Analysis And Design Of Priority Encoder Circuit Using Quantum Dot Cellular Automata

P. Ilanchezhian
Associate Professor, Department of IT, Sona College of Technology, Salem

Dr. R. M. S. Parvathi
Principal, Sengunthar College of Engineering, Tiruchengode.

Abstract

Today digital circuits play a prominent role in most of the communication applications. In this paper an effective approach to analysis and design of priority encoder using quantum dot cellular automata is explored in nanoscale. This paper we use three input majority gate is the fundamental component of the QCA circuit implementation. The proposed priority encoder circuit is designed and simulated using quantum dot cellular automata designer tool for the four input levels and also this simulator tool is more useful for building a complex priority encoder input levels. The proposed structure of encoder required only less number of majority gate functions compared to previous structures because of its three input levels.

1. Introduction

Logic circuits for digital systems may be combinational digital circuit or sequential digital circuits. A combinational circuit is represented with logic gates whose outputs at any occasion are computed from only the current combination of input levels. A combinational circuit performs a function that can be indicated understandably by a set of Boolean functions. Quantum dot cellular automata (QCA) is a computational methodology as an alternate to field effect transistor (FET) [8] devices. It was introduced in early 90's by Lent, et. al and gained importance after its fabrication and experimental success using Coulomb blockade phenomenon [9]. QCA is a transistor less computation paradigm that addresses the issues of device density and interconnection. The basic quantum dot cell is charged with two excess electrons and performs computation on coulomb interactions of electrons [5],[6]. Quantum dots are nanostructures created from standard semi conductive materials. These structures are modeled as quantum wells. By using Quantum dot cellular automata instead of interconnecting wires, the cells transfer the information throughout the circuit [10]. The most important operators used in the QCA technology are the three input majority gate and inverter. To implement any QCA circuit can be built using only majority gates and inverters.

2. QCA Designer

The circuit designers need a more accurate simulation and design layout tool to determine the functionality of QCA circuits. QCA designer tool is the important publicly available design and simulation tool for QCA. It is developed at the ATIPS Laboratory, at the University of Calgary, QCA Designer currently supports three different simulation engines, and many of the CAD features required for complex circuit design. [3],[4]. The first is a digital logic simulator, which requires cells to be either null or fully polarized. The second is a nonlinear approximation engine, which uses the nonlinear cell-to-cell response function to iteratively determine the stable state of the cells within a design. The third uses a two-state Hamiltonian to form an approximation of the full quantum mechanical model of such a system. [3],[4].

3. QCA Fundamentals

3.1. QCA cell

One of the fundamental units of QCA is the QCA cell made up of four quantum dots arranged along the corners of a square out of which two are inhabited by electrons. The electrons are quantum mechanical particles they are able to tunnel between the dots in a cell. The electrons in the cell that are placed adjacent

to each other will interact; as a result the polarization of one cell will be directly affected by the polarization of its neighboring cells. In order to represent binary information logic 1 and logic 0 the cell polarization P = +1 and P= -1 is used respectively [11],[6]. QCA cell polarization is shown in Figure 1.

Figure 1: QCA cell polarization

3.2. QCA Logic Devices

Majority gate (MG) and inverter are the fundamental logic elements available with QCA. Digital circuits are generated using a combination of these two gates.

3.2.1. QCAWire

QCA wire helps in propagation of logic levels with help of electrostatic repulsion and not current flow. In a QCA wire, the binary signal propagates from input to output because of the electrostatic interactions between cells [5],[7]. The Figure 2 represents the propagation of 90° QCA wire. The alternative for 90° QCA wire, a 45° QCA wire can also be used. Here the propagation of the binary signal alternates between the two polarizations.

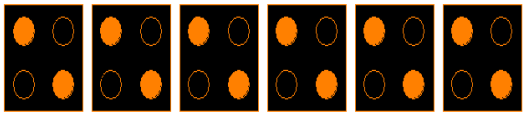


Figure 2.1: QCA wire (90°).

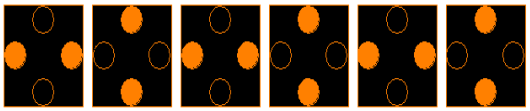


Figure 2.2: QCA wire (45°).

3.2.2. QCA Inverter:

The different structure of the QCA inverter is shown. Fig.3.which is usually formed by placing the cells with only their corners touching. The electrostatic interaction is inverted, because the quantum-dots corresponding to different polarizations are misaligned between the cells [2],[7]. The inverter is built by neighboring QCA cells on the diagonal, which causes Coulomb forces to place the two electrons in opposing wells of the cell compared to the source.

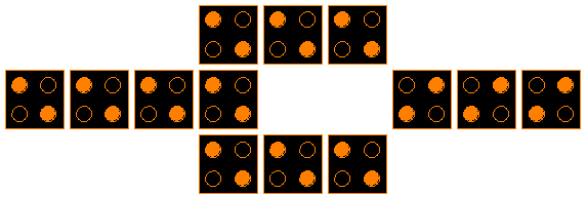


Figure 3.1: QCA Inverter

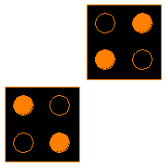


Figure 3.2: QCA 2 cell Inverter

3.3 QCA Majority Gate

The majority gate produces an output that reflects the majority of the inputs. The QCA majority gate has four terminal cells out of which three are representing input terminal cells and the remaining one represents the output cell [7]. Let us assuming the three inputs are X, Y and Z, the logic function of the majority gate is

$$M(X,Y,Z) = XY + YZ + ZX. \quad (1)$$

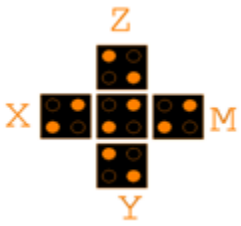


Figure 4.1: QCA majority gate

Figure 4.2: Majority gate Symbol

A most fundamental QCA majority gate is shown in Fig. 4. The propensity of the majority device cell to move to a ground state ensures that it takes on the polarization of the majority of its neighbours. The device cell will tend to follow the majority

polarization because it represents the lowest energy state. To produce AND gate function and OR gate function applying the polarization of one input to the QCA majority gate as binary logic "1" and binary logic "0," respectively. These functions are mathematically written as follows:

$$M(X, Y, 0) = X.Y \quad (2)$$
$$M(X, Y, 1) = X+Y \quad (3)$$

Thus, we can construct all QCA logic circuits on three-input majority gates. To produce well-organized QCA design, the digital circuits are implemented with the help of majority gate-based design techniques are needed. [1].

4. Priority Encoder circuit

A general encoder circuit is digital combinational circuit that computes the reverse operation of decoder circuits. A digital encoder circuit consists of 2^n number of input lines and produce n number of output lines. The generated output lines represent binary code corresponding to the input lines. The general encoder has two important problems. Suppose two inputs are active simultaneously then the generated output produce an undefined combination. In order to avoid this problem we introduce the priority function. Another problem in the general encoder circuit is that an output with all zero's is produced when all the input levels are equal to zero, but this output is equal when A input is equal to one. A priority encoder is an encoder circuit that accepts the priority function. The basic principle of the priority encoder circuit perform the function if two or more than two inputs are equal to one at simultaneous instant then the input having greater priority level will be considered. The simplified function table of the four input level priority encoder circuit is indicated in table I.

Table I. Function table of Priority encoder

Input Levels				Output Levels		
D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

From table along with two functional output levels one more output level namely valid bit indicator indicates the correct input levels. If all the input levels are equal to zero then the valid bit indicator indicates its output as low output and otherwise it indicates the output level at high state. Because of the simplified form the function table it is not indicated for all the possible input combinationsand only indicates the few input combinations and X in the table indicates whether the input level is either active high or active low. Based on the table the greater priority will be given on input level D_3 then followed by other input levels D_2,D_1 and D_0 respectively. When the input level D_3 is active high then there is no need for verify the other input levels likewise if D_3 is in active low then D_2 is active high then C will be higher priority level compared to other input levels. The karnaugh map is providing a simple procedure for simplifying the given Boolean function. This method provides a pictorial form of a function table.A karnaugh map is a diagram and it is made up of squares with each square represent a min term of the function that is to be simplified.Consider a new procedure for understanding and reduce the logical operations in the form of Majority of majority functions. In each square that has aBoolean one value we represent that was11-, and the square with Boolean 0 will be represent the value 00- toproduce a new map method that will be pointed to asJ-map denoting majority function.. Now we use the Boolean functionin the form of majority functions.The simplified J map sum of product term representation of priority encoder is given by

$$X = M((D_3,D_2,1)) \quad (4)$$
$$Y = M(M(D_1,D_2',0),1) \quad (5)$$
$$V = M(M(D_3,D_2,1,M(D_1,D_0,1)) \quad (6)$$

Based on the simplified Boolean functions the three input majority gate implementation of combinational priority encoder circuit is shown in Figure 5.

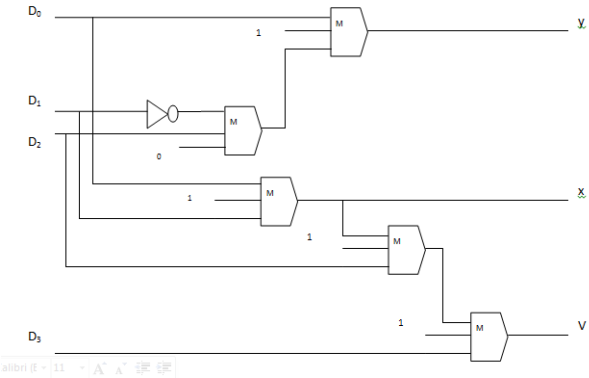
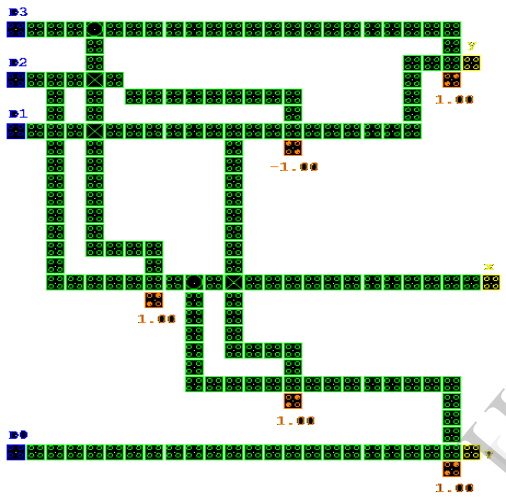


Figure 5. Three input majority gate priority encoder circuit

5. QCA Implementation

QCA computation proceeds by 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 wells of the cell compared to the source. The combinational priority encoder circuit is designed with 5 majority gates and one inverters. The corresponding QCA implementation of combinational priority encoder circuit is shown in Fig.6. In our implementation the total number of cells required is 183cells, with an area 498.00nmx586.43 nm.

Figure 6. QCA implementation of Priority encoder circuit



6. Simulation Results

The combinational priority encoder circuit functionality is verified using QCA Designer tool ver.2.0.3. The simulated waveforms of combinational priority encoder circuit are shown in Fig.7. The priority encoder circuit has four clocking zones.

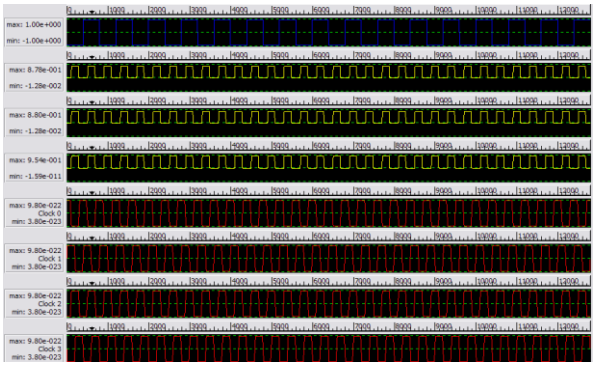


Figure 7.Simulation result of Priority encoder circuit

Initially clock 0 is used to get the inputs A and B. Clock 1 is used to route inputs for majority gate logic, clock 2 is used for finding majority logic and clock 3 is used to compute output. The output is available at clock 0 again. Clock 1 to 3 considered here is a sequence of setup for hold, relax and release phase, to control the flow of information in QCA circuits. Similarly to design priority encoder circuit also need 4 clock zones are used to produce required output.

8. Conclusion

The design of combinational priority encoder circuit is presented in this paper. The digital AND gate and OR gate was implemented with the help of three input majority gate functions and majority gate implementation of priority encoder circuit have been designed and tested using QCA Designer software. The function of the three input majority AND and OR gate function has been verified according to the truth table. The proposed layouts of combinational priority encoder circuit are significantly smaller than the circuits using CMOS technology and it reduces the area as well as complexity required for the circuit than the previous QCA circuits. The designed QCA based combinational priority encoder circuit can be used to represent only four input levels and generate only two output levels. In future this can be extended to represent sixteen input levels and generate four output levels.

References

[1] E.N.Ganesh1, Lal Kishore2 and M.J.S. Rangachar3"Implementation of Quantum cellular automata combinational and sequential circuits using Majority logic reduction method" International Journal of Nanotechnology and Applications.Volume 2, Number 1, 2008.

[2] W. J. Townsend and J. A. Abraham, "Complex gate implementations for quantum dot cellular automata," in Proc. 4th IEEE Conf. Nanotechnology, 2004, pp. 625–627.

[3] K. Walus, G. Schulhof, and G. A. Jullien, "High level exploration of quantum-dot cellular automata (QCA)," in Conf. Rec. 38th Asilomar Conf. Signals, Systems and Computers, 2004, vol. 1, pp. 30–33.

[4] K. Walus, T. Dysart, G. Jullien, and R. Budiman, "QCADesigner: A rapid design and simulation tool for quantum-dot cellular automata,"IEEE Trans. Nanotechnology., vol. 3, no. 1, pp. 26–29, Mar. 2004.

[5] Heumpil Cho, Student Member, and Earl E. Swartzlander, Jr., Fellow, IEEE. "Adder Design and Analyses for Quantum-Dot Cellular Automata", IEEE Trans. Nano., Vol.6, No.3, may 2007.
[6] K. Walus, G.A. Jullien, and V. Dimitrov. Computer arithmetic structures for quantum cellular automata. Asilomar Conference on Signals, Systems, and Computers, November 2003.
[7] K. Walus, G. Schulhof, G. A. Jullien, R. Zhang, and W. Wang, "Circuit design based on majority gates for applications with quantum-dot cellular automata," in Conf. Rec. 38thAsilomar Conf. Signals, Systems and Computers, 2004, vol. 2, pp. 1354–1357.
[8] K. Walus, G. A. Jullien, V. S. Dimitrov, "Computer Arithmetic Structures for Quantum Cellular Automata", Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA. 2003.
[9] W. Wang, K. Walus, G.A. Jullien, "Quantum-Dot Cellular Automata Adders" Copyright IEEE Nano 2003 Conference, August 12-14, San Francisco, CA.
[10] R. Zhang, K. Walus, W. Wang, and G. A. Jullien, "A method of majority logic reduction for quantum cellular automata," IEEE Trans.Nanotechnol., vol. 3, no. 4, pp. 443–450, Dec. 2004.
[11] W. Wang, K. Walus, and G.A. Jullien.Quantum-dot cellular automata adders. IEEE Nano Conference, August 2003.