Microprocessors and Microsystems 84 (2021) 104237

Contents lists available at ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro

QCA-based Hamming code circuit for nano communication network

Jianhong Huang, Guangjun Xie, Rui Kuang, Feifei Deng, Yongqiang Zhang *

School of Electronic Science and Applied Physics, Hefei University of Technology, Hefei, 230009, China

ARTICLE INFO

Keywords:
Hamming code
Error-detecting
Error-correcting
Quantum-dot cellular automata
Nano communication

ABSTRACT

Quantum-dot cellular automata (QCA) is one of the most promising technologies for designing digital circuits with high integration, ultra-low energy consumption, and high switching speed. The efficient implementation based on QCA has been verified in some nanocommunication code circuits. In this manuscript, a bottom-up method is adopted to design and implement novel 3–8 and 4–16 decoders. Based on the proposed structure, (7,4) and (15,11) Hamming codec circuits are implemented. The general design method of QCA Hamming code communication network is also given. For different generator matrices in Hamming codes, the proposed design only needs to modify the corresponding interconnection lines in the communication network without major adjustments to the circuit structure to achieve the same function. Compared with the Hamming code related designs in other studies, the proposed design has better hardware complexity. The proposed Hamming code circuit has a great performance improvement in terms of cell number, area, and clock delay. The above designs have been functionally verified by the QCADesigner.

1. Introduction

CMOS technology is approaching its physical limits, and the problems such as quantum effect and power dissipation have become more prominent. When CMOS technology shrinks to the nanometer level, the design complexity is even higher. In order to continue to shrink circuits and improve the performance of microprocessors, CMOS alternatives are indispensable. As technology advances, new nanoscale computing designs are emerging. Quantum-dot cellular automata (QCA) is a research topic that has the potential to design new hardware components. It is a nano-level technology in which the cell has nano-size, ultra-low power consumption, and high clock rate [1-3]. A very useful feature of QCA is that it uses the Coulomb interaction between electrons to perform calculations instead of current, which makes it an ultra-low power implementation technology to avoid the problem of leakage current. Compared with CMOS, it also provides faster switching speed and higher device density.

Using QCA to realize nano communication network has broad prospects in the application field. For the further development of nano computing, nano communication circuits should have the ability to detect and correct errors [4]. Hamming code plays an important role in the error detection and correction of communication circuits. It is a linear debugging code in the telecommunication field [5]. Data bit errors may occur when a computer stores or moves data, Hamming code detects and corrects the single-bit error by inserting supervision bits into the transmitted information stream. This manuscript implements the codec circuit of (7, 4) and (15, 11) Hamming code based on QCA. It can not only verify whether the data is valid but also indicate the error position and correct it when single-bit of data is wrong. The main contributions of this paper are as follows.

(1) The low-power design and implementation of Hamming code based on QCA are given. The design adopts the bottom-up design method, and the implementation adopts the multi-layered crossing method [6], which has good stability and a small area.
(2) A novel 3–8 and 4–16 decoder based on QCA is implemented. The proposed design occupies a small area, has a fast response speed, not only has good regularity and scalability, but also has high signal strength, and the I/O ports are easy to access, very suitable as the basic unit of other complex circuits.
(3) A complete QCA Hamming communication network is constructed based on the proposed Hamming code component, which can be well compatible with different generator matrices, and at the same time ensures that the information can be identified and corrected when errors occur in the transmission process.

* Corresponding author.
E-mail address: ahzhangyq@hfut.edu.cn (Y. Zhang).

https://doi.org/10.1016/j.micpro.2021.104237
Received 19 November 2020; Received in revised form 19 February 2021; Accepted 22 February 2021
Available online 24 February 2021

(4) The function of the proposed design was verified using QCADesigner [7], which proved the feasibility of the design.
(5) The proposed design is analyzed and evaluated based on area, delay, accessibility of I/O ports, and the number of logic gates.

The rest paper is organized as follows. Section 2 introduces the application of QCA technology in communication and related fields. Section 3 discusses the related principles of parity check and Hamming code. Section 4 presents the QCA-based Hamming code circuit and its communication network. Section 5 shows the simulation results of the proposed design using QCADesigner. Section 6 analyzes and compares the proposed designs. Section 7 concludes this manuscript.

2. Related works

At present, the research of QCA in the encryption circuit and communication network is deepening. Sardinha et al. [8] proposed a nano-router circuit composed of external components such as crossbar switch, demultiplexer, and parallel-to-serial converter, and described in detail the efficiency of the nano-router in information routing. Das et al. designed a nano-polar encoder circuit in [9]. The stuck-at-fault effect in the generation of effective polarity code is discussed, and the test vector group with 100% fault coverage is proposed. Das et al. [10] proposed a reversible QCA 2 × 2 crossbar switch based on Fredkin and studied the influence of stuck-at-fault on the designed crossbar switch and nano communication circuit. By considering a single missing & additional cell, the defects in the proposed QCA crossbar switch are identified and explored. Das et al. designed and implemented a 4 × 4 router that can transfer data from four different input links to four different output links in [11]. The design adopts a single-layer design, and its area and number of cells are greatly improved compared with similar routers, with less power consumption and lower clock delay. Cesar et al. designed a QCA single byte correction circuit based on the Reed-Solomon code in [12]. To design all the parts, the author designs building blocks, such as the word inverter, the pulse generator, the syndrome calculator, etc., which can be used in future works and even on designs that are not directly associated with communication systems. Norouzi et al. [13] proposed a reversible parity generator and a parity checker composed of the Feynman gate and TIEO gate. This design uses a single-layer structure, which has a smaller area and delay compared with similar designs. Das et al. proposed a novel single-layer design of circuit switch networks based on QCA in [14]. The design is implemented using a crossbar switch, multiplexer, and demultiplexer. All these designs have low energy consumption and high device density. Based on the input-output plug and plug fault effect, it provides the basis for the fault-free design of the crossbar switch circuit. The fault-tolerant nature of the designs is established by exploring and analyzing the effect on communication due to stuck-at-fault at the control signals. Debnath et al. [15] proposed an architecture using QCA technology to perform image steganography. In order to encode information in an image, an encoder/decoder steganography circuit based on the least significant bit (LSB) is proposed.

