ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com

International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE) Vol. 2,Special Issue 6, March 2016

DESIGN OF COMBINATIONAL CIRCUITS USING QUANTUM CELLULAR AUTOMATA

Ms.R.Pandimeena[1],Mrs.S.Vallimayil[2], Mrs.S.Jeeva[3], Ms.P.Gayathri[4]
Assistant professor[1,2,3,4]
Electronics and Communication Engineering
Renganayagi Varatharaj College of Engineering, Sivakasi-626140, India[1, 2, 3,4]

***Abstract*— Testing is a process used to verify the correctness of the systems. In this project testing is carried out to detect the two stuck at fault namely stuck at 0 and stuck at 1. Here, the combinational circuits are designed based on majority logic which automatically detect the two stuck at faults. Any combinational designs can be tested for classical unidirectional stuck at faults using only two test vectors such as all 1s and all 0s. Two vectors testable circuits are designed. The importance of this design lies in the fact that it provides the design of reversible combinational circuits completely testable for any stuck at fault by only two test vectors. There by eliminating the need of any type of scan-path access to internal memory cells. The sequential circuits are simulated using QCAD designer. This method provides 100% fault coverage for single missing/additional cell defect**

Keywords*— *Cellular automata, majority logic, quantum-dot, reversible logic.

I. INTRODUCTION

Conservative logic is a logic which exhibits the property that there are an equal number of 1s in the outputs and inputs. Conservative logic is reversible in nature and also not reversible in nature. The circuit has reversibility one to-one mapping between the inputs and the output vector, that is for each input vector there is a unique output vector and vice-versa that conservative logic is called reversible conservative logic when there is a one-to-one mapping between the inputs and the outputs with the property that there is equal number of 1s in the output as in the input. Conservative logic circuits are not reversible, if one-to-one mapping between the inputs and the outputs vectors is not same. So, computation is performed in an irreversible manner and each bit of information lost will produce KTln2 Joules of heat energy. From a thermodynamic point of view, it is also proved that kTln2 energy dissipation would not occur, if a computation is carried out in a reversible way. There are emerging nanotechnologies, such as quantum-dot cellular automata (QCA) computing, optical computing, and superconductor flux logic family, etc., where the energy dissipated due to information destruction will be a significant factor of the overall heat dissipation of the system [1], [2]. Thus, one of the primary motivations for adopting reversible logic lies in the fact that it can provide a logic design methodology for designing ultralow power circuits beyond Further, QCA is one of the improving technologies in which it is possible to implement reversible logic gates [2], [3].QCA makes reduces the circuit clock frequency [4], [5]. In QCA, logic values are represented by the position of the electrons present in the QCA cell shown in Figure. 1. Thus, when the bit Is changed from 1 to 0 there is no actual discharging of the capacitor as in Conventional CMOS.

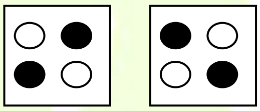


Figure 1 QCA logic cells

Hence, QCA has no energy dissipation electron moment from one cell to another is due to the circular polarisation between the cells and depends upon the adjust cells electron interaction. So, there is no current flow in it. Thus, QCA has no dissipation in KTln2 limit for those emerging nanotechnologies in which the energy dissipated due to erasing of information will be a significant factor of the overall heat dissipation. Therefore, QCA has significant advantage compared to CMOS technology in terms of power dissipation which reduces higher error rate in nano-scale manufacturing, QCA [6]. Further, QCA is one of the improving nanotechnologies in which it is possible to implement reversible logic gates [2], [3]. QCA clock frequency is less than the existing CMOS technology [4], [5].

In this paper, we designed the testable combinational circuits based on majority logic which test the circuits with only two test vectors 1 and 0.So, the circuit provide high fault coverage in single stuck at fault detection in unidirectional faults. Further, we implemented the combinational circuits in the QCA technology and observed that all test vectors 0s and 1s can't provide 100% fault coverage for single missing/additional cell defect in the QCA layout of the gate.

