TEST Engineering & Management

May - June 2020
ISSN: 0193-4120 Page No. 3001 - 3005

Designing of Multiplexers using Quantum-Dot Cellular Automata

N.Partheeban[1], S.Senthilkumar[2], S.Arun[3], Ravindran K[4], A.Nageswaran[5], S.Padmapriya[6]

[1]Professor, Dept of CSE, GALGOTIAS UNIVERSITY, Uttar Pradesh, India
[2]Professor, Department of Electronics and Instrumentation Engineering,
[2]Bharath University, Chennai,
[3]Professor, Department of ECE, Prathyusha Engineering College,
[4]Assistant Professor, Department of IT, Easwari Engineering College, Chennai,
[5]Assistant Professor, Department of IT, Kings Engineering College,
[6]Professor, Department of CSE, Prathyusha engineering college,
[1]n.partheeban@galgotiasuniversity.edu.in, [2]senthilsasthans@gmail.com, [3]yesarun1810@prathyusha.edu,
[4]ravindran.k@eec.srmrmp.edu.in, [5]nageswaraan@gmail.com, [6]padmapriya.it@prathyush.edu.in

Article Info
Volume 83
Page Number: 3001 - 3005
Publication Issue:
May - June 2020

Article History
Article Received: *19 August 2019*
Revised: *27 November 2019*
Accepted: *29 January 2020*
Publication: *12 May 2020*

Abstract

Current transistor based semiconductor gadgets are getting impervious to scaling. Because of diminishing supply voltage, the force utilization from leakage current is a major test. Nano innovation is the conceivable option in contrast to these issues. In these potential choices, Quantum dot cell automata is designed to represent data and perform computations. It has potential for faster speed, littler measure and low force utilization. It is having a basic cell as the essential component. Combinational circuits like adders and multipliers were actualized beforehand. In this paper, we are going to execute multiplexers and demultiplexers utilizing quantum cells. Multiplexers are the combinational circuits which can switch at least two information signs to a yield port. Multiplexers find wide applications in correspondence systems, exchanging exercises, instruments and so on.

Keywords: *Quantum dot automata, Quantum Cellular, Multiplexers, Demultiplexers*

1. Introduction

From the statement of Moore's law, we are well-known that the transistor thickness duplicates for at regular intervals. In any case, in nowadays the scaling of MOS devices getting extremely troublesome. On the off chance that we need to scale the gadgets we should scale the channel length, however in the event that we scale the channel length the spillage flows will increment, and furthermore the door and channel may get converse predisposition. Thus, there is a breaking point to the scaling of the gadgets. So as to defeat with the issues of scaling the MOS gadgets, such a large number of examines are made. Among them Quantum-dab Cellular Automata [1] which can be an answer for the scaling issues in CMOS innovation.

Quantum-dot Cellular Automata (QCA), which make use of arrays of coupled quantum dots to put in operation Boolean logic functions. Conventional digital technologies use ranges of voltage or current to represent binary values. In contrast, QCA uses the position of electrons in quantum dots to represent binary values '0' and '1'. The advantage of QCA is the exceptionally high packing derived from the small size of dots along with the interconnection simplicity and the notably low power-delay product [2],[3].

The paper is organized as follows. In Section II, the back bone of QCA circuits are explained, Section III and IV show the architectural design and implementation of multiplexers and demultiplexers. Simulation results are presented in Section V and in Section VI the conclusion follows.

May - June 2020
ISSN: 0193-4120 Page No. 3001 - 3005

2. Back Bone Of Qca Circuits

The implementations of QCA circuits [4] are worked out with the help of Majority gates and Inverter circuits.

A. Majority Gate

The majority gate output will depend on the majority of the inputs. The functioning of majority gate is explained with an example, in the Fig. 1. A, B and C are inputs and O is the output, when A = '1', B = '0', C = '1' then the majority gate output is '1'.

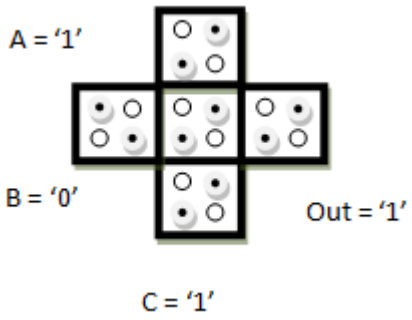


Figure 1: Majority Gate

B. Inverter

An inverter is a circuit which inverts the given input and gives the result as output. The Fig. 2 represents the implementation of inverter with QCA cells.

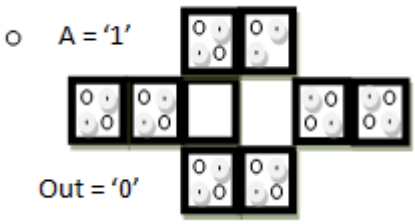


Figure 2: Represents the Implementation of Inverter with QCA Cells

3. Multiplexers

In electronics, a multiplexer (mux) is a device that performs multiplexing; it chooses one of numerous simple or advanced info signals and advances the chose contribution to a single line. A multiplexer of 2n inputs has n select lines, which are utilized to choose one of the input lines to be sent to the output. An electronic multiplexer makes it feasible for a few signs to share one device or asset, for instance one A/D converter or one correspondence line, rather than having one device for every information signal. On the opposite end, a demultiplexer (demux) is a gadget taking a solitary info signal and choosing one of numerous information yield lines, which is associated with the single info. A multiplexer is frequently utilized with a reciprocal demultiplexer on the less than desirable end.

A. 2:1 Multiplexer

In general for a multiplexer there will 2^n inputs and n selection lines and one output will be there. The bit combinations of the selection line determine which input line to be selected. For a 2:1 mux [5] there will be 2 inputs and one selection line and one output line. Fig. 3 shows the realization of the 2:1 multiplexer using majority gate.

The Boolean equation for the 2:1 mux is given as

$$F = A.\bar{S} + B.S$$

Here the *S* represents the selection line, *A* and *B* are the input lines and *out* represents the output line.

This function can be described in terms of the majority function as

$$F = \mathrm{M}\{\mathrm{M}(A,\bar{S},0),\mathrm{M}(B,S,0),1\}$$

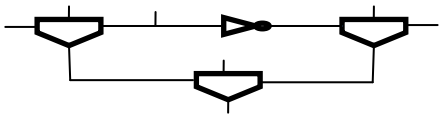


Figure 3: Shows the realization of the 2:1 multiplexer using majority gate

The majority gates M1 and M2 will perform the AND operations and M3 will perform the OR operation.

B. 4:1 Multiplexer

For a 4:1 mux the number of input lines are 4 and the selection lines are 2. Here the inputs are *A, B, C, D* and the selection lines are *S0* and *S1*.

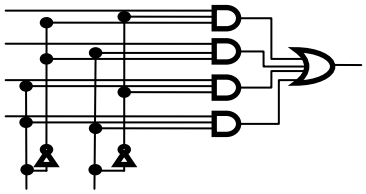


Figure 4: Representation of 4:1 multiplexer using AND/OR logic.

For the implementation of Fig. 4 with the majority gates we require 274 cells and area occupied on the plane is 0.37μm². So with the new design we can reduce the number of cells required, which ultimately decrease the area occupied.

The new design is implemented by reducing the Boolean expression [6] as below. The Boolean equation for the 4:1 mux is given

$$F = (A.\overline{S_0}.\overline{S_1}) + (B.\overline{S_0}.S_1) + (C.S_0.\overline{S_1}) + (D.S_0.S_1)$$

The majority equation is obtained by solving the Boolean function.

$$F = (A\overline{S_1} + BS_1)\overline{S_0} + (C\overline{S_1} + DS_1)S_0$$

$$F = (A.\overline{S_1} + B.S_1)\overline{S_0} + (C.\overline{S_1} + D.S_1)S_0 + (A.\overline{S_1} + B.S_1)\overline{S_0}(C.\overline{S_1} + D.S_1)S_0$$

From the basic equation of the majority function M (a, b, c) = ab + bc + ca, the above equation can be written as

TEST Engineering & Management

May - June 2020
ISSN: 0193-4120 Page No. 3001 - 3005

$F = \mathrm{M}((A.\overline{S_1} + B.S_1)\overline{S_0}, (C.\overline{S_1} + D.S_1)S_0, 1)$

Similarly the function $A.\overline{S_1} + B.S_1$ can be written as

$\mathrm{M}(\mathrm{M}(A, \overline{S_1}, 0), \mathrm{M}(B, S_1, 0), 1$

By substituting the above equations (7) and (8) in (6) we get $F = \mathrm{M}\{\mathrm{M}\{\mathrm{M}\{\mathrm{M}(A, \overline{S_1}, 0), \mathrm{M}(B, S_1, 0), 1\}, \overline{S_0}, 0\}, \mathrm{M}\{\mathrm{M}\{\mathrm{M}(C, \overline{S_1}, 0), \mathrm{M}(D, S_1, 0), 1\}, S_0, 0\}, 1$

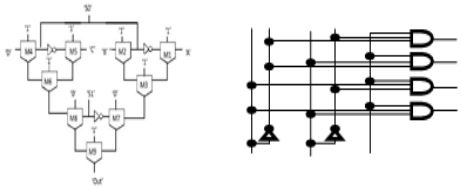


Figure 5: Shows the 4:1 multiplexer implementation using majority gates

Here the 4:1 mux is obtained from two 2:1 mux by considering the equation (9). The majority gates M1, M2, M4, M5, M7, M8 will perform the AND operation and the majority gates M3, M6, M9 will do the OR operation. Fig. 5 shows the 4:1 multiplexer implementation using majority gates, for this implementation we require 158 cells and the area occupied on the layout is 0.30 μm^2.