3. Coding principle

3.1. Parity check

When transferring binary data, errors may occur. One way to detect errors is to add extra check bits to the binary data, such as parity bits. The parity check is performed according to whether the number of "1″s in a set of transmitted binary codes is odd or even. The odd number is called odd parity, otherwise, it is called even parity. What kind of parity should be specified in advance. Since parity bit cannot determine which bit is wrong, it cannot correct the error code. But the parity bit also has its advantages, it is the best check code that can be achieved with one bit of data, and it only needs some XOR gates to generate it.

3.2. Hamming communication network

Fig 1 shows the overall block diagram of the Hamming communication network. In the communication network, the message $u = (u_1, u_2, \cdots, u_k)$ will be encoded as a code word $b = (b_1, b_2, \cdots, b_k)$. For the Hamming communication network, the code word satisfies

$$\boldsymbol{b} = \boldsymbol{uG} = \boldsymbol{u}\left(\boldsymbol{I}_k \middle| \boldsymbol{A}_{k,n-k}\right) = \left(\boldsymbol{u} \middle| \boldsymbol{uA}_{k,n-k}\right) \tag{1}$$

where $\boldsymbol{G}$ is a$k \times k$generator matrix, $\boldsymbol{A}_{k,n-k}$is some fixed $k \times (n-k)$ matrix of 0 and 1, and I_k is a $k \times k$ unit matrix. Meanwhile, the dimensions of the message and codeword must satisfy

$$\begin{cases} n = 2^r - 1 \\ k = 2^r - r - 1 \end{cases} \left(r \geq 2\right) \tag{2}$$

where r is the number of parity checks.

Then the codeword is sent to the channel. Because of channel noise, the received vector $\boldsymbol{r} = (r_1, r_2, \cdots, r_k)$ may be different from b. Then the received vector is decoded into $\widehat{b}$ by the decoder which carries out syndrome decoding as the dashed rectangle shows. The detailed process of decoder consists of (1) error checking - to compute the syndrome vector $\boldsymbol{s}$ according to

$$\boldsymbol{s}^T = \boldsymbol{Hr}^T \tag{3}$$

where $\boldsymbol{H}$ is the parity check matrix and it follows

$$\boldsymbol{HG}^T = 0 \tag{4}$$

and then to obtain the error vector $\widehat{e}$ from the look-up table according to the syndrome vector $\boldsymbol{s}$; and (2) error correcting - to obtain the decoded codeword $\boldsymbol{b}$ as

$$\widehat{\boldsymbol{b}} = \boldsymbol{r} \oplus \widehat{\boldsymbol{e}} \tag{5}$$

3.3. Hamming code

Suppose the code group is (a_{n-1}, a_{n-2} … a_1, a_0), where the first n-1 bits are the information bits, and the last bit a_0 is the supervision bit. If a_0 is added to make the number of "1″s in the code group even (or odd), it is satisfied

$$a_{n-1} \oplus a_{n-2} \oplus \cdots \oplus a_0 = 0 \tag{6}$$

It is called even supervision code (or odd supervision code). In the Hamming code, r parity bits are added to k bits of information to form a code group with a length of n ($n = k + r$) bits. Taking even parity as an example, it uses one bit of supervision symbol and information symbol to form an algebraic formula. When decoding at the receiving end, it actually calculates the received code group as follows:

$$S = a_{n-1} \oplus a_{n-2} \oplus \cdots \oplus a_0 \tag{7}$$

If $S = 0$, it is considered that there is no error code; if $S = 1$, it is considered that there is an error code. Eq. (7) is called the supervision relational expression, and S is called the syndrome.

Since the syndrome S has two values, it can only represent two states of error and error-free, but cannot indicate the location of the error code. To obtain an (n, k) code with error correction capability, the number of supervision bits needs to be increased. If the supervision bit is increased by one bit, a supervision relational expression similar to Eq. (7) can be added.

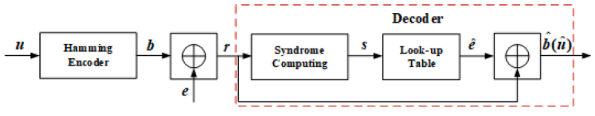

Fig. 1. Block diagram of an overall Hamming communication network.

Suppose $k = 4$ in the block code (n, k). To correct the single-bit error code, it can be seen from Eq. (2) that the number of supervision bits is $r \geq 3$. If $r = 3$, then $n = k + r = 7$. We use $a_6\ a_5 \ldots a_0$ to represent these 7 symbols, and use S_1, S_2, and S_3 to represent the syndromes in the three supervision relations. Then the corresponding relationship between the values of S_1, S_2, and S_3 and the position of the error code can be specified as shown in Table 1. According to the regulations in the table, only when the position of the single-bit error code is a_2, a_4, a_5 or a_6, the syndrome S_1 is 1, otherwise, S_1 is 0. This means that the 4 symbols a_2, a_4, a_5, and a_6 constitute an even supervision relationship:

$$S_1 = a_6 \oplus a_5 \oplus a_4 \oplus a_2 \tag{8}$$

Similarly

$$S_2 = a_6 \oplus a_5 \oplus a_3 \oplus a_1 \tag{9}$$

$$S_3 = a_6 \oplus a_4 \oplus a_3 \oplus a_0 \tag{10}$$

When encoding at the transmitter, the supervision bits a_2, a_1, and a_0 should be determined by the supervision relationship according to the value of the information bits, that is, the supervision bits should be such that the values of S_1, S_2, and S_3 in Eq. (8)-(10) are all 0.

$$\begin{cases} a_6 \oplus a_5 \oplus a_4 \oplus a_2 = 0 \\ a_6 \oplus a_5 \oplus a_3 \oplus a_1 = 0 \\ a_6 \oplus a_4 \oplus a_3 \oplus a_0 = 0 \end{cases} \tag{11}$$

After shifting the calculation, the supervision bit is solved as:

$$\begin{cases} a_2 = a_6 \oplus a_5 \oplus a_4 \\ a_1 = a_6 \oplus a_5 \oplus a_3 \\ a_0 = a_6 \oplus a_4 \oplus a_3 \end{cases} \tag{12}$$

After the information bits are given, the supervision bits can be calculated according to (12), and the results are shown in Table 2.

