Decimal to Excess-3 and Excess-3 to Decimal Code Converters in QCA Nanotechnology

Farhad Fouladinia  
Kermanshah University of Technology

Mohammad Gholami (✉ m.gholami@umz.ac.ir)  
University of Mazandaran

Research Article

Keywords: QCA, Excess-3, Decimal, Energy Dissipation, Nanotechnology

Posted Date: November 2nd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-2218039/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
Abstract

In this paper, two digital code converters are presented, excess-3 to decimal, and decimal to excess-3. The tile method is used to design proposed circuits in quantum-dot cellular automata (QCA) nanotechnology. The tile method gives a unique block for the majority and NOT gates. This property facilitates integrating circuits and since the NOT gate is not used in the tile method, the proposed circuits can do their work as fast as possible. Both of the proposed code converters has 1.75 clock cycles delay and have an energy dissipation of about 100meV. In the excess-3 code to decimal converter 516 cells are used, which occupy an area equal to 0.43µm$^2$ also in the decimal to excess-3 code converter. 321 cells are used, which occupy an area equal to 0.28 µm$^2$.

1. Introduction

Nowadays Quantum-dot Cellular Automata (QCA) is one of the new technologies in nanoscale which can be used in future circuits [1]. Most of the digital circuits are implemented with CMOS technology, but CMOS has some problems like power consumption and circuit size [2]. So, for solving these problems a new method (QCA) is presented [3, 4]. QCA technology can solve scaling issues and presents high frequency up to terahertz [5]. This technology was first introduced by Lent and Tougaw in 1993 [6].

Systems in digital electronic like digital computers are worked in binary form. So, for working and calculating with this type of system, data should be converted to the right codes, here the importance of code converting is felt, and many code converters like decimal to binary, binary to decimal, decimal to excess-3, excess-3 to decimal and etc. are presented by researchers [7]. Many ICs convert codes. ICs that are based on transistors have some issues like power dissipation and short channel effects at very low device sizes [5]. QCA is here to solve these problems. On the other hand, some methods are presented that make improve the performance of QCA circuits, one of these methods is the tile which is presented in [8]. Tile presents a combined logic function for INV and Majority gates, so the tile method is area efficient because it presents one unique block for INV and Majority gates [9]. Since reducing the occupied area and improving the performance is very important, the technology of integrated circuits is expanding [11]. In this paper by using tile method, two code converters are presented: excess-3 code to decimal (that is topology of 7443 IC from 74 series IC) and decimal to an excess-3 code.

There aren't many papers that present QCA digital code converters [7], especially papers about ICs but here some papers about code converters are presented. In [10], a BCD to seven segment code is presented. A seven segment has seven parts and every part includes a light emitted diode. This reference presents seven circuits for each segment with the gate method style. One of the defects in this reference is presenting seven circuits instead of one circuit. In [11], at first, a full adder is presented then with three of that FA a binary to BCD converter is presented. This circuit also uses the gate method and isn't optimized as much as possible. In [12] just like the others, the gate method has been used and some digital code converted is presented. One of these converters is excess-3 code to the binary that this circuit is implemented with Feynman gate. In this converter two Feynman gate has been used. In [13], a LTeX
module with gate method style is presented in which a two-input LTEx module consisting of 26 cells and 4 clock phases are used to produce outputs. Every LTEx module actually is a two-input NAND gate with the new structure, then, for implementation of 2 to 16 bits binary to gray code converter this module has been used. In [14], as mentioned in this paper with basic structure in QCA technology like wire, majority gate and ... a simple converter is presented. This circuit converts BCD to excess-3 code and the gate method is used in this design which has no new idea. In [15], at first, a majority gate with five inputs is presented, this majority gate uses 3 clock phases to give an output, then with this majority and rotating in cells a BCD to Excess-3 code converter is implemented. As is clear, due to the high number of inputs and outputs in converting decimal to Excess-3 code, this type of converter has not been paid much attention. Therefore, in this paper we will be focused on the design of 4 bits Decimal to Excess-3 and vice versa code converters in QCA nanotechnology.

This paper has some main parts. The next part is about QCA basic concepts. Section 3 is about both proposed digital code converters and how they can be implemented in the QCA nanotechnology. In the next section “simulations and results” the results of both circuits are reported. Finally, with the “conclusion” part, the paper is concluded.

2. Qca Basic Concepts

A QCA cell consists of several parts. Two freely moving electrons and four areas used as wells to trap electrons are known as dots; all of these are placed inside a square showing each cell's perimeter. All these cases can be seen in Fig. 1. Two valid states can be defined for cells, one is used for logical zero and the other for logical one (known as binary 0 and binary 1 in Fig. 1).

One of the advantages of QCA is its very low energy consumption, which is due to the movement of electrons only inside the cell instead of the movement of electrons in the wire, which results in high energy consumption and energy loss. One of the most important gates in this technology is the majority gate, which has the function Maj (A, B, C) = AB + AC + BC, and using it, you can implement AND and OR gates (by setting one of the three inputs as binary 0 for AND gate and binary 1 for OR gate), this gate is shown in Fig. 2, as can be seen, five cells are used to implement this gate. Another important gate that is very widely used in the implementation of logic circuits is the inverter gate which is shown in Fig. 3. Finally, the wire gate in this technology, which is responsible for data transmission and has many applications, is presented in Fig. 4.

In QCA, clocking is very important because the simultaneous arrival of the inputs of a gate is very necessary to get the correct answer. As can be seen in Fig. 5, clocking in this technology has 4 main phases. The first phase is called the switch, which is the increase of the force that prevents the movement of electrons inside each cell, and the movement of electrons gradually becomes difficult. In the next phase, which is called hold, the forces preventing the movement of electrons inside the cell reach their maximum and the location of the electrons remains constant. In the third phase, which is called release, the amount of the blocking force decreases, and the electrons are slowly released. Finally, in the fourth
phase, known as relax, the cell has no polarity and the electrons move completely freely inside the cell. To apply clocking, it is very important to observe the order of the clock phases, at first, the cells should be in the switch phase (with green color for cells), then hold (with purple color for cells), then release (with light blue color for cells), And finally, relax phase (with white color for cells), after a complete clocking cycle, if more phase is needed, enter the second cycle and again clocking is done from the switch phase according to the mentioned method.

There are several general mechanisms for the design and simulation of QCA circuits, the most common method used for design is the gate-based method and the explanations given are all based on this method. One of the recently used methods is the tile method, which is also used in this paper to design and simulate the circuit [8]. As seen earlier, in the gate-based method, there are unique structures for majority and inverter gates, which eliminates the possibility of integrating these circuits, but in the tile-based method, due to the proposed structure, it is possible to integrate the circuit because a common structure is used for both majority and inverter gates. As is shown in Fig. 6a majority gate with three inputs and one output is presented with a tile block. Also in Fig. 7 inverter gate with this method is presented. In Fig. 8 an example gate is presented that gives $a \oplus b$ output without any use of an inverter gate.

Also, as it can be seen the inverter gate can be implemented just like the example that is presented in Fig. 8, this type of inverter is shown in Fig. 9.

In this inverter in the first place the invert of input “a” is calculated, then that value is OR with binary “0” or that value is AND with binary “1”, both of these calculations have the same results means invert of input “a”.

A QCA cell consists of several parts. Two freely moving electrons and four areas used as wells to trap electrons are known as dots; all of these are placed inside a square showing each cell's perimeter. All these cases can be seen in Fig. 1. Two valid states can be defined for cells, one is used for logical zero and the other for logical one (known as binary 0 and binary 1 in Fig. 1).

One of the advantages of QCA is its very low energy consumption, which is due to the movement of electrons only inside the cell instead of the movement of electrons in the wire, which results in high energy consumption and energy loss. One of the most important gates in this technology is the majority gate, which has the function $\text{Maj} (A, B, C) = AB + AC + BC$, and using it, you can implement AND and OR gates (by setting one of the three inputs as binary 0 for AND gate and binary 1 for OR gate), this gate is shown in Fig. 2, as can be seen, five cells are used to implement this gate. Another important gate that is very widely used in the implementation of logic circuits is the inverter gate which is shown in Fig. 3. Finally, the wire gate in this technology, which is responsible for data transmission and has many applications, is presented in Fig. 4.
In QCA, clocking is very important because the simultaneous arrival of the inputs of a gate is very necessary to get the correct answer. As can be seen in Fig. 5, clocking in this technology has 4 main phases. The first phase is called the switch, which is the increase of the force that prevents the movement of electrons inside each cell, and the movement of electrons gradually becomes difficult. In the next phase, which is called hold, the forces preventing the movement of electrons inside the cell reach their maximum and the location of the electrons remains constant. In the third phase, which is called release, the amount of the blocking force decreases, and the electrons are slowly released. Finally, in the fourth phase, known as relax, the cell has no polarity and the electrons move completely freely inside the cell. To apply clocking, it is very important to observe the order of the clock phases, at first, the cells should be in the switch phase (with green color for cells), then hold (with purple color for cells), then release (with light blue color for cells), And finally, relax phase (with white color for cells), after a complete clocking cycle, if more phase is needed, enter the second cycle and again clocking is done from the switch phase according to the mentioned method.

There are several general mechanisms for the design and simulation of QCA circuits, the most common method used for design is the gate-based method and the explanations given are all based on this method. One of the recently used methods is the tile method, which is also used in this paper to design and simulate the circuit [8]. As seen earlier, in the gate-based method, there are unique structures for majority and inverter gates, which eliminates the possibility of integrating these circuits, but in the tile-based method, due to the proposed structure, it is possible to integrate the circuit because a common structure is used for both majority and inverter gates. As is shown in Fig. 6a majority gate with three inputs and one output is presented with a tile block. Also in Fig. 7 inverter gate with this method is presented. In Fig. 8 an example gate is presented that gives \( a \overline{b} \) output without any use of an inverter gate.

Also, as it can be seen the inverter gate can be implemented just like the example that is presented in Fig. 8, this type of inverter is shown in Fig. 9.

In this inverter in the first place the invert of input “a” is calculated, then that value is OR with binary “0” or that value is AND with binary “1”, both of these calculations have the same results means invert of input “a”.

3. Proposed Digital Code Converters In Qca Circuits

In this section, two digital converters with tile method are proposed, excess-3 code to decimal and decimal to excess-3 code. At first logical circuit for the excess-3 code to decimal converter is presented, then an equivalent circuit for this converter with tile method is given, next step the QCA circuit of the excess-3 code to decimal converter is presented, In the following All the mentioned steps for the second converter namely decimal to excess-3 code are presented.
3.1 Proposed Excess-3 to decimal converter

74 series ICs are very popular, widely used, and have many educational, research, and industrial uses. One of these ICs is 7443, which converts excess-3 code to decimal. The internal circuit of this IC can be seen in Fig. 10. Also, the truth table of this IC is given in Table 1.

<table>
<thead>
<tr>
<th>INPUTS</th>
<th>OUTPUTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>D  C  B  A</td>
<td>0 1 2 3 4 5 6 7 8 9</td>
</tr>
<tr>
<td>0 0 1 1</td>
<td>0 1 1 1 1 1 1 1 1</td>
</tr>
<tr>
<td>0 1 0 0</td>
<td>1 0 1 1 1 1 1 1 1</td>
</tr>
<tr>
<td>0 1 0 1</td>
<td>1 1 1 0 1 1 1 1 1</td>
</tr>
<tr>
<td>0 1 1 0</td>
<td>1 1 1 0 1 1 1 1 1</td>
</tr>
<tr>
<td>0 1 1 1</td>
<td>1 1 1 0 1 1 1 1 1</td>
</tr>
<tr>
<td>1 0 0 0</td>
<td>0 1 1 1 1 1 0 1 1 1</td>
</tr>
<tr>
<td>1 0 0 1</td>
<td>1 1 1 1 1 1 0 1 1 1</td>
</tr>
<tr>
<td>1 0 1 0</td>
<td>1 1 1 1 1 1 0 1 1 1</td>
</tr>
<tr>
<td>1 0 1 1</td>
<td>1 1 1 1 1 1 0 1 1 0</td>
</tr>
<tr>
<td>1 1 0 0</td>
<td>1 1 1 1 1 1 1 1 1</td>
</tr>
</tbody>
</table>

As it is seen, this IC includes 4 inputs for the excess-3 numbers and ten outputs for the decimal numbers, ten NAND gates with four inputs and eight NOT gates are used in this circuit.

As mentioned, the goal is to implement the circuit in Fig. 10 with the tile method in QCA technology. For this purpose, NOT and NAND gates are needed. For NOT gate with tile, the gate that is presented in Fig. 9 will be used. For the NAND gate, the gate that can be seen in Fig. 11 will be used.

As it can be seen in Fig. 11 the NAND gate with tile method can be implemented with just 12 cells without any use of the NOT gate which is so nice for decreasing delays in the circuit. Also because this gate has two inputs, for a NAND gate with four inputs Fig. 12 will be used in QCA circuits.

The following QCA circuit of excess-3 to the decimal converter (7443 IC) with tile method is presented in Fig. 13.

The presented circuit is implemented in three layers, the main layer, the via layer, and the top layer, all of these layers are shown in Figs. 14 to 16.
To reduce the occupied area of the circuit as much as possible, its design and implementation have been done in three layers. Also, in this circuit, 516 cells are used, which occupy an area equal to 0.43µm. In addition, 7 clock phases have been used in this circuit. The total energy consumption for this circuit is about 1.14e-001 eV. As can be seen from Fig. 13, there are not any NOT gates which make the circuit as faster as possible. In this circuit 22 main blocks (which here are tiles and for gate method is majority gate) are used.

### 3.2 Proposed Decimal to excess-3 code converter

In this section, the circuit of a converter is presented which gives an excess-3 code instead of a decimal number. Just like the previously proposed circuit this circuit is implemented with the tile method. The logical circuit for this converter is given in Fig. 17. Also, the truth table for this circuit is presented in Table 2.

<table>
<thead>
<tr>
<th>INPUTS</th>
<th>OUTPUTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>I0</td>
<td>O1</td>
</tr>
<tr>
<td>I1</td>
<td>O2</td>
</tr>
<tr>
<td>I2</td>
<td>O3</td>
</tr>
<tr>
<td>I3</td>
<td>O4</td>
</tr>
<tr>
<td>I4</td>
<td></td>
</tr>
<tr>
<td>I5</td>
<td></td>
</tr>
<tr>
<td>I6</td>
<td></td>
</tr>
<tr>
<td>I7</td>
<td></td>
</tr>
<tr>
<td>I8</td>
<td></td>
</tr>
<tr>
<td>I9</td>
<td></td>
</tr>
</tbody>
</table>

As it is seen, this converter includes ten inputs for the decimal number (from 0 to 9) and four outputs for the excess-3 number (from 0011 to 1100), and four OR are used in this circuit. In this circuit OR gate which is implemented with the tile method shown in Fig. 18 is used.

In the following, the QCA circuit of decimal to excess-3 converter with tile method using is presented in Fig. 19.
Just like the previously proposed circuit this circuit also is implemented in three layers, the main layer, the via layer, and the top layer, all of these layers are shown in Figs. 20 to 22.

Same as the previous circuit to reduce the occupied area of the circuit as much as possible, its design and implementation have been done in three layers. Also, in this circuit, 321 cells are used, which occupy an area equal to 0.28 square micrometers. In addition, 7 clock phases have been used in this circuit. As can be seen from Fig. 19, there are zero NOT gates which make the circuit as faster as possible. In this circuit 14 main blocks (which here are tiles and for gate method is majority gate) are used. The total power consumption for this circuit is about 1.04e-001 eV.

4. Simulations And Results

In this section, the accuracy of the proposed circuits is checked using QCADesigner-E software. QCADesigner-E is one of the most reliable software in the field of QCA, and version 2.2 is used in this paper [16, 17]. This software was developed by the University of Calgary. With this software values such as the number of cells, the number of clock phases, the occupied area, and the power consumption can be extracted.

In the following, the simulation results of each of the converters will be given. In Fig. 23, the inputs of the excess-3 code to decimal converter can be seen. These inputs have produced the outputs in Fig. 24, as it is clear from Fig. 24, the proposed circuit has correctly produced the expected values, for example for producing $(ABC'D')'$ inputs are $A = 1, B = 1, C = 0, D = 0$ that produce $ABC'D'$ output, with inverting this output the main output is produced (this is shown in Fig. 23 and Fig. 24). Figure 25 also shows the state of the used clocks.

For the next converter, decimal to excess-3 code, in Fig. 26, the decimal inputs of this converter can be seen. These inputs have produced the outputs in Fig. 27, as it is clear from the outputs, that the proposed circuit has correctly produced the expected values, for example to producing $O4$ output which one of $I0$ or $I2$ or $I4$ or $I6$ or $I8$ is binary “1” the output will be binary “1” (this is shown in Fig. 26 and Fig. 27). This circuit also has the same clocking for simulation just like Fig. 25.

As mentioned before the first proposed circuit has 516 cells and 7 clock phases have been used in it. Also, the total power consumption for this circuit is about 1.14e-001 eV. The second proposed circuit has 321 cells and 7 clock phases have been used in it. Also, the total power consumption for this circuit is about 1.04e-001 eV. Because there are no similar works for this type of conversion, we cannot compare proposed design with other related works. In Table 3 details of the proposed circuits are given.
Table 3
Details of proposed circuits.

<table>
<thead>
<tr>
<th>Items</th>
<th>Excess-3 to Decimal converter</th>
<th>Decimal to Excess-3 converter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell count</td>
<td>516</td>
<td>311</td>
</tr>
<tr>
<td>Delay (clock phases)</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Main gate count</td>
<td>22</td>
<td>14</td>
</tr>
<tr>
<td>Inverter gate count</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Power consumption</td>
<td>1.14e-001 eV</td>
<td>1.04e-001 eV</td>
</tr>
<tr>
<td>Designing type</td>
<td>Tile method</td>
<td>Tile method</td>
</tr>
<tr>
<td>Ability to integrate</td>
<td>HIGH</td>
<td>HIGH</td>
</tr>
</tbody>
</table>

5. Conclusion

This paper presents two digital converters, the first one is for converting excess-3 code to decimal and another one is for converting decimal to excess-3 code. The method that is used in both of them is the tile method. This method has some advantages, with this method circuits can be integrated and get faster because there is no need to use an inverter gate lonely. In the end, both of the proposed circuits were evaluated with QCADesigner-E, and some reports like cell count, delays, power consumption, and ... were presented.

Declarations

Ethics approval and consent to participate: Not applicable.

Consent for publication: Not applicable.

Availability of data and materials: Not applicable.

Competing interests: Not applicable.

Funding: Not applicable.

Authors' contributions

Farhad Fouladinia: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Writing - Original Draft, Visualization.

Mohammad Gholami: Conceptualization, Methodology, Investigation, Resources, Writing - Review & Editing, Supervision.
Acknowledgements: Not applicable.

References


**Figures**

**Figure 1**

QCA cell and polarization states

**Figure 2**

Majority Gate in QCA
Figure 3

Inverter Gate in QCA

Figure 4

Wire Gate in QCA.
Figure 5
Clocking in QCA.

Figure 6
Majority Gate in QCA using Tile method.

Figure 7
Inverter Gate in QCA using Tile method.
**Figure 8**

An example with the Tile method.
Figure 9

Another implementation of inverter gate with Tile method.

Figure 10

7443 IC internal circuit.
**Figure 11**

NAND gate with tile.
Figure 12

Four inputs NAND gate with tile.
Figure 13

Proposed excess-3 to the decimal converter (7443 IC) QCA circuit.
Figure 14

Main layer of proposed excess-3 to decimal converter (7443 IC) QCA circuit.
Figure 15

Via layer of proposed excess-3 to decimal converter (7443 IC) QCA circuit.

Figure 16
Top layer of proposed excess-3 to decimal converter (7443 IC) QCA circuit.

Figure 17

Decimal to excess-3 code converter logical circuit.
Figure 18

OR gate with tile.
Figure 19

Proposed decimal to excess-3 converter QCA circuit.
Figure 20

Main layer of proposed decimal to excess-3 converter QCA circuit.
Figure 21

Via layer of proposed decimal to excess-3 converter QCA circuit.
Figure 22

Top layer of proposed decimal to excess-3 converter QCA circuit.

Figure 23
Inputs of proposed excess-3 to decimal converter QCA circuit.

![Figure 24](image)

Outputs of proposed excess-3 to decimal converter QCA circuit.

![Figure 25](image)

Clock phases in QCADesigner-E software.
Figure 26

Inputs of proposed decimal to excess-3 converter QCA circuit.
Figure 27

Outputs of proposed decimal to excess-3 converter QCA circuit.