Thus, to have 100% fault coverage implementation of gates in QCA layout is important which provides 100% fault coverage. Further, while designing a QCA layout of combinational circuit designer should concentrate on the number of cells used in the circuit design. Thus, we propose a method to design a inverter in QCA layout design which reduces number of cells in the layout design.

II.BACKGROUND

In this paper, the conservative logic gates are implemented in the QCA nanotechnology thus we are also providing the introduction of QCA layout design which has 4 quantum dots. The logic value representation of the quantum dots is shown in figure 1.In the quantum dot representation has additional of two cells to represent the logic values and QCA layout design uses majority logic to design QCA layout of gates which is shown in figure 2. Majority logic has 5 cells and the output produced in the logic is F= AB+BC+CA and has three inputs.QCA layout has two wire representation one is binary wire which is shown in Figure 3 and another one is inverter wire which is shown in figure 4. The binary and inverter wire not interact with each other. So, the signal can pass through the wires.

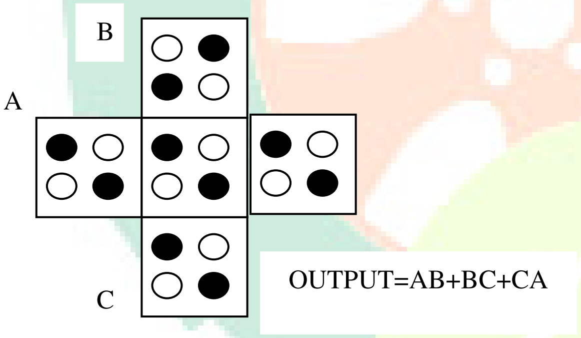


Figure 2 Majority logic

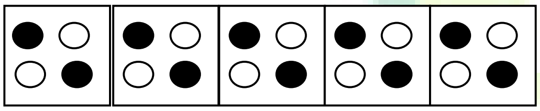


Figure 3 Binary wire

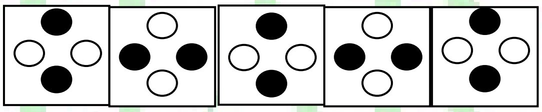
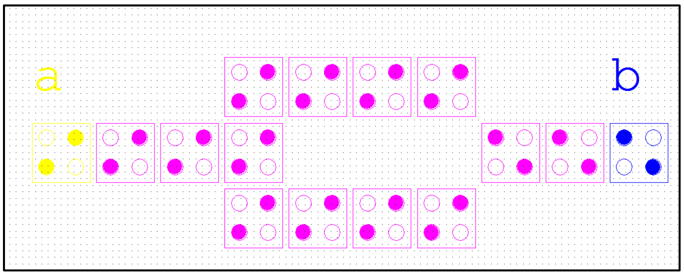


Figure 4 Inverter wire

III. DESIGN OF INVERTER

The inverter is the basic component for all circuits which perform inversion operation.



The Figure shown in 5 represents the schematic layout of an inverter based on Quantum Cellular Automata (QCA) design. Here the number of cells used is 15. The *a* represents the Input to the inverter, also *b* represents the inverted result of *a*. Figure 6 shows the corresponding simulation result for the inverter