4. QCA implementation of Hamming communication network

4.1. Basic logic gate

The XOR gate is one of the most important gates in QCA [15-20]. A large number of XOR gates are used in the Hamming code. However, these existing XOR gate designs have disadvantages such as a large number of cells, large delay, difficult access to I/O ports, and complex structures that are not conducive to expansion. Ahmad et al. [21] used a device similar to a 5-input majority gate to implement a 3-input XOR gate. As shown in Fig. 2(a), the proposed design has ultra-high speed and low complexity, only contains 14 QCA cells and 0.5 clock cycle delay. The author named it TIEO (Three-Input Exclusive-Or) gate. The TIEO gate is not only simple in structure, less in the number of cells, and small in area, but another outstanding advantage is its strong scalability, which is reflected in if one input of the TIEO gate is fixed to logic 0, as shown in Fig 2(b), it can be implemented as a 2-input XOR gate. If the driving cells on both sides of the TIEO gate are fixed to logic 0, as shown in Fig 2(c), it can be implemented as a 3-input AND gate. It is accurate because of these advantages that the TIEO gate is widely used in the design of various QCA circuits. Expand the input port of the TIEO gate, and fix the values of several of the inputs to logic 0, as shown in Fig 3, which can be implemented as a 4-input AND gate. In this manuscript,

Table 1
The relationship between syndrome and error code position of (7, 4) Hamming code.

$S_1\ S_2\ S_3$	Error code position	$S_1\ S_2\ S_3$	Error code position
001	a_0	100	a_4
010	a_1	110	a_5
100	a_2	111	a_6
011	a_3	000	–

Table 2
The relationship between data bits and parity bits.

Data bit $a_6\ a_5\ a_4\ a_3$	Parity bit $a_2\ a_1\ a_0$	Data bit $a_6\ a_5\ a_4\ a_3$	Parity bit $a_2\ a_1\ a_0$
0000	000	1000	111
0001	011	1001	100
0010	101	1010	010
0011	110	1011	001
0100	110	1100	001
0101	101	1101	010
0110	011	1110	100
0111	000	1111	111

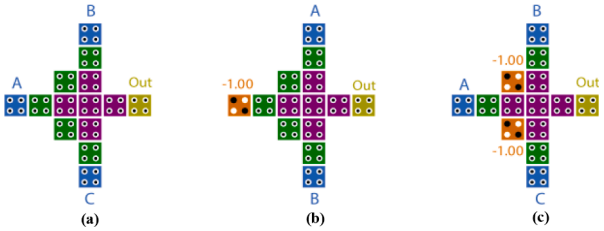

Fig. 2. (a) TIEO gate. (b) Two-input XOR gate. (c) Three-input AND gate.

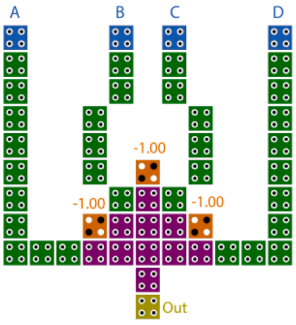

Fig. 3. A four-input AND gate.

the Hamming code circuit and 3–8 decoder are implemented based on the TIEO gate and its variants, and the 4–16 decoder is implemented based on the proposed four-input AND gate, which greatly reduces the number of cells in the circuit and reduce the area and delay of the circuit.

4.2. Implementation of (7, 4) Hamming communication network

In the (7, 4) Hamming encoder, the four information bits can be directly output without any changes, while the three supervision bits can be obtained by Eq. (12), which means that three 3-input XOR gates are needed in the QCA circuit, and the circuit diagram of the encoder can be obtained as shown in Fig 4. Applying the 3-input XOR gate proposed in [14] to the encoder, the QCA implementation diagram shown in Fig 5 can be obtained. This layout uses a multi-layered crossing method to

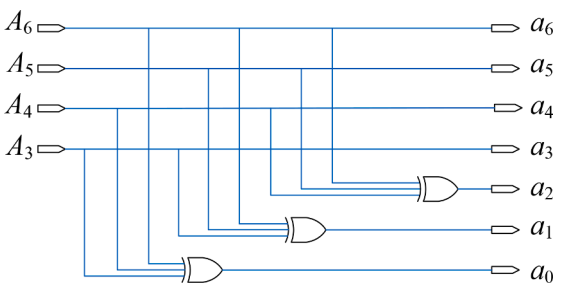

Fig. 4. (7, 4) Hamming encoder circuit.

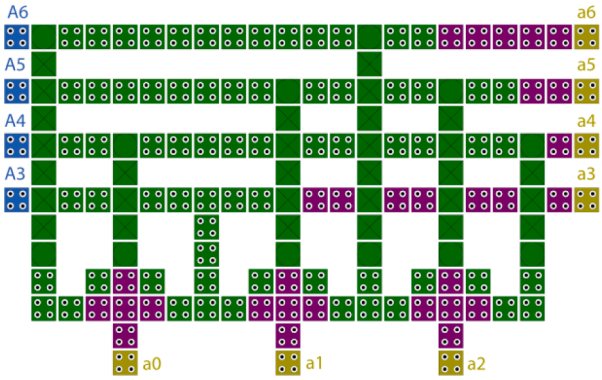

Fig. 5. QCA circuit diagram of (7, 4) Hamming encoder.

further reduce the area of the circuit. It only consumes an area of 0.13 μm^2 and 188 cells. After 0.5 clock cycle delay, the coded code group can be obtained. The encoder has a simple structure and a very fast response.

To implement the Hamming decoder, three syndromes must be obtained first. From Eqs. (8), (9), (10), it can be seen that the syndrome can be obtained by a 4-input XOR gate. According to the value of the 3-bit syndrome, it is possible to determine whether there is an error and the position of the error, but the error cannot be corrected now. In the digital circuit, XOR with "0″ is itself, and XOR with "1″ is the reverse. The correct information bits can be obtained by XOR with "1″ and the error code. Therefore, the 3-bit syndrome is connected to the input of the 3–8 decoder, and then the output set to "1″ is XOR with the corresponding error code to obtain the correct code group. Table 3 shows the relationship between syndrome, error position and the value of the 3–8 decoder. There are many QCA implementations of 3–8 decoders, but most of the I/O ports of these designs are inside the gate unit, surrounded by other cells, so it is difficult to access, which makes it impossible to use widely and as a basic component to design more complex circuits. Based on the 3-input AND gate shown in Fig 2(c), the QCA 3–8 decoder shown in Fig 6 is proposed. Compared with the previous 3–8 decoder, the proposed design occupies a small area and has a fast response speed. It not only has good regularity and scalability, but also has high signal strength and is easy to be used as the basic unit of other complex circuits.

