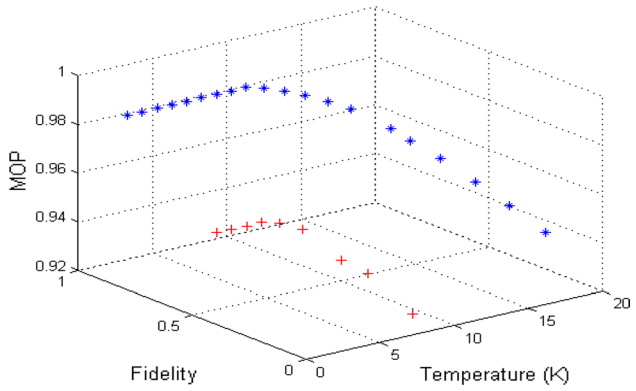

Fig. 10 Comparison of computational fidelity versus temperature characteristics of a 4-bit binary-to-Gray ("+" symbol) and binary-to-excess-3 code ("*" symbol) converters

temperature is shown in Figs. 8 and 9. The observations from the graphs are as follows.

1. Mop decreases by raising temperature. Thus, the computational fidelity decreases with increasing temperature for both the code converter circuits.
2. Both the circuits perform reliable computation over the low temperature range, i.e., stability of the code converter circuits decreases under thermal randomness.

4.2 Comparative analysis

The comparative study of the computational fidelity for both the code converter circuits is shown in Fig. 10 which shows results as follows.

1. The fidelity of the binary code-to-Gray code converter is much less than that of the binary-to-excess-3 code converter even in the low-temperature regime.
2. Further, the fidelity of the binary-to-excess-3 code converter starts to degrade at a relatively higher temperature range than that of the other converter.
3. The binary-to-Gray code and binary-to-excess-3 code converters can perform logical transformation or computation efficiently over the temperature range 1–4 K and 1–11 K, respectively.

The fidelity of the binary code-to-Gray code converter is much less than that of the binary-to-excess-3 code converter even in the low-temperature regime because of MOP. Higher value of MOP means higher fidelity as in that case $I(Z;Y)$ has maximum value of $H_L(Y)$. Section 3.2 shows that both the converters have same value of $H_L(Y)$, i.e., 4. Thus, the variation in fidelity depends on $I(Z;Y)$. If $I(Z;Y)$ increases, fidelity also increases, and if $I(Z;Y)$ decreases, fidelity also decreases. Now, the value of $I(Z;Y)$ depends on MOP.

Table 7 Computational faithfulness under thermal randomness

Degree of computational fidelity	Temperature (T) range (binary code-to-Gray code converter)	Temperature (T) range (binary code-to-excess-3 code converter)
Good	$0 \le T < 4$ K	$0 \le T < 11$ K
Adequate	5 K $\le T < 7$ K	11 K $\le T < 18$ K
Poor	7 K $\le T < 9$ K	18 K $\le T < 20$ K

Table 8 Data statistics of the code converters

	Y-axis (fidelity)	
	Binary code-to-Gray code converter	Binary code-to-excess-3 code converter
Min	0.1187	0.2497
Max	0.4567	0.8497
SD	0.1219	0.1915

So, if MOP increases, $I(Z;Y)$ also increases, and if MOP decreases, $I(Z;Y)$ also decreases. For example, as described in Sect. 3.2, at $T=1$ K, the MOP of binary code-to-Gray code converter is 0.954 which causes $I(Z;Y)=1.8094$ and thus fidelity $(F_{\mathrm{L}})=0.4567$. But in case of binary-to-excess-3 code converter, at $T=1$ K, the MOP is 0.988 which causes $I(Z;Y)=3.3963$ and thus fidelity $(F_{\mathrm{L}})=0.8497$. Therefore, due to lower MOP, the fidelity of the binary code-to-Gray code converter is much less than that of the binary-to-excess-3 code converter even in the low-temperature regime.

4.3 Computational faithfulness

During estimation process, the degree of computational faithfulness of proposed converters with temperature is observed and the result is plotted in Table 7. The result is analyzed as follows.

1. The computational fidelity of the QCA binary-to-Gray and binary-to-excess-3 code converters is good over the temperature range $0 \leq T < 4$ K and $0 \leq T < 11$ K, respectively. Thus, both the code converters have reliable computation over that range of temperatures.
2. Adequate over the range $5\,\mathrm{K} \leq T < 7\,\mathrm{K}$ and $11\,\mathrm{K} \leq T < 18\,\mathrm{K}$, respectively. Thus, the output from both of the code converters can be considered as valid outputs over that range of temperatures.
3. Poor over the range $7\,\mathrm{K} \leq T < 9\,\mathrm{K}$ and $18\,\mathrm{K} \leq T < 20\,\mathrm{K}$, respectively. Thus, both the code converters have faulty outputs.

4.4 Data statistics of the proposed code converter circuits

The data statistics for both the code converter circuits is tabulated in Table 8. Table 8 shows that the standard deviation of the computational fidelity of the binary code-to-excess-3 code converter is slightly larger than that of the binary-to-Gray code converter. This little difference of the order of approximately 0.07 is manifested in the small difference in the slope of the two curves (Fig. 10). The phenomenon reveals the performance of the binary-to-excess-3 code converter to be more reliable than the binary-to-Gray code converter under thermal randomness.

5 Conclusion

This article shows the computation of channel fidelity in QCA channel routing for noiseless and noisy QCA channel routing. Shannon's information-theoretic measure of computational fidelity confirms the robustness of the proposed binary-to-Gray and binary-to-excess-3 code converter-based QCA routing channels. The proposed routing channels yield reliable computation under certain range of temperatures. The computational fidelity is found to deteriorate with increasing temperature for both the routing channels. This routing channels exhibit considerable fidelity when operated in the temperature regime 1–5 K and 1–11 K, respectively. Hence, both the channels yield appreciable computational efficacy over the low-temperature regime. Moreover, the extent of variation of the computational fidelity of the circuits with the thermal fluctuations reflects the fuzzy multivalued status of the performance of the QCA-based routing channels. The simulation result is verified through theoretic values that agreed the design accuracy of the proposed channels.

Acknowledgements The authors are grateful to DST FIST Project, WBUT, India, for providing with the grant for accomplishment of the project under File No. SR/FST/ETI-296/2011, and TEQIP II, WBUT, India.

References

1. Lent, C.S., Tougaw, P.D., Porod, W., Bernstein, G.H.: Quantum cellular automata. Nanotechnology **4**, 49–57 (1993)
2. Porod, W.: Quantum-dot devices and quantum-dot cellular automata. Int. J. Bifurc. Chaos **7**, 2199–2218 (1997)
3. Tougaw, P.D., Lent, C.S.: Logic devices implemented using quantum cellular automata. J. Appl. Phys. **75**, 1818–1825 (1994)
4. Orlov, A.O., Amlani, I., Toth, G., Lent, C.S., Bernstein, G.H., Snider, G.L.: Experimental demonstration of a binary wire for quantum-dot cellular automata. Appl. Phys. Lett. **74**, 2875–2877 (1999)
5. Amlani, I., Orlov, A.O., Snider, G.L., Lent, C.S., Porod, W., Bernstein, G.H.: Experimental demonstration of electron switching in a quantum-dot cellular automata (QCA) cell. Superlattices Microstruct. **25**, 273–278 (1999)
6. Campos, C.A.T., Marciano, A.L., Vilela Neto, O.P., Torres, F.S.: USE: a universal, scalable and efficient clocking scheme for QCA. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **35**, 513–517 (2016)
7. Pudi, V., Sridharan, K.: A bit-serial pipelined architecture for high-performance DHT computation in quantum-dot cellular automata. IEEE Trans. Very Large Scale Integr. Syst. **23**, 2352–2356 (2015)

8. Das, J.C., De, D.: Operational efficiency of novel SISO shift register under thermal randomness in quantum-dot cellular automata design. Microsyst. Technol. (2016). https://doi.org/10.1007/s00542-016-3085-y
9. Sen, B., Nag, A., De, A., Sikdar, B.K.: Towards the hierarchical design of multilayer QCA logic circuit. J. Comput. Sci. **11**, 233–244 (2015)
10. Sen, B., Dutta, M., Mukherjee, R., Nath, R.K., Sinha, A.P., Sikdar, B.K.: Towards the design of hybrid QCA tiles targeting high fault tolerance. J. Comput. Electron. **15**(2), 429–445 (2016)
11. Das, K., De, D., De, M.: Realization of semiconductor ternary quantum dot cellular automata. IET Micro Nano Lett. **8**, 258–263 (2013)
12. Rashidi, H., Rezai, A., Soltany, S.: High-performance multiplexer architecture for quantum-dot cellular automata. J. Comput. Electron. **15**(3), 968–981 (2016)
13. Das, J.C., De, D.: Reversible comparator design using quantum dot-cellular automata. IETE J. Res. **62**, 323–330 (2016)
14. Roy, D., Maitra, S., Mukherjee, K., De, D.: Analysis of effect of temperature variation on computational faithfulness of a QCA XOR gate. Presented at International Conference on Electronics, Communication and Instrumentation (ICECI), Kolkata, India, pp. 1–4 (2014)
15. K. Das, D. De, M. De, Re-Programmable Logic Array for Logic Design and Its Reliability Analysis in QCA, in Emerging Trends in Computing and Communication (ETCC), Lecture Notes in Electrical Engineering, Springer, 298 (2014) 341-352
16. Roohi, A., Khademolhosseini, H., Sayedsalehi, S., Navi, K.: A symmetric quantum-dot cellular automata design for 5-input majority gate. J. Comput. Electron. **13**, 701–708 (2014)
17. Kianpour, M., Sabbaghi-Nadooshan, R., Navi, K.: A novel design of 8-bitadder/subtractor by quantum-dot cellular automata. J. Comput. Syst. Sci. **80**, 1404–1414 (2014)
18. Tehrani, M.A., Safaei, F., Moaiyeri, M.H., Navi, K.: Design and implementation of multistage interconnection networks using quantum-dot cellular automata. Microelectron. J. **42**, 913–922 (2011)
19. Ahmad, F., Bhat, G.M., Khademolhosseini, H., Angizi, S.A.S., Navi, K.: Towards single layer quantum-dot cellular automata adders based on explicit interaction of cells. J. Comput. Sci. **16**, 8–15 (2016)
20. Perez-Martinez, F., Petersson, K.D., Farrer, I., Anderson, D., Jones, G.A.C., Ritchie, D.A., Smith, C.G.: Realization of a GaAs/AlGaAs-based quantum cellular automata cell. Microelectron. J. **39**, 674–677 (2008)
21. Macuccia, M., Gattobigio, M., Bonci, L., Iannaccone, G., Prins, F.E., Single, C., Wetekamb, G., Kern, D.P.: A QCA cell in silicon on insulator technology: theory and experiment. Superlattices Microstruct. **34**, 205–211 (2003)
22. Farazkish, R.: A new quantum-dot cellular automata fault-tolerant full-adder. J. Comput. Electron. **14**(2), 506–514 (2015)
23. Vankamamidi, V., Ottavi, M., Lombardi, F.: A serial-memory by quantum-dot cellular automata. IEEE Trans. Comput. **57**, 606–618 (2008)
24. Das, J.C., De, D.: Novel low power reversible binary incrementer design using quantum-dot cellular automata. Microprocess. Microsyst. **42**, 10–23 (2016)
25. Kassa, S.R., Nagaria, R.K.: A novel design of quantum dot cellular automata 5-input majority gate with some physical proofs. J. Comput. Electron. **15**, 324–334 (2016)
26. Das, J.C., De, D.: Quantum dot-cellular automata based reversible low power parity generator and parity checker design for nanocommunication. Front. Inform. Technol. Electron. Eng. **17**, 224–236 (2016)
27. Hashemi, S., Navi, K.: New robust QCA D flip flop and memory structures. Microelectron. J. **43**, 929–940 (2012)
28. Du, H., Lv, H., Zhang, Y., Peng, F., Xie, G.: Design and analysis of new fault-tolerant majority gate for quantum-dot cellular automata. J. Comput. Electron. **15**(4), 1484–1497 (2016)
29. Arjmand, M.M., Soryani, M., Navi, K.: Coplanar wire crossing in quantum cellular automata using a ternary cell. IET Circuits Dev. Syst. **7**, 263–272 (2013)
30. Sayedsalehi, S., Azghadi, M.R., Angizi, S., Navi, K.: Restoring and non-restoring array divider designs in quantum-dot cellular automata. Inf. Sci. **311**, 86–101 (2015)
31. Fijani, A., Toomarian, B.N.: New design for quantum dots cellular automata to obtain fault tolerant logic gates. J. Nanopart. Res. **3**, 27–37 (2001)
32. Farazkish, R., Khodaparast, F.: Design and characterization of a new fault-tolerant full-adder for quantum-dot cellular automata. Microprocess. Microsyst. **39**, 426–433 (2015)
33. Ma, X., Huang, J., Metra, C., Lombardi, F.: Detecting multiple faults in one-dimensional arrays of reversible QCA gates. J. Electron. Test. **25**, 39–54 (2009)
34. Sen, B., Dutta, M., Sikdar, B.K.: Efficient design of parity preserving logic in quantum-dot cellular automata targeting enhanced scalability in testing. Microelectron. J. **45**, 239–248 (2014)
35. Anderson, N.G., Maalouli, F., Mestancik, J.: Quantifying the computational efficacy of nanocomputing channels. Nano Commun. Netw. **3**, 139–150 (2012)
36. Gallager, R.G.: Information Theory and Reliable Communication. Wiley, New York (1968)