SIMULATION OF QCA BASED BINARY TO BCD CONVERTER IN XILINX USING HDLQ

By

A.G. SASIKALA *　　S. MARAGATHARAJ **

* PG Scholar, Knowledge Institute of Technology, Salem, India.
** Assistant Professor, Knowledge Institute of Technology, Salem, India.

ABSTRACT

Quantum Dot Cellular Automata (QCA) is an emerging nanotechnology in the field of Quantum electronics for low power consumption and high speed of operational phenomenon. Such type of circuit can be used in many digital applications where CMOS (Complementary Metal- Oxide- Semiconductor) circuits cannot be used due to high leakage and low switching speed. The code converters are the basic units for conversion of data to perform arithmetic operations. A new effective binary to BCD(Binary Coded Decimal) converter design using QuantumDot Cellular Automata is presented in this paper. Compared to the available code converters in VLSI technology, this method of using Quantum dots reduces area and increases switching speed. 3-input Majority gate is the basic and universal gate in QCA design. In accordance with the code converter design using 3-input majority gate logic, a 5-input majority gate logic structure is also used here to design the binary to gray code converter. This replacement improves the speed of the system by reducing the number of clock cycles required to produce the output. The simulations are done using Xilinx ISE Design suite to get power and timing analysis.

*Keywords: Clocking of QCA, Code Converter, Majority Gate, QCA (*Quantum dot Cellular Automata*), HDLC (High-level*

INTRODUCTION

In the past few decades, the exponential scaling in feature size and the increase in processing power have been successfully achieved by Very Large Scale Integration (VLSI) technology, mostly using CMOS; However, in the not-so-distant future [1], this technology will face serious challenges as the fundamental physical limits of its devices are reached. In recent years, there has been extensive research at nanoscale to supersede the conventional CMOS using the so-called emerging technologies. It is anticipated that these fundamentally different technologies can achieve extremely high densities and high operational speed. Among these new devices, Quantum-dot Cellular Automata (QCA) not only gives a solution at nanoscale, but also offers a new method of computation and information transformation (often referred to as processing-in-wire).

1. QCA Basics

Quantum dot Cellular Automata is used to represent Quantum cells, which are the basic building blocks of Quantum electronics, as CMOS is used in VLSI technology. The basic structures used in the design of QCA digital circuits are the majority gate, inverter and wire. In the case of Majority gate 3, 5 and multi input majority gates are available. As two logic states [-1 and +1] are established in QCA, it is easy to implement many digital circuits in this technology. The clocking zones which are introduced [5], provide us a wide platform for designing combinational as well as many sequential circuits. As we are using the code converters often in digital systems, this paper will provide us an easy way to reach our goals effectively.

A schematic diagram of a four-dot QCA cell is shown in Figure 1. The cell consists of four quantum dots positioned at the corners of a square. The cell contains two extra mobile electrons, which are allowed to tunnel between neighboring sites of the cell. The compensating positive

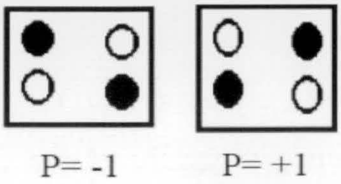


Figure 1. QCA Cell Polarization

charge is fixed and immobile. Tunneling out of the cell is assumed to be completely suppressed by the potential barriers between the cells. It is also possible to add a fifth dot at the center of the square; the addition of this dot improves the behavior of the cell slightly but for simplicity the authors will focus mainly on the four-dot cell.

If the barriers between cells are sufficiently high, the electrons will be well localized on individual dots. The Coulombic repulsion between the electrons will tend to make them occupy antipodal sites in the square as shown in Figure 1. For an isolated cell, there are two energetically equivalent arrangements of the extra electrons which is denoted as a cell polarization P= +1 and P= -1 [1]. The term "cell polarization" refers only to this arrangement of charge and does not imply a dipole moment for the cell. The cell polarization is used to encode binary information P= +1 which represents binary 1and P= -1 which represents binary 0.

2. Basic Logic Cells Design In QCA

When QCA cells are arranged nearby and one cell is fixed with any one polarization, all the neighbour cells are polarized in the same way because of the coulombic interaction. This property of QCA cells lets us to use them as wires. The regular arrangement of these QCA cells forms a binary wire [3], through which information can be carried out from one end to the other end. Figure 2. shows the binary wire. The left most cell is fixed with a polarization, which is carried out to the right most cell through this arrangement.

Cells which are positioned diagonally from each other tend to anti-align. This feature is employed to construct an inverter as shown in Figure 3. The anti-alignment can also be seen by examination to be simple on sequence of the mutual repulsion between electrons and the geometry of the cells. Although two diagonal cells function as an inverter [4], this more symmetric design ensures exact symmetry between the inversion of a one and a zero.

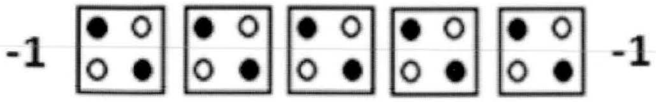


Figure 2. A Binary Wire

Figure 4 shows the fundamental QCA logical device, a three-input majority gate, from which more complex circuits can be built. The central cell, labeled the device cell, has three fixed inputs, labeled A, B, and C. The device cell has its lowest energy state if it assumes the polarization of the majority of the three input cells.

M (A, B, C) = AB+BC+CA

Five inputs majority gate is a new structure used for 3- input AND and OR gate implementation which is shown in Figure 5. This device can be used to reduce circuit complexity.

Computation in a QCA majority gate is performed by driving the device cell to its lowest energy state, as shown in Figure 4 and Figure 5. This is achieved when the device cell assumes the polarization of the majority of the three or five input cells. The reason why the device cell always assumes a majority polarization is because it is in this polarization state that the Coulombic repulsion between electrons in the inputs cells is minimized [1]. The polarization of the device cell is then transfered to the output cell.

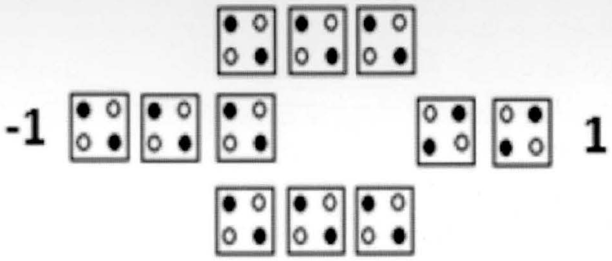
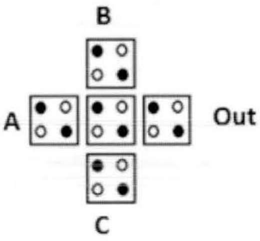


Figure 3. Inverter



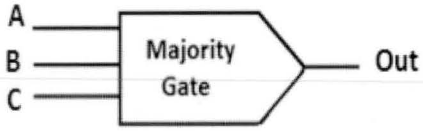


Figure 4. Majority Voting Gate (3-inputs)

3. QCA Clocking

Figure 6 shows a pipelined clocking scheme that must be employed for a QCA circuit to function properly. In this scheme, the clock is divided into four phases with each phase incurring a 90 phase delay. The phases are called switch, hold, release, and relaxed [6],[7]. The clocking scheme allows an array of cells to perform a certain computation, have its state frozen by the raising of the inter-dot potential barriers, and have the result of the computation serve as the input to the successor array of cells. During the computation, the successor cell array is kept unpolarized so that it does not influence the computation. Finally, the neighboring cell arrays concurrently receive neighboring clock phases.

In the switch phase, the initial state of the QCA cell is unpolarized and the inter-dot potential barriers are kept low. The barriers are then raised and the cells become polarized according to the state of their neighboring (input) cells. It is in this state where the actual computation is performed. In the hold phase, the cell states are fixed and the potential barriers are held high so that the cells can serve as inputs to the next 3 stage. In the release phase, the potential barriers are lowered and the cells are allowed to relax to the unpolarized state. Finally, in the relaxed state, the potential barriers remain low and the cells remain unpolarized.

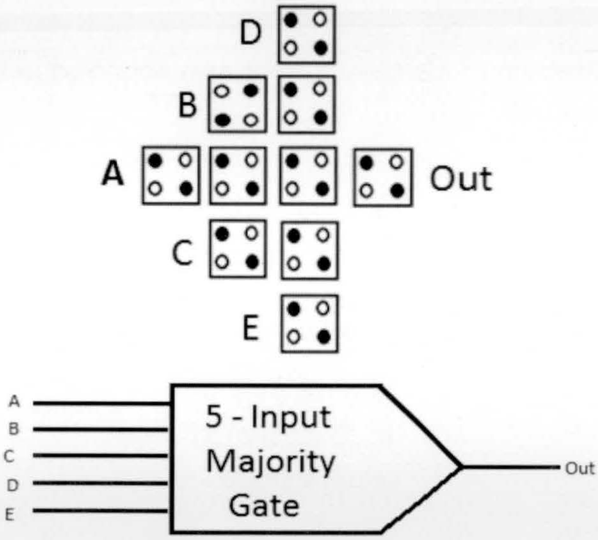


Figure 5. Five input Majority Gate

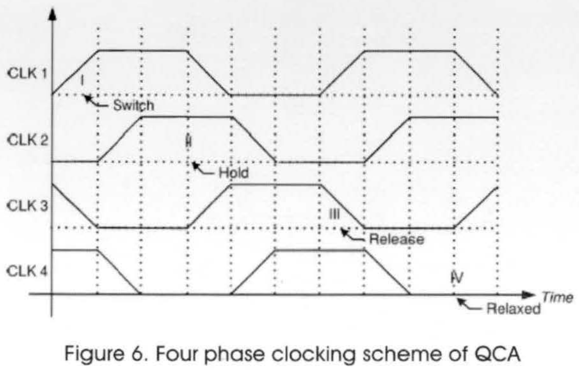


Figure 6. Four phase clocking scheme of QCA

4. Design of Binary to BCD Converter

A binary code is a group of n bits that assumes up to 2n distinct combinations of one's and zero's, with each combination representing one element of the set that is being coded. Discrete quantities of information are represented in digital systems by binary codes. The availability of a large variety of codes for the same discrete elements of information results in the use of different codes by different logical digital systems. Sometimes, it is necessary to use the output of one system applied as the input to another system.

A conversion circuit must be inserted between the two systems, if each uses maybe different for the same information. A code each uses different codes for the same information. A code converter is a combinational circuit that makes the two systems compatible, even though each uses a different binary code. To convert from one binary code X to another binary code Y, the input lines must supply the bit combinations of elements as specified by code X and the output lines must be generate the corresponding bit combinations of code Y. A combinational logic circuit performs this transformation with the help of logic gates.

In order to design a combinational logic circuits, first form the specifications of the circuit, determine the required number of input levels and output levels and assign a specific symbol to each levels. After that, derive the function table that defines the required relationship between input and outputs. Then obtain the simplified Boolean function for each output as a function of the input variables. Finally, draw the logic diagram and verify the results of design by simulation.

The Karnaugh map is providing a simple procedure for simplifying the given Boolean function. This method provides a pictorial form of a function table. Consider a new procedure for understanding and reduce the logical operations in the form of Majority of majority functions. In each square that has a Boolean variable one value represent that was11-, and the square with Boolean 0 will be representing the value 00- to produce a new map method that will be pointed to as J-map denoting majority function.

To designate the four binary input variables by b1, b2, b4 and b8 and the four output variables by g0, g1, g2 and g3. Here, four input binary variable may have 16 bit combinations. In order to obtain the simplified Boolean functions for the outputs, better use J map instead of K map. There are four maps that represent four output levels of the circuit as a function of the four input variables. A two level logic diagram may be obtained directly from the Boolean expressions derived from the proposed Methodology.

Figure 7. shows the CMOS based layout of Binary to BCD converter design with 4 inputs and 5 outputs. The total area occupied for this circuit layout is calculated as 1140μm2.

5. Proposed Methodology

5.1 Design I: 3- Input Majority Gate:

QCA computation proceeds by the orientation of cells based on polarization of neighboring cells. The QCA inverter is built by neighboring QCA cells on the diagonal, which causes Coulomb forces to place the two electrons in opposing walls of the cell compared to the source. The combinational four-bit binary to BCD code converter circuit consists of 12 number of 3-input majority gates, to implement the function. Figure 8 shows the design of QCA binary to BCD converter using three input majority gate shown in Figure 4.

5.2 Design II: 5- Input Majority Gate:

The alternative approach, which uses 5 input majority logic is used to reduce the number of quantum cells used and the total number of clock cycles required for computation. Figure 9. shows the design of QCA binary to BCD converter using Five input majority gate shown in Figure 5.

6. Review of HDLQ

HDLQ is a HDL-based design tool and associated environment for QCA circuits. VHDL and Verilog are the

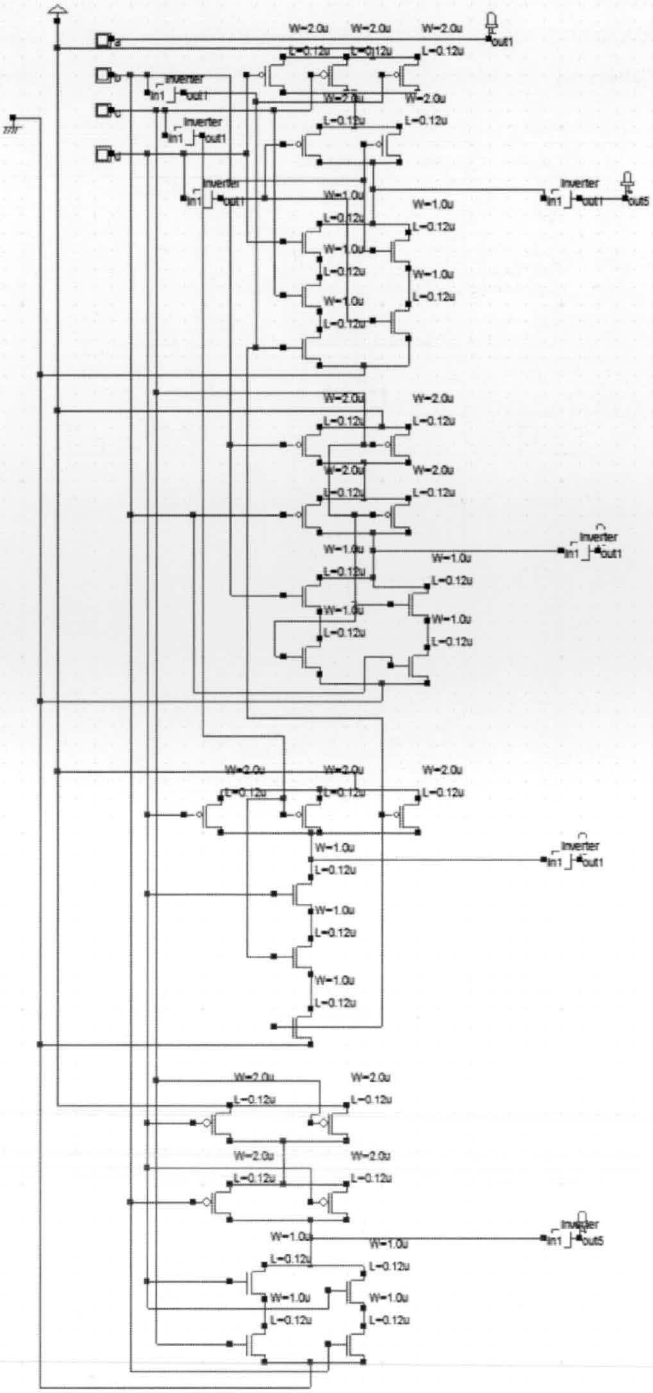


Figure 7. Circuit diagram of Binary to BCD code converter using NMOS and PMOS

most widely used HDL standards (http://dftgroupsvn.uniroma2.it/wiki/HdlQ). Verilog HDL has been utilized in this work to build a QCA device library for assembling different circuits with the objective of using it for fast assembly of larger designs. In QCA designing environment using Verilog-based simulation tool, QCA blocks can be assembled and simulated in their own clocking zones. When a device is in the Release or Relax phases, its outputs are in a high impedance state (denoted by Z). When the Switch phase is entered, computation is performed and the logic outputs reflect the operation on a time-dependant basis for the inputs of the device. When the device is in Hold phase, the last attained logic value is locked at the output during the

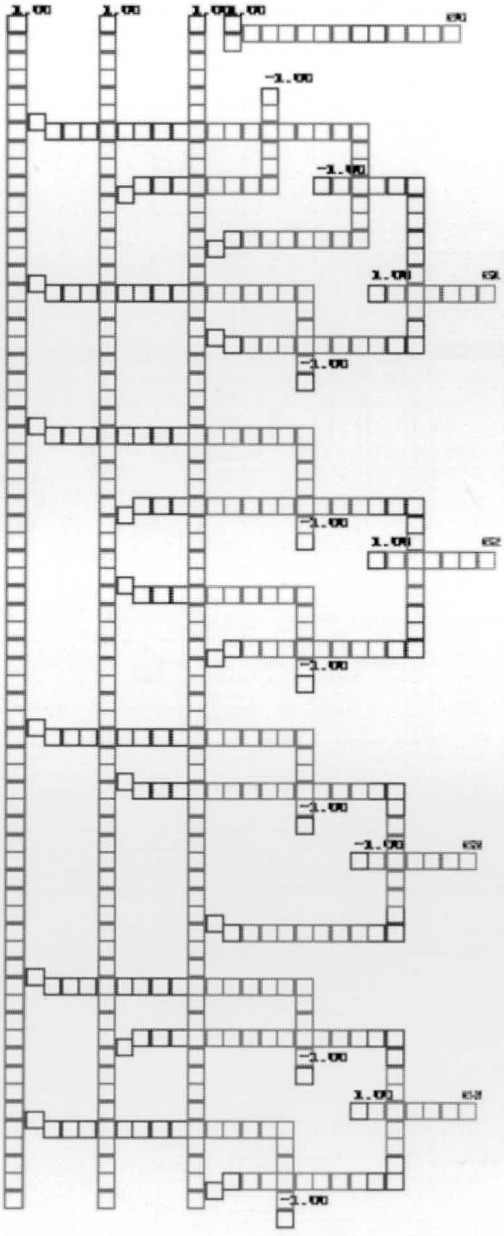


Figure 8. Binary to BCD converter design using QCA

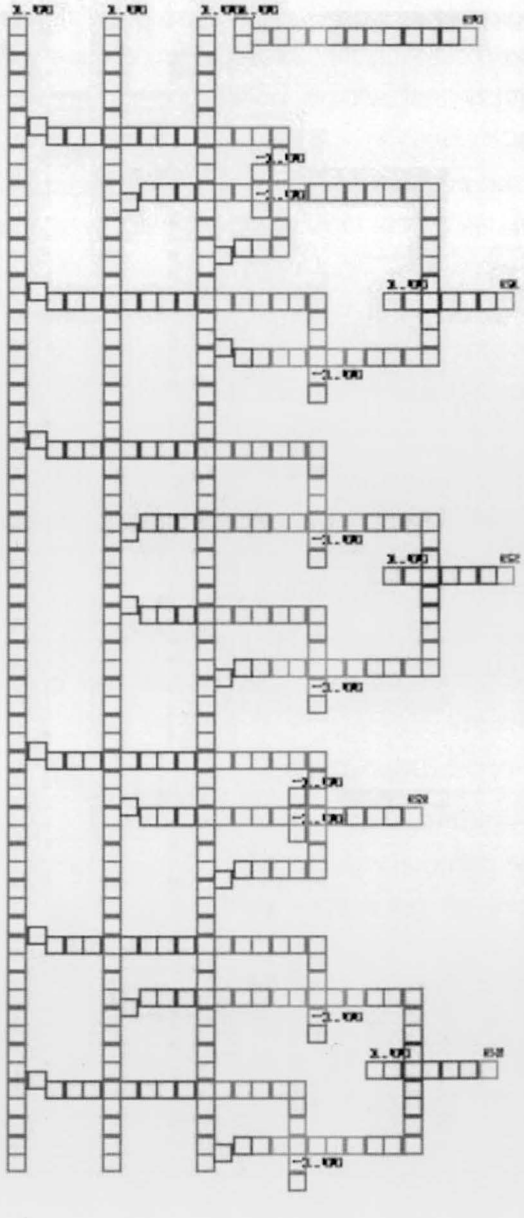


Figure 9. Binary to BCD converter using 5 input majority gate

Model	No. of cells required	No. Of clock cycles required
3 – Input majority gate	512	3
5 – Input majority gate	474	2

Table 1. Results of Comparison of Two Designs

whole phase. To implement a QCA circuit by HDLQ, at first a logic block diagram is depicted that consists of constructive blocks and their clock zones. To implement logic block diagram by verilog, at first each constructive block is coded and simulated as a module. Then by proper relations between modules, the final block diagram is coded and simulated. In this paper, Xilinx ISE is used for simulation. With reference to the two papers [8],[9] the authors have implemented the design of Binary to BCD converter in Xilinx ISE, which is clearly explained in section 7.

7. Results and Discussions

The functionality of the proposed combinational code converter circuit is verified using QCA Designer tool ver.2.0.3. The simulated waveforms of combinational four bit binary to BCD code converter circuit using 3-input majority gate and 5-input majority gate are shown in Figure 8. and Figure 9. respectively. The code converter circuit has 4 clocking zones. When the authors use 5-input majority gate, initially clock 0 is used to get the inputs A and B. Clocks 1 and 2 are used to route inputs for majority gate logic, clock 3 is used for finding majority logic at first stage and clock 0 is again used to carry the output to the next stage of design. Finally, clock 1 is used to compute

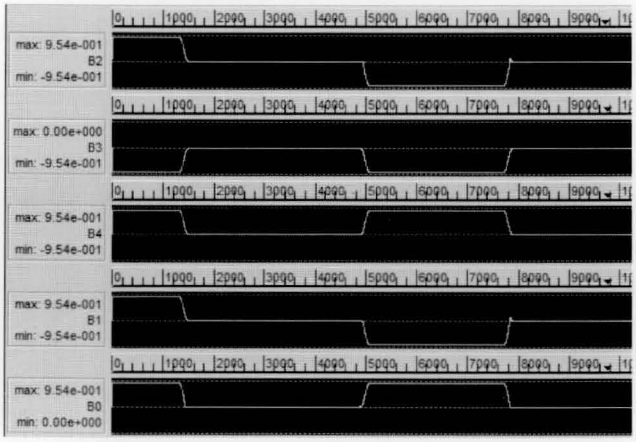


Figure 10. Binary to BCD converter using 3-input majority gate with inputs (A=1,B=1,C=1,D=0)

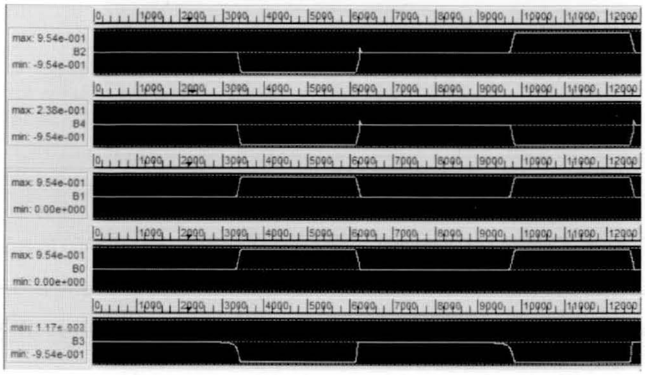


Figure 11. Binary to BCD converter using 5-input majority gate with inputs (A=1,B=1,C=1,D=0)

Technology	Area
CMOS layout	1140μm^2
QCA (3 – input majority gate based) layout	0.76 μm^2
QCA (5 – input majority gate based layout)	0.72 μm^2

Table 2. Comparison of Areas of Different Layouts

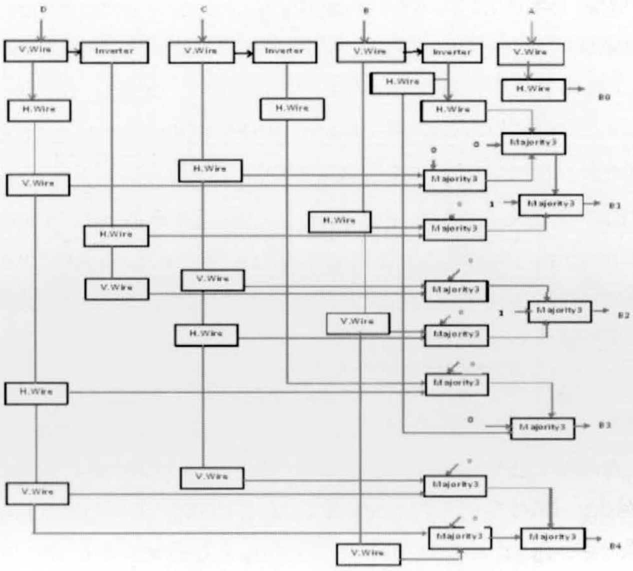


Figure 12. Logic block diagram of Binary to BCD converter for Design I in QCA

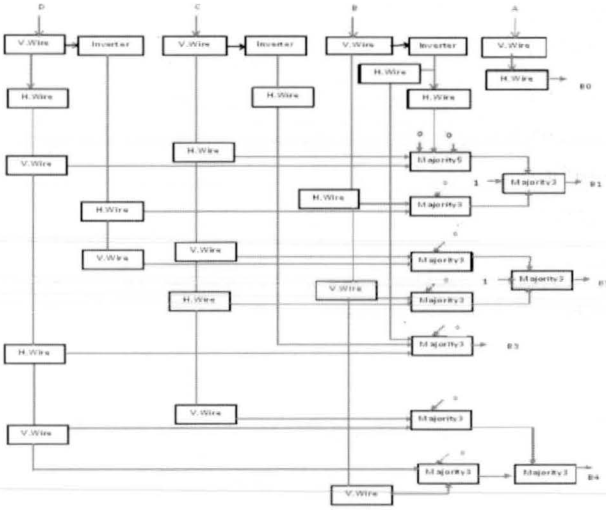
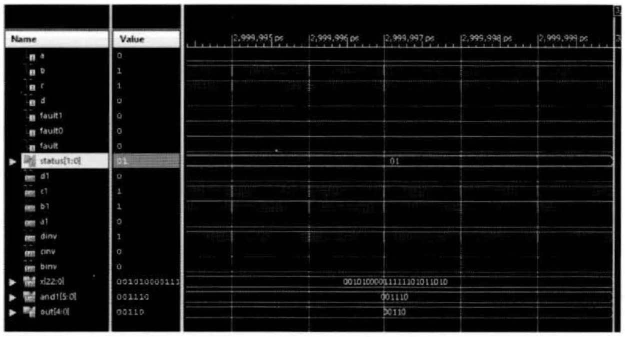


Figure 13. Logic block diagram of Binary to BCD converter for Design II in QCA



Name	Value
a	0
b	1
c	1
d	0
fault1	0
fault0	0
fault	0
status[1:0]	01
d1	0
c1	1
b1	1
a1	0
dinv	1
cinv	0
binv	0
x[22:0]	001010000111
and1[5:0]	001110
out[4:0]	00110

Figure 14. Simulation results using Xilinx ISE v12.1

the majority logic again and the output is available at this zone.

In the case of 3-input majority logic, one extra clock is needed when compared to the BCD converter designed using 5-input majority gate. Similarly, to design different combinational circuit based code converters, 4 clock zones are required to produce the output.

Table 1, shows that, the introduction of 5-input majority gate logic structure is an area efficient structure, which is a good replacement for the 3-input majority gate structure in terms of area and speed.

Figure 10. and 11 shows the time graph of the proposed BCD code converters. From these two waveforms, the authors conclude that, both 3-input majority based design and 5-input majority based design are giving the same output, but at differet timings. Compared to the first one i.e., 3-input majority based structure, second one i.e., 5-input majority structure gives output earlier.

Table 2. clearly shows the comparison of three different designs of Binary to BCD converters. The areas are represented in micrometer2. So it is proved that QCA is an area efficient technology compared to the CMOS based layout. By modifying the structure further by using vertical and crossover cells, the authors can further reduce the area.

Figures 12 and 13 show the Logic block diagrams of the QCA based layouts (Figures 8 and 9), which are used for the programming of the Deisgns I and II in Xilinx ISE.

The simulation results of the proposed Binary to BCD code converter using 3-input majority gate is shown in Figure 14.

Conclusion

In this paper, two effective methods for designing Binary to BCD conversion have been explained and simulated. Compared to the available designs (P.Ilanchezhian, et al. May 2013), this proposed QCA- based design approach opens a wider path for digital circuit design with smaller dimensions. By using this approach, power efficient circuits can be designed with greater accuracy. The main drawback of this design approach is the fabrication of such minute circuits. The chip yield is very low, when the designs are practically integrated.

References

[1]. **C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernstein, (1993).** "Quantum cellular automata," Nanotechnology, Vol. 4, No. 1, pp. 49–57.

[2]. **C. S. Lent, P. D. Tougaw, and W. Porod, (1993).** "Bistable saturation in coupled quantum dots for quantum cellular automata," *Appl. Phys. Lett.*, Vol. 62, No. 7, pp. 714–716.

[3]. **C. S. Lent and P. D. Tougaw,** "Lines of interacting quantum-dot cells: A binary wire," *J. Appl. Phys.*, Vol. 74, No. 10, pp. 6227–6233.

[4] **P.D. Tougaw and C. S. Lent, (1994).** "Logical devices implemented using quantum cellular automata," *J. Appl. Phys.*, Vol. 75, No. 3, pp. 1818–1825.

[5] **A. Gin, S. Williams, H. Meng, and P. D. Tougaw, (1999).** "Hierarchical design of quantum cellular automata," *J. Appl. Phys.*, Vol. 85, No. 7, pp. 3713–3720.

[6]. **K. Hennessy and C. S. Lent, (2001).** "Clocking of molecular quantum-dot cellular automata," *J. Vac. Sci. Technol. B: Microelectron. Nanometer Struct.*, vol. 19, no. 5, pp. 1752–1755.

[7]. **C. S. Lent, M. Liu, and Y. Lu, (2006).** "Bennett clocking of quantum-dot cellular automata and the limits to binary logic scaling," *Nanotechnology,* Vol. 17, pp. 4240–4251,

[8]. **M. Malekpour and R. Sabbaghi-Nadooshan, (2012).** "A Novel Simulation of Decoders using HDLQ in QCA" , *Canadian Journal on Electrical and Electronics Engineering,* Vol. 3, No. 7.

[9]. **M. Malekpour and R. Sabbaghi-Nadooshan, A. Kashaninia, (2012).** "A Novel Multiplexer Simulation using HDLQ in QCA" *Canadian Journal on Electrical and*

Electronics Engineering, Vol. 3, No. 8.

[10]. P.Ilanchezhian, R.M.S.Parvathi, (2013). "Nanotechnology based Effective Design Approach for Code Converter Circuits using QCA," *International Journal of Computer Applications* (0975–8887), Vol. 69–No.8.

[11]. http://dftgroupsvn.uniroma2.it/wiki/HdlQ

ABOUT THE AUTHORS

Ms.A.G.Sasikala, is currently pursuing her ME degree in VLSI Design in Knowledge Institute of Technology, Salem. She completed BE degree in Electronics and Instrumentation Engineering from Anna University of Technology, Coimbatore in Erode Sengunthar Engineering College, Erode. She is a member in IEEE. Her area of Interests include Quantum dot Cellular Automata, VLSI architectures and Virtual Instrumentation.

Mr.S.Maragatharaj, is currently working as an Assistant Professor in Knowledge Institute of Technology, Salem. He completed her ME degree in Anna University of Technology, Coimbatore and BE degree in Electronics and Communication Engineering in Annai Mathammal Sheela Engineering College, Namakkal. He is a member in IEEE. His research interests include Low Power VLSI, Analog VLSI and Nano Electronics.