According to the above analysis, the circuit diagram of the (7, 4) Hamming decoder is shown in Fig 7. The proposed 3–8 decoder is applied to the Hamming decoder, and the QCA implementation diagram of the (7, 4) Hamming decoder as shown in Fig 8 is obtained. The information bits first perform 3-input XOR according to the supervision relation (12), and then performs 2-input XOR with the corresponding supervision bit to obtain the 3-bit syndrome. The 3-bit syndrome is used as the input of the 3–8 decoder, and the output of the 3–8 decoder is XOR with the corresponding error code to obtain the correct code group A_6, A_5, A_4, A_3. Since there are only 4 information bits, only four outputs of the 3–8 decoder shown in Table 3 are required, and the other outputs can be omitted. This design consumes an area of 0.45 μm^2 and 593 cells. After 0.75 clock cycle delay, it can be detected which position has an error code, and after 1.75 clock cycles delay, the correct group code can be obtained.

Table 3
The relationship between syndrome, error code position and 3–8 decoder.

$S_1\ S_2\ S_3$	Error code position	$D_0\ D_1\ D_2\ D_3\ D_4\ D_5\ D_6\ D_7$	Correct data bit
000	–	10,000,000	–
001	a_0	01,000,000	–
010	a_1	00,100,000	–
011	a_3	00,010,000	$a_3 \oplus D_3$
100	a_2	00,001,000	–
101	a_4	00,000,100	$a_4 \oplus D_5$
110	a_5	00,000,010	$a_5 \oplus D_6$
111	a_6	00,000,001	$a_6 \oplus D_7$

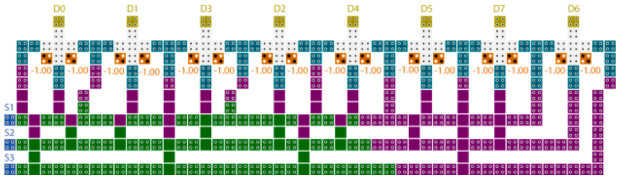

Fig. 6. QCA circuit diagram of 3–8 decoder.

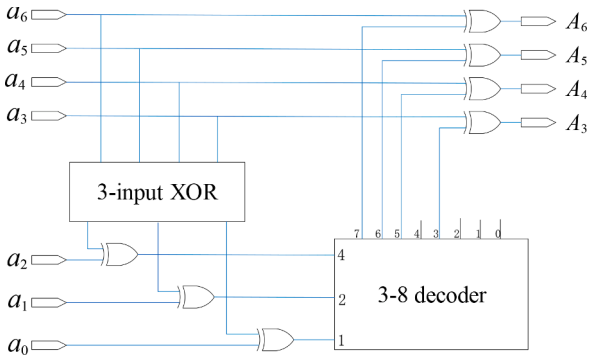

Fig. 7. (7, 4) Hamming decoder circuit.

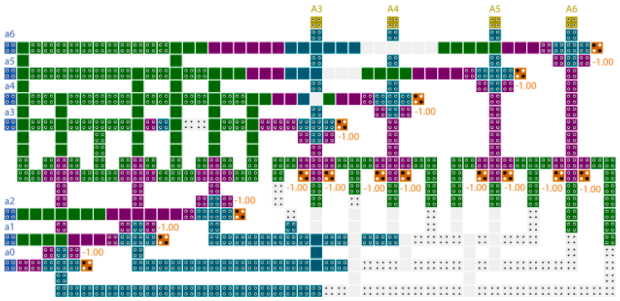

Fig. 8. QCA circuit diagram of (7, 4) Hamming decoder.

Fig 9 shows the QCA Hamming code network obtained from the overall block diagram of the Hamming communication network. It mainly consists of three parts: Hamming encoder, communication channel, and Hamming decoder. The decoder includes a syndrome calculator and a look-up table. The communication channel can not only transmit information, but also process information, depending on actual demand. When the information passes through the channel, errors may occur. The decoder can identify and correct the errors that occur at a single location, which improves the error tolerance of the information during transmission. The biggest advantage of the proposed communication network design is that it has good adaptability. For different generator matrices, it is only necessary to modify the interconnection lines in the circuit according to the supervisory relationship without changing the circuit structure. As far as the look-up table, it is essentially a part of the 3–8 decoder. When the generator matrix is different, it is only necessary to select the appropriate 3–8 decoder output to satisfy the circuit function. The entire communication network structure is simple, the number of gates used and the occupied area are less, and the response speed is fast.

4.3. Extension of QCA Hamming code

The basic principle of the Hamming code is to link the information code element and the supervision code element through a linear equa-

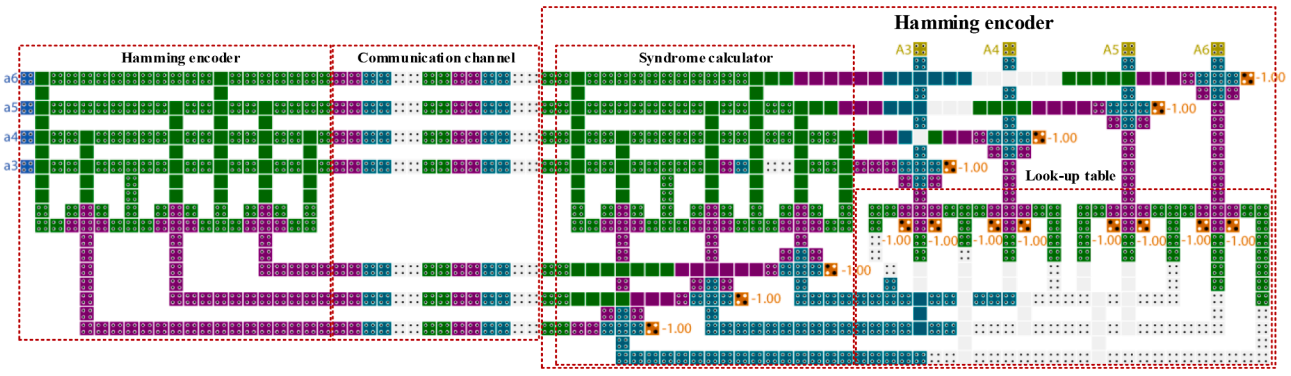

Fig. 9. QCA implementation of Hamming communication network.