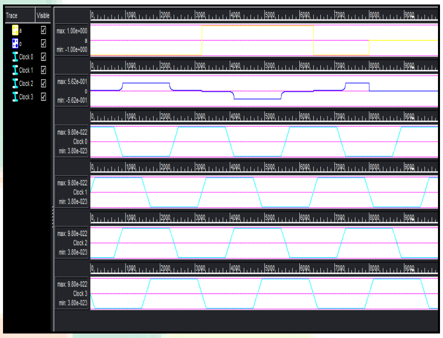


Figure 6 Simulation Result of an Inverter

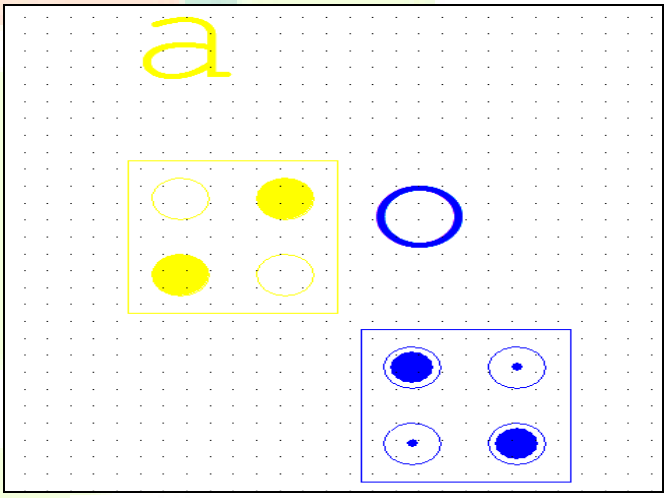


Figure 7 Proposed Inverter Designs

The Figure shown in 7 represents the schematic layout of an inverter based on Quantum Cellular Automata (QCA) design. Here the number of cells used is 2. The *a* represents the Input to the inverter, also *o* represents the inverted result of *a*. In this inverter design two quantum cells are placed in 45^0 so the input signal got inverted and produce the output signal. No of cells got reduced with use of this technique.

IV. DESIGN OF COMBINATIONAL CIRCUITS

A. Half Adder

The half adder is a combinational circuit that performs addition of two bits. It is designed conventionally by EXOR and AND gates. When two inputs A and B are added, the Sum and Carry outputs are produced according to the truth table.The logic function for half-Adder is *Sum = A'B+AB'* and Carry = AB. The majority gate expression for above equation is Sum = M (M (A, B', 0), M (A', B, 0), 1) and Carry = M ((A, B, 0)