4. Demultiplexers

A. 1:2 Demultiplexer

In a Demultiplexer for N number of inputs we get 2^N outputs. Fig. 6 shows the implementation of 1:2 Demux using AND/OR logic. For a 1:2 Demux we have one input line and one selection line and two outputs.

We represent the implementation with the majority gates in Fig. 7The majority equations (10), (11) represent outputs of 1:2 Demux.

B. 1:4Demultiplexer

Now we explain the realization of a 1:4 Demux. For 1:4 demux we need one input line, two selection lines which gives four output lines. The majority equations for 1:4 demux are given as

$$Y1 = \mathrm{M}\{\mathrm{M}(\overline{S_0}, \overline{S_1}, 0), I, 0\}$$
$$Y2 = \mathrm{M}\{\mathrm{M}(\overline{S_0}, S_1, 0), I, 0$$
$$Y3 = \mathrm{M}\{\mathrm{M}(S_0, \overline{S_1}, 0), I, 0$$
$$Y4 = \mathrm{M}\{\mathrm{M}(S_0, S_1, 0), I, 0$$

Implementation of this circuit in QCA layout it requires 434 cells and area occupied on the plane is 0.52 μm^2.

The equations (12), (13), (14) and (15) can be implemented with the 3 input AND gate [7], [8] using majority function.

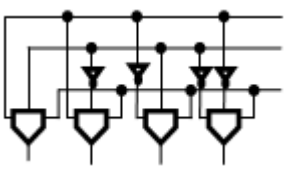


Figure 6: shows the implementation of 1:2 Demux using AND/OR logic

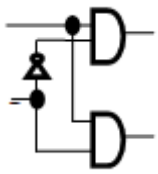
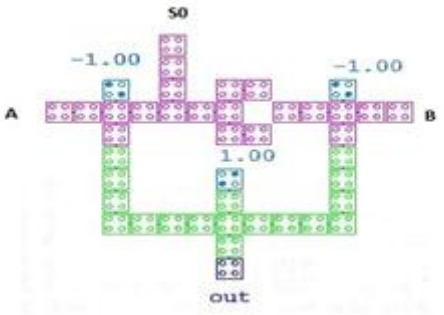


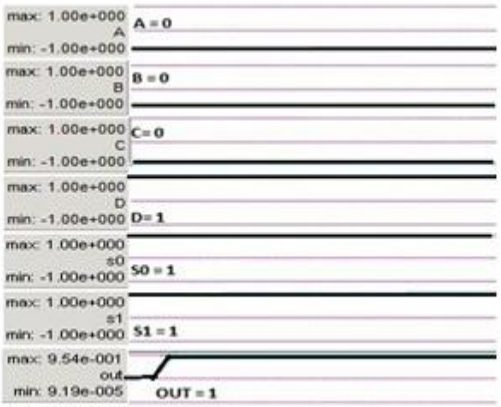
Figure 7: The majority equations (10), (11) represent outputs of 1:2 Demux

5. Simulation Results

For the proposed circuit's layout and functionality checking, a simulation tool for QCA circuits, QCADesigner version2.0.3 is used. All the parameters used are default values in QCADesigner. For all the simulation outputs X- axis represents number of samples per simulation and Y- axis represents the input and output variables. The values below the reference line are '0' and above are '1'.



Mux	Area (μm^2)	Cells
4:1	0.34	258
4:1(proposed)	0.30	158
1:4	0.72	434
1:4(proposed)	0.37	262



TEST Engineering & Management

May - June 2020
ISSN: 0193-4120 Page No. 3001 - 3005

For K stages we require $2^{(N-K)}$ *38 cells + interconnects, where K = 1, 2, 3 ….N.

2:1 Multiplex

For the simulation of 2:1 mux shown in Fig. 10 the inputs are given as *A*= 0, *B*= 1 and selection line *S0* = 1, whenever the clock goes low the output is hold, so the output rises when the clock going to low, resulting output = 1 shown in Fig. 11, where *A, B* are inputs and *S0* selection line.

4:1 Multiplexer

For the simulation of the 4:1 mux shown in Fig. 12, the inputs are given as *A* = 0, *B* = 0, *C* = 0, *D* = 1, and selection lines *S0*=*S1*=1, whenever the clock goes low the output is hold, so the output rises when the clock going to low resulting output = 1, shown in Fig 13, where *A, B, C, D* are inputs and *S0, S1* are selection lines.

1:2 Demultiplexer

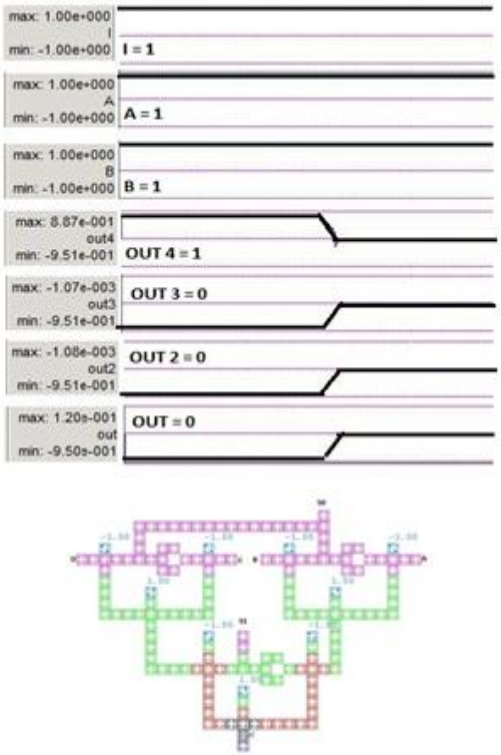


Figure. Layout for 1:2 Demultiplexer.

The 1:2 demux shown in Fig. 14 is simulated by giving the inputs as *I*= 1 and *S*= 0, whenever the clock goes low the output is hold, so the output rises when the clock going to low resulting *Y1*=1 and *Y2*=0, shown in Fig. 15 where *I* is the input line and *S* is the selection line.

1:4 Demultiplexer

The 1:4 demux in Fig. 16 is simulated by giving the inputs as *I*=1, selection lines *A*=1, *B*=1, whenever the clock goes low the output is hold, so the output rises when the clock going to low resulting *out*=0, *out2*=0, *out3*=0, *out4*=1, shown in Fig. 17 where *I* is the input line, *A, B* are selection lines

6. Conclusion

From the above results we can infer that area occupied by the proposed method will reduce 11% to the original implementation in multiplexers while in case of the demultiplexers the proposed layout will reduce 48%. Since the number of cells are going to reduce the power consumed by the QCA circuits also reduce.

References

[1] C.S. Lent, P.D. Tougaw, W. Porod, and G.H. Bernstein, "*Quantum Cellular Automata*," Nanotechnology, vol. 4, no. 1 pp. 49- 57, Jan.1993.

[2] L. Bonci, M. Macucci. *"Analysis of power dissipation in clocked Quantum Cellular Automata circuits,"* Solid –state circuits conference, pp 58-61, IEEE 2006.

[3] John Timler, Craig S. Lent, *"Power gain and dissipation in quantum dot cellular automata,"* J. Appl. Phys. volume 01, number 2, January 2002.

[4] Mrs.M.Parameswari, Dr.T.Sasilatha, Dr.S.Arun and Sharmila.V, "An Proficient Quad Verification Strategy in Cloud Storage", International Conference on Recent Trends in Computing, Communication and Networking Technologies (ICRTCCNT'19)(2019**),** 18-19.

[5] S.Padmapriya, J.S. Richard Jimreeves, P.Kalaiselvi, A.Nageswaran, S.Arun, "Metaclassifiers for Predicting the Robotic Navigational Performance", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8, Issue-11S,(2019), pp. 1234-1238.

[6] Arun, S., Joshua, V., Pillai, A., Nageswaran, "A Fingerprints age determination in different methods using image processing", International Journal of Soft Computing (**2014).**

[7] A.VinothKumar, V.Kannarasu, S.Padmapriya, N.Partheeban and S.Arun, "Design and Implementation of Autonomous Flower Harvester using Image Processing", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-2, July 2019, pp. 2638-2642.

[8] P. Douglas Tougaw, Craig S. Lent, *"Logical devices implemented using quantum cellular automata,"* J. Appl. Phys. 75(3), 1 February 1994.

[9] V. Mardiris, Ch. Mizas, L. Fragidis and V. Chatzis "*Design and simulation of a QCA 2 to 1 multiplexer*," 12th WSEAS International Conference on Computers, Heraklion, Greece, July 23-25, 2008.

[10] W. Wang, K. Walus, and G.A. Jullien, *"Quantum-Dot Cellular Automata Adders,"*

May - June 2020
ISSN: 0193-4120 Page No. 3001 - 3005

Proc. Third IEEE Conf. Nanotechnology, pp. 461-464, 2003.

[11] Aaron Gin, Sara Williams, Haoyu Meng *"Hierarchial design of quantum-dot cellular automata devices,"* Journal of applied Physics, Volume 85, number 7,April 1999.

[12] Dr. S. Padmapriya , T. M. Nivethitha , V. S. Padmini , K. Pavithra , V. S. Sriraja Balaguru , G. Varun Kumar , S. Karthik "Implementation Of Employee Quarters Management System In ERP SAP Foran Organization", Test Engineering & Management, Vol 83: March/April 2020