tion, and each supervision bit is encoded in a specific position of the transmission code word. The system can separate the wrong bit, whether it is in the original information bits or the additional supervision bits. Therefore, for a multi-bit Hamming code, the codec circuit can be easily obtained by finding the supervision relationship. Table 4 shows the relationship between syndrome and error code position in (15, 11) Hamming code. From the table, it can be concluded that the supervision relation is:

$$\begin{cases} S_1 = a_3 \oplus a_7 \oplus a_8 \oplus a_9 \oplus a_{11} \oplus a_{12} \oplus a_{13} \oplus a_{14} \\ S_2 = a_2 \oplus a_5 \oplus a_6 \oplus a_9 \oplus a_{10} \oplus a_{12} \oplus a_{13} \oplus a_{14} \\ S_3 = a_1 \oplus a_4 \oplus a_6 \oplus a_8 \oplus a_{10} \oplus a_{11} \oplus a_{13} \oplus a_{14} \\ S_4 = a_0 \oplus a_4 \oplus a_5 \oplus a_7 \oplus a_{10} \oplus a_{11} \oplus a_{12} \oplus a_{14} \end{cases} \tag{13}$$

According to the supervision relation, the (15, 11) Hamming encoder shown in Fig 10 can be obtained. The 4-bit check code $a_3\ a_2\ a_1\ a_0$ is derived from the 7-input XOR of the information bits according to the supervision relationship. The 7-input XOR gate is implemented by cascading three 3-input XOR gates. The structure of (15, 11) Hamming encoder is similar to (7, 4) Hamming encoder, for different generator matrices, the same function can be realized only by changing the interconnection line. This design uses 1508 cells, takes up an area of 0.95 μm^2, and has a delay of 2 clock cycles.

Because the (15, 11) Hamming code has 15 bits in total, including 11 information bits and 4 check bits. If only a single-bit error is detected, there are 16 states (including the state where no error occurs), so a 4–16 decoder is needed as a look-up table to correct the error. Each output unit of the 4–16 decoder is composed of a 4-input AND gate. Applying the 4-input AND gate mentioned in Fig 3 to the QCA implementation of the 4–16 decoder, the circuit diagram in Fig 11 is obtained. The proposed 4–16 decoder has a regular structure and easy access to I/O ports. The 4–16 decoder uses a total of 1417 cells, occupies an area of 1.13 μm^2, and has a delay of 1.75 clock cycles.

Similarly, according to the supervision relation, the syndrome of (15, 11) Hamming code is first obtained, and then the error code at the corresponding position is corrected through the look-up table to obtain the (15, 11) Hamming decoder shown in Fig 12. The entire decoder uses a total of 3904 cells, occupying an area of 3.35 μm^2, and has a delay of 4.5 clock cycles.

Table 4
The relationship between syndrome and error code position of (15, 11) Hamming code.

$S_1\ S_2\ S_3\ S_4$	Error code position	$S_1\ S_2\ S_3\ S_4$	Error code position
0001	a_0	1010	a_8
0010	a_1	1100	a_9
0100	a_2	0111	a_{10}
1000	a_3	1011	a_{11}
0011	a_4	1101	a_{12}
0101	a_5	1110	a_{13}
0110	a_6	1111	a_{14}
1001	a_7	0000	–

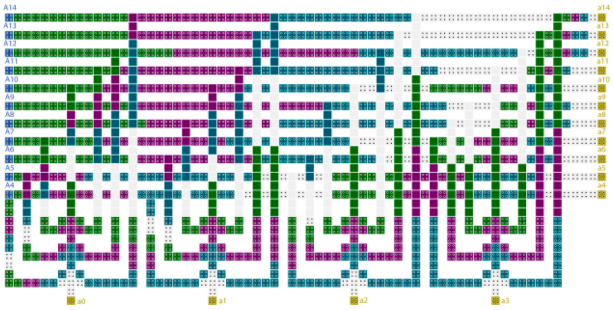

Fig. 10. QCA circuit diagram of (15, 11) Hamming encoder.

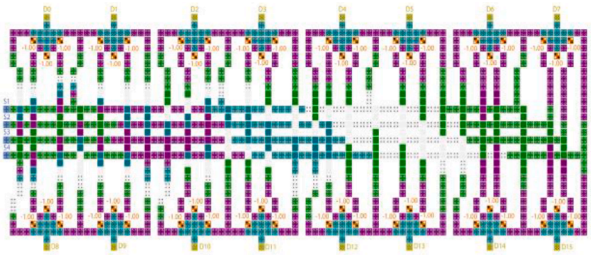

Fig. 11. QCA circuit diagram of 4–16 decoder.

5. Simulation results

All the designs proposed in this manuscript have been verified using the simulation tool QCADesigner 2.0.3, and the bistable approximation engine is selected to obtain the simulation waveform. The simulation result of the (7, 4) Hamming encoder is shown in Fig 13. Combined with Table 2, it can be seen that the output a_6, a_5, a_4, and a_3 are consistent with the input, while a_2, a_1, and a_0 are respectively obtained according to the supervision relationship, which is consistent with the data in the table, indicating that the encoder has successfully realized the expected functions. The simulation result of the 3–8 decoder is shown in Fig 14, where D Bus is composed of the output D_0, D_1, D_2, D_3, D_4, D_5, D_6, D_7 of the 3–8 decoder. Since only one bit in D Bus is "1", its value is a power of 2. It can be seen from Fig 14 that the correct waveform is obtained after a delay of a 1.00 clock cycle. The simulation result of the 4–16 decoder is shown in Fig 15. Similarly, D Bus is a combination of 16 outputs of the 4–16 decoder, and the correct waveform is obtained after a delay of 1.75 clock cycles. Select the test vector as in Table 5 to test the correctness of the (7, 4) Hamming decoder. In the test vector, there are five error-free code groups at first, and their output should be consistent with the input

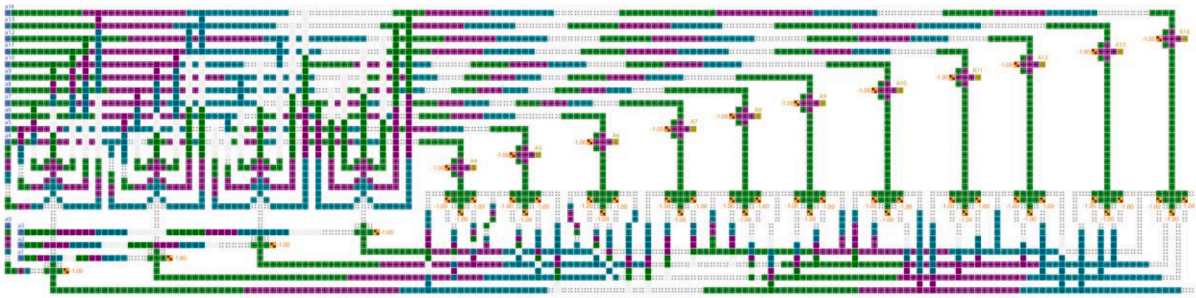

Fig. 12. QCA circuit diagram of (15, 11) Hamming decoder.

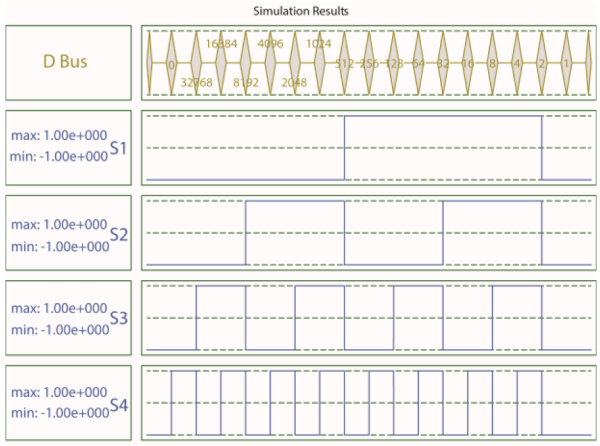

Fig. 13. Simulation waveform of 4–16 decoder.

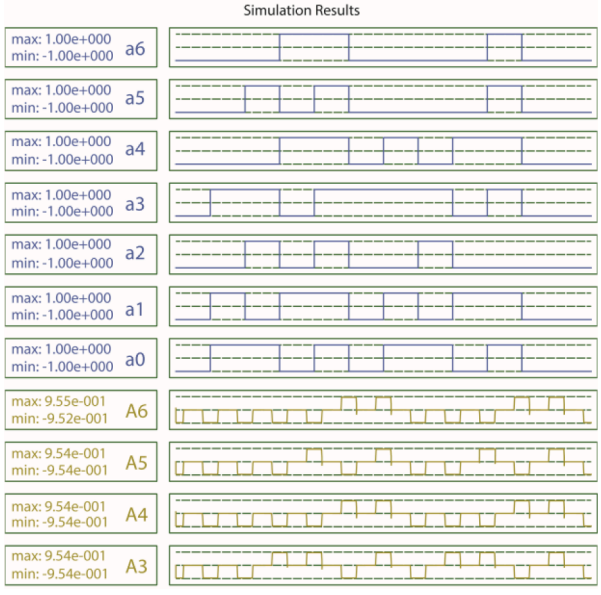

Fig. 14. Simulation waveform of (7, 4) Hamming decoder.

information bits. Then there are five code groups each containing a single-bit error code, their error code positions are respectively a_3, a_4, a_5, a6, a_6, the output of the first four code groups should be the same as the input that corrected the corresponding error code, the output of the fifth code group is the same as the input information bits, because its error code occurs on the check bit, so it does not need to be corrected. The simulation result is shown in Fig 16. After the 1.75 clock cycles delay, the waveform is the same as the data in the table. It can be seen

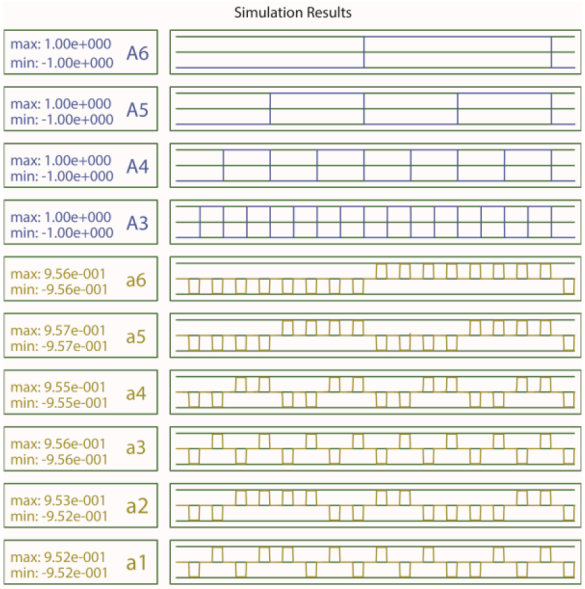

Fig. 15. Simulation waveform of (7, 4) Hamming encoder.

Table 5
A set of test vectors for the (7, 4) Hamming decoder.

$a_6\ a_5\ a_4\ a_3$	$a_2\ a_1\ a_0$	$S_1\ S_2\ S_3$	Error	$A_6\ A_5\ A_4\ A_3$
0000	000	000	–	0000
0001	011	000	–	0001
0101	101	000	–	0101
1010	010	000	–	1010
1111	111	000	-	1111
0001	000	011	a_3	0000
0011	011	101	a_4	0001
0001	101	110	a_5	0101
0010	010	111	a_6	1010
1111	011	100	a_2	1111

that the decoder has successfully realized the expected function.

6. Analysis and discussion

Table 6 shows the physical characteristics of the (7, 4) Hamming encoder in terms of cell number, area, delay, and gate number. It can be seen that the proposed encoder has significant improvements in all aspects. This is due to the adoption of the TIEO gate and the multi-layered crossing method. Compared with the optimal performance of the encoders proposed in [22-24], the proposed design reduces the number of cells, area, delay, and the number of gates by at least 51.04%, 81.61%, 75.00% and 83.33% respectively. Table 7 shows the comparison of the physical characteristics of the 3–8 decoder in [25-29]. Although the

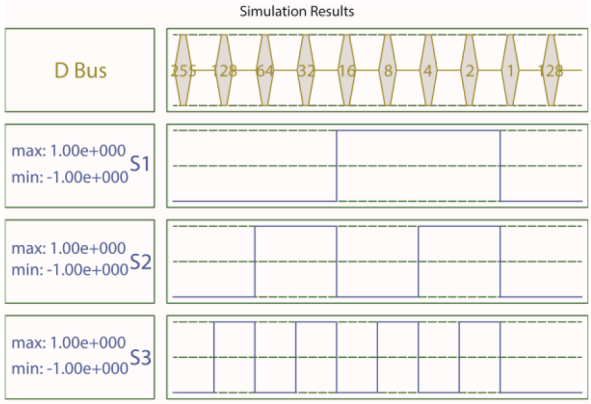

Fig. 16. Simulation waveform of 3–8 decoder.

Table 6
Performance of (7, 4) Hamming encoders.

Design	Cell count	Area(μm^2)	Latency(clocks)	Gate count
[22]	632	1.13	5	24
[23] (a)	571	0.87	3	24
[23] (b)	578	0.90	3	24
[24] (a)	384	0.96	2	18
[24](b)	547	1.19	6	35
Proposed	188	0.16	0.5	3

Table 7
Performance of 3–8 decoders.

Design	Cell count	Area (μm^2)	Latency (clocks)	Gate count	Full I/O accessibility
[25]	1074	2.24	2.75	19	Yes/yes
[26]	651	0.77	2.75	8	Yes/yes
[27]	237	0.28	1	15	No/no
[28]	859	1.2	3.5	21	Yes/yes
[29]	344	0.44	2.5	12	Yes/yes
Proposed	386	0.28	1	8	Yes/yes

proposed 3–8 decoder does not have the least number of cells, it performs well in all aspects. The number of cells used in [27] is also very few, but its I/O ports are difficult to access, which makes it more difficult to use as a part of other circuits, and can only be used as an independent device, with poor scalability. Compared with the 3–8 decoder in [29], the design proposed in this manuscript uses slightly more cells, but the clock delay and occupied area are much smaller. In general, the latter has lower hardware complexity. The 3–8 decoder design proposed in this manuscript has regularity, high signal strength, easy access to I/O ports, and is very suitable for expansion as the basic unit of a complex circuit. Table 8 shows the performance indicators of the (7, 4) Hamming decoder in [22, 24]. The syndrome calculator is an indispensable component in the Hamming decoder. Compared with the existing design, the syndrome calculator proposed in this manuscript can reduce the cell number, area, clock delay, and gate number by 39.06%, 82.35%, 50.00%, and 66.67% at least respectively. A complete QCA Hamming decoder is implemented in [22]. Compared with the decoder proposed in this manuscript, the latter has been greatly improved in all aspects. Although there is a part of the reason is that the decoder proposed in [22] is a part of its communication system, the design in this manuscript still has obvious advantages such as a small occupied area and fast response speed. From the data in the three tables, it can be seen that the design scheme proposed in this manuscript is simple and efficient, not only fulfilling the expected functions, but also reducing consumption in all aspects as much as possible, which has good practicability.

Table 8
Performance of syndrome calculator and (7, 4) Hamming decoders.

Design	Cell count	Area (μm^2)	Latency (clocks)	Gate count
Syndrome calculator (a) [24]	489	1.26	2	27
Syndrome calculator (b) [24]	547	1.19	8	18
Syndrome calculator [22]	898	1.24	7	36
Proposed	260	0.18	0.75	6
Decoder [22]	1618	2.56	18	54
Proposed	593	0.45	1.75	14

7. Conclusion

In this manuscript, (7, 4) and (15, 11) Hamming code circuits that can be used to detect and correct single-bit error code are designed in QCA, and their physical characteristics are analyzed from the aspects of cell count, area, delay, and gate count. The evaluation result shows that the proposed design has obvious advantages, is more suitable for the design of QCA circuits, and reduces the cost of circuits. A Hamming communication network is constructed based on the proposed Hamming code components. For different generator matrices, the same effect can be achieved by only changing the interconnection lines of the circuit without major changes to the structure. In addition, this design has enlightening significance for the realization of nano communications with QCA, and plays an important role in the detection and correction of communication circuits.

Declaration of Competing Interest

All the data presented in this paper is original and is not submitted to any other journals.The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work is supported by the Fundamental Research Funds for the Central Universities of China under Grant No. JZ2020HGTA0085 and No. JZ2020HGQA0162.

References

[1] J.C. Das, D. De, S.P. Mondal, A. Ahmadian, F. Ghaemi, N. Senu, QCA based error detection circuit for nano communication network, IEEE Access 7 (2019) 67355–67366.
[2] L. Wang, G. Xie, A novel xor/xnor structure for modular design of QCA circuits, IEEE Trans. Circuits Syst. II Express Briefs 67 (12) (2020) 3327–3331, https://doi.org/10.1109/TCSII.2020.2989496.
[3] S. Babaie, A. Sadoghifar, A.N. Bahar, Design of an efficient multilayer arithmetic logic unit in quantum-dot cellular automata (QCA), IEEE Trans. Circuits Syst. II Express Briefs 66 (2019) 963–967.
[4] R.K.urnia Nasaruddin, Hamming coding for multi-relay cooperative quantize and forward networks, in: 2016 IEEE Region 10 Symposium (TENSYMP), Bali, Indonesia, 2016, pp. 321–325.
[5] A.A.A.A. Shal, A novel formulation of hamming code, in: 2009 6th International Conference On Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Pattaya,Chonburi, Thailand, 2009.
[6] R.K. Nath, B. Sen, B.K. Sikdar, Optimal synthesis of QCA logic circuit eliminating wire-crossings, IET Circuits Devices Syst 11 (2017) 201–208.
[7] K. Walus, T.J. Dysart, G.A. Jullien, R.A. Budiman, QCADesigner: a rapid design and simulation tool for quantum-dot cellular automata, IEEE Trans. Nanotechnol. 3 (2004) 26–31.
[8] L.H.B. Sardinha, A.M.M. Costa, O.P.V. Neto, L.F.M. Vieira, M.A.M. Vieira, Nanorouter: a quantum-dot cellular automata design, IEEE J. Sel. Areas Commun. 31 (2013) 825–834.
[9] J.C. Das, D. De, QCA based design of polar encoder circuit for nano communication network, Nano Commun. Networks 18 (2018) 82–92.
[10] J.C. Das, D. De, Nanocommunication network design using QCA reversible crossbar switch, Nano Commun. Networks 13 (2017) 20–33.

[11] B. Das, J.C. Das, D. De, A.K. Paul, Nano-router Design for Nano-Communication in Single Layer Quantum Cellular Automata, Springer Nature Singapore, 2017, pp. 121–133.
[12] T.F. Cesar, L.F.M. Vieira, M.A.M. Vieira, O.P.V. Neto, Cellular automata-based byte error correction in QCA, Nano Commun. Networks 23 (2020) 13.
[13] A. Norouzi, 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. 38 (2019) 231–243.
[14] J.C. Das, D. De, Circuit switching with quantum-dot cellular automata, Nano Commun. Networks 14 (2017) 16–28.
[15] B. Debnath, J.C. Das, D. De, Design of image steganographic architecture using quantum-dot cellular automata for secure nanocommunication networks, Nano Commun. Networks 15 (2018) 41–58.
[16] M. Mustafa, M.R. Beigh, Design and implementation of quantum cellular automata based novel parity generator and checker circuits with minimum complexity and cell count, Indian J. Pure Appl. Phys. 51 (2013) 60–66.
[17] S. Sheikhfaal, S. Angizi, S. Sarmadi, M.H. Moaiyeri, S. Sayedsalehi, Designing efficient QCA logical circuits with power dissipation analysis, Microelectron. J. 46 (2015) 462–471.
[18] H. Chen, H. Lv, Z. Zhang, X. Cheng, G. Xie, Design and analysis of a novel low-power exclusive-OR gate based on quantum-dot cellular automata, J. Circuits Syst. Comput. 28 (2019) 1–17.
[19] M. Poorhosseini, A.R. Hejazi, A fault-tolerant and efficient xor structure for modular design of complex QCA circuits, J. Circuits Syst. Comput. 27 (2017) 1–24.
[20] S. Angizi, E. Alkaldy, N. Bagherzadeh, K. Navi, Novel robust single layer wire crossing approach for exclusive or sum of products logic design with quantum-dot cellular automata, J. Low Power Electron. 10 (2014) 259–271.
[21] F. Ahmad, G.M. Bhat, H. Khademolhosseini, S. Azimi, S. Angizi, K. Navi, Towards single layer quantum-dot cellular automata adders based on explicit interaction of cells, J. Comput. Electron. 16 (2016) 8–15.
[22] D.S. Silva, L.H.B. Sardinha, M.A.M. Vieira, L.F.M. Vieira, O.P.V. Neto, Robust serial nanocommunication with QCA, IEEE Trans. Nanotechnol. 14 (2015) 464–472.
[23] C. Mukherjee, S. Panda, B. Maji, A.K. Mukhopadhyay, Hamming code generators using LTEx module of quantum-dot cellular automata, in: A. Sarkar (Ed.), 2019 Devices for Integrated Circuit, 2019, pp. 16–20.
[24] G.J. Xie, Y.L. Xiang, Y.Q. Zhang, S. Liu, H.J. Lv, Design and implementation of encoding and check code circuit with hamming code on QCA, Int. J. Unconv. Comput. 10 (2014) 391–404.
[25] M. Kianpour, R. Sabbaghi-Nadooshan, A novel modular decoder implementation in quantum-dot cellular automata (QCA), 2011 International Conference on Nanoscience, Technology and Societal Implications 2011, pp. 1–5.
[26] J.-.C. Jeon, Low hardware complexity QCA decoding architecture using inverter chain, Int. J. Control Autom. 9 (2016) 347–358.
[27] F. Deng, G. Xie, R. Zhu, Y. Zhang, A matrix representation method for decoders using majority gate characteristics in quantum-dot cellular automata, J. Supercomput. 76 (2020) 2842–2859.
[28] J.C. Das, D. De, T. Sadhu, A novel low power nanoscale reversible decoder using quantum-dot cellular automata for nanocommunication, in: 2016 3rd International Conference on Devices, Circuits and Systems, 2016, pp. 220–224.
[29] D. De, T. Purkayastha, T. Chattopadhyay, Design of QCA based programmable logic array using decoder, Microelectron. J. 55 (2016) 92–107.

Jianhong Huang received the B.E. degree in Integrated circuit design and systems from Hefei University of Technology, Hefei, China, in 2019. He is studying for the master's degree in the School of Electronic Science and Applied Physics in Hefei University of Technology, Hefei, China. His-research interests include digital circuit design and nanoelectronic circuits and systems.

Guangjun Xie received the Ph.D. degree in Signal and Information Processing from University of Science and Technology of China, Hefei, China, in 2002. He worked as a postdoctor in Optics in University of Science and Technology of China, Hefei, China, from 2003 to 2005. He was a senior visitor at IMEC in 2007 and ASIC in 2011. He is currently a Professor in the School of Electronic Science and Applied Physics, Hefei University of Technology, Hefei, China. His-research interests include VLSI design, reliability, and fault tolerance, nano-electronic circuits and systems.

Rui Kuang received the B.E degree in electrical engineering and automation from Wuhan University of Technology, Wuhan, China, in 2017. He is studying in the school of Electronic Science and Applied Physics, Hefei University of technology. His-research interests include digital circuit design, nano circuit design and related software development.

Feifei Deng received the B.E. degree in Electronic Science and Technology from Hefei University of Technology, Hefei, China, in 2016. He is now pursuing his doctoral degree in this school. His-research interests include the design of quantum-dot cellular automata and other beyond-CMOS technologies.

Yongqiang Zhang received the B.E. degree in Electronic Science and Technology from Anhui Jianzhu University, Hefei, China, in 2013 and the Ph.D. degree from Hefei University of Technology, Hefei, China, in 2018. He was with the Department of Electrical and Computer Engineering, University of Alberta, as a Visiting Student for one year. He is currently with the School of Electronic Science and Applied Physics, Hefei University of Technology, Hefei, China. His-research interests include VLSI design and nanoelectronic circuits and systems.