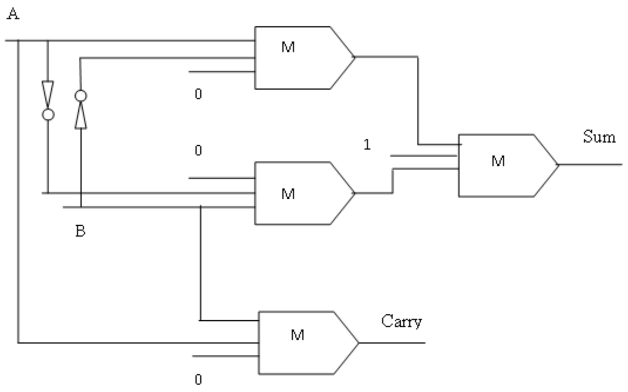


Figure 8 Majority Logic Design Of Half Adder

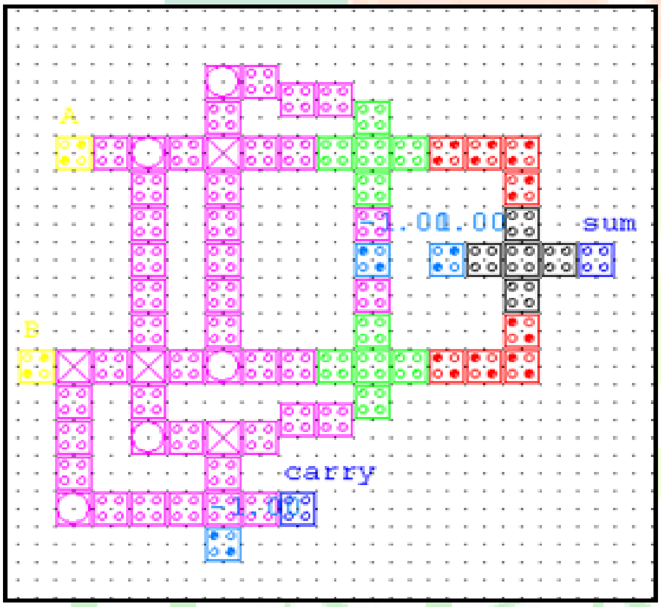


Figure 9 QCA layout of Half Adder

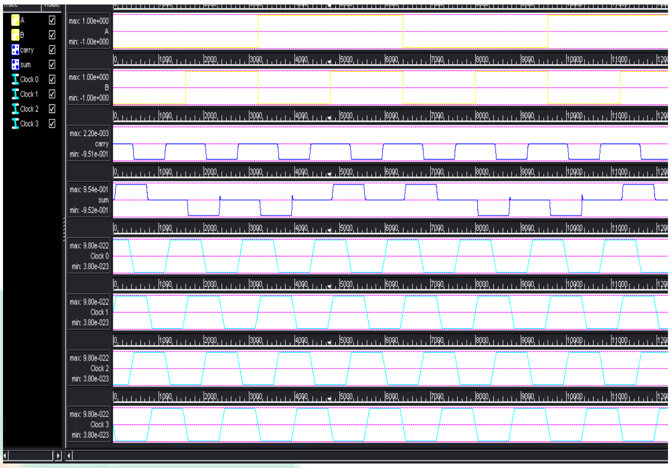


Figure 10 Simulation Result Of Half Adder

The Figure shown in 9 represents the schematic layout of a half adder based on Quantum Cellular Automata (QCA) design. Here the number of cells used is 77. The *A, B* represents the Inputs to the half adder, also *OUT* represents the sum and carry result of inputs *A,B*. Figure 10 shows the corresponding simulation result for the half adder.

B. Full Adder

The full adder circuit is implemented by digital logic gates. When three inputs A, B and C are added, the Sum and Carry outputs are produced according to the truth table.

Majority logic of carry:

Cout = AB+BC+AC
= M (M (B, M(A,C,1) ,0) , M(A,C,0) , 1)
= M(A,B,C)

Majority logic of sum:

Sum= ABC+A'B'C+A'BC'+AB'C'

Direct implementation:

Sum = M(M(A,M(M(B,C,0), M(B',C',0) ,1, 0),

Majority gates:

M (A' M (M (B',C,0),1) ,0),1)

Reduction Technique:

Sum = ABCin+A'B'Cin+A'BC'in+AB'C'in
= (A.B + A'.B') Cin + (A'.B + A.B') C'in
= [A.B + A '.B ' + A.C ' in + A '.C ' in + B.C ' in + B ' C ' in] Cin + (A '.B + A.B ') C ' in
= [(A '.B ' + A '.C ' in + B '.C ' in) + (A.B + A C ' in + B C ' in)] Cin +(A '.B + A.B') C ' in
= [(A '.B ' + A '.C ' in + B '.C ' in) + (A.B + A C ' in + B C ' in)] Cin + (A.C 'in + B. C 'in) +(A'. in + B '.C 'in)
= [(A '.B ' + A '.C ' in + B '.C ' in) Cin + (A.B + AC ' in + B C ' in)] Ci+ (AB + A.C 'in + B.C 'in) (A '.C 'in + B '.C 'in +A 'B ')
=M (A, ' B ',C 'in) .Cin + M(A,B,C ' in). Cin +M(A ',B ',C ' in) M(A,B,C 'in)
= M [M (A ',B ',C ' in), Cin, M(A,B,C ' in)]

Sum =M [C ' out, Cin, M (A, B, C ' in)]

This reduction technique is used to reduce the number of majority gates. The Figure shown in 11 represents the majority logic diagram of a full adder

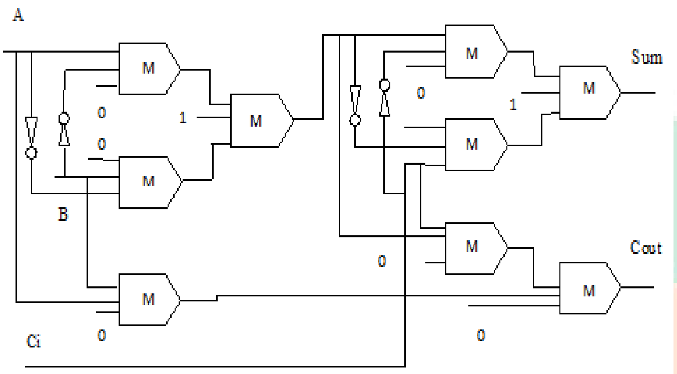


Figure 11 Majority Logic Design of Full Adder

The Figure shown in 12 represents the schematic layout of a full adder based on Quantum Cellular Automata (QCA) design. Here the numbers of cells used are 122. The *A,B,CIN* represents the Inputs to the full adder, also *sum* and *cout* represents the sum and carry out result of inputs *A,B,CIN.* Figure 13 shows the corresponding simulation result for the full adder

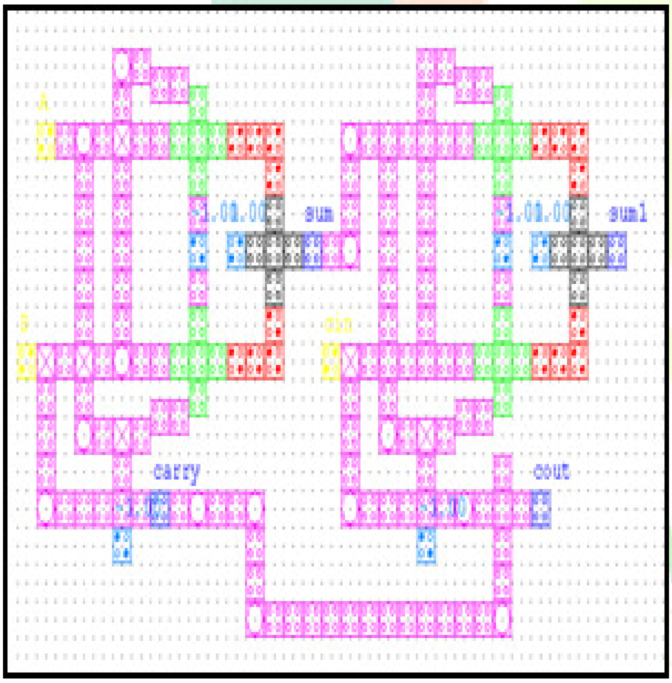


Figure 12 QCA layout of Full Adder

Figure 13 Simulation Result Of Full Adder

C. Ripple Carry Adder

In the 4 bit Ripple carry adder circuit, the input to each full-adder will be *Ai, Bi* and *Ci,* and the outputs will be *Si, Ci+1*, where *i* varies from 0 to 3 as shown in the figure 14 Also the carry output of the lower order stage is connected to the carry input of the next higher order stage. Hence this type of the adder is called ripple-carry adder.

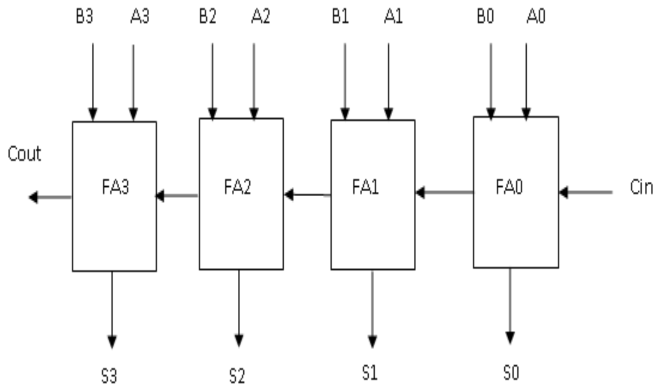


Figure 14 Ripple Carry Adder

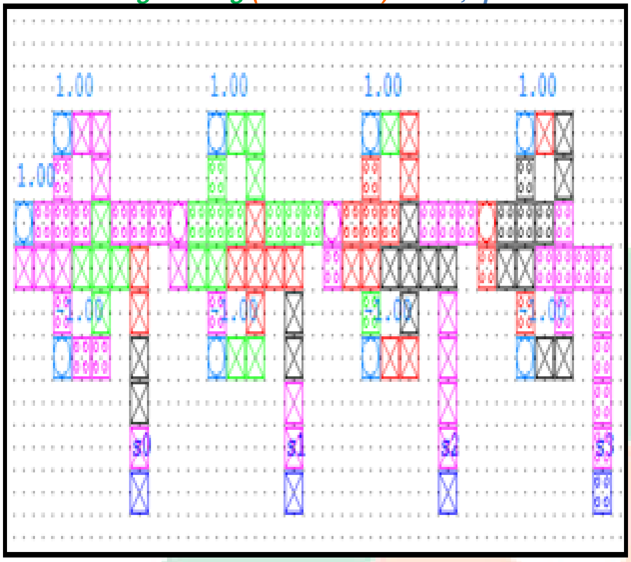


Figure 15 QCA layout of Ripple Carry Adder

Figure 16 Simulation Result Of Ripple Carry Adder

The Figure shown in 15 represents the schematic layout of an full adder based on Quantum Cellular Automata (QCA) design. Here the number of cells used is 125. The *A0,B0, A1,B1, A2,B2, A3,B3,CIN* represents the Inputs to the ripple carry adder, also *S0,S1,S2,S3* and *cout* represents the sum and carry out result of inputs. Figure 16 shows the corresponding simulation result for the ripple carry adder

D. Serial Adder

The parallel adder performs the addition of two binary numbers at a relatively faster rate, the disadvantage of the parallel addition is that it requires a large amount of logic circuitry. This increases in direct proportion with the number of bits in the number being added. On the other hand, in serial addition, the addition operation is carried out bit-by-bit as shown in the figure 17. Therefore, the serial adder requires simpler circuitry than a parallel adder but results in a low speed of operation.

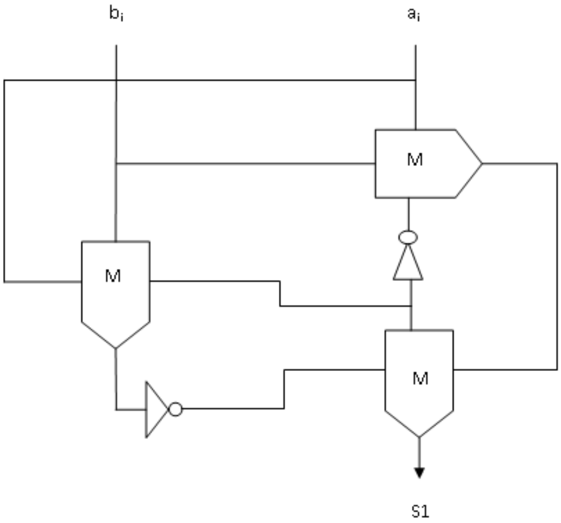


Figure 17 Serial Adder

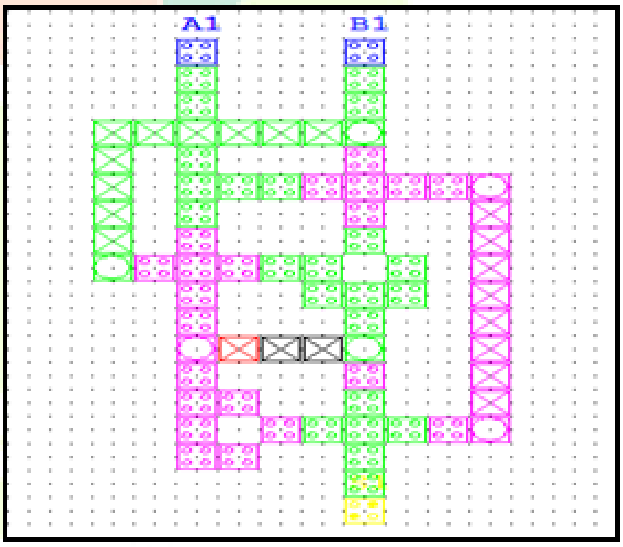


Figure 18 QCA layout of Serial Adder

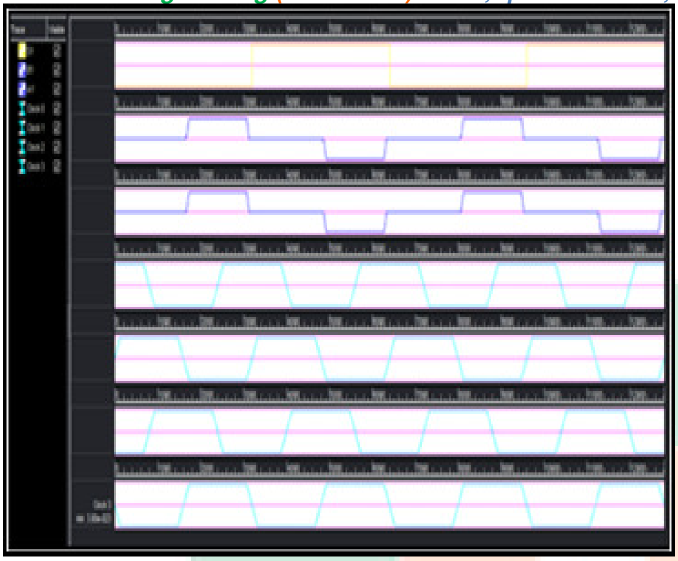


Figure 19 Simulation Result of Serial Adder

The Figure shown in 18 represents the schematic layout of an serial adder based on Quantum Cellular Automata (QCA) design. Here the number of cells used is 74. The *A1, B1* represents the Inputs to the serial adder, also *sum* and *cout* represents the sum and carry out result of inputs *A1, B1.* Figure 19 shows the corresponding simulation result for the serial adder.

V. CONCLUSION

In this project, combinational circuits have been proposed. The important combinational circuits like half adder, full adder, ripple carry adder and serial adder are designed and tested for stuck at 0,1 fault with use of majority logic results are compared which has less simulation time and area. The designs are simulated and tested using QCAD designer and the number of cells and the simulation time are calculated for the designed circuits and the results are compared between base paper method and 45^0 rotation method from the obtained result. The area required for realizing each circuits are minimized. Since the number of cells are in minimum, which decreases the power dissipation. Thereby, the performance of the circuit is greatly improved in QCA.

REFERENCES

[1] Himanshu Thapliyal,Saurabh Kotiyal and Nagarajan Ranganathan,"Design of Testable Reversible Sequential Circuits ,“,” IEEE Trans. Very Large Scale Integration (Vlsi) Systems, Vol. 21, No. 7, July 2013.

[2] J. Ren and V. K. Semenov, “Progress with physically and logically reversible superconducting digital circuits,” IEEE Trans. Appl. Superconduct.,vol. 21, no. 3, pp. 780–786, Jun. 2011S.

[3] F. Murphy, M. Ottavi, M. Frank, and E. DeBenedictis, “On the design of reversible QDCA systems,” Sandia National Laboratories, Albuquerque, NM, Tech. Rep. SAND2006-5990, 2006.

[4] H. Thapliyal and N. Ranganathan, “Reversible logic-based concurrently testable latches for molecular QCA,” IEEE Trans. Nanotechnol., vol. 9, no. 1, pp. 62–69, Jan. 2010.

[5] P. Tougaw and C. Lent, “Logical devices implemented using quantum cellular automata,” J. Appl. Phys., vol. 75, no. 3, pp. 1818–1825, Nov. 1994.

[6] P. Tougaw and C. Lent, “Dynamic behavior of quantum cellular automata,” J. Appl. Phys., vol. 80, no. 8, pp. 4722–4736, Oct. 1996.

[7] M. B. Tahoori, J. Huang, M. Momenzadeh, and F. Lombardi, “Testing of quantum cellular automata,” IEEE Trans. Nanotechnol., vol. 3, no. 4, pp. 432–442, Dec. 2004.

[8] G. Swaminathan, J. Aylor, and B. Johnson, “Concurrent testing of VLSI circuits using conservative logic,” in Proc. Int. Conf. Comput. Design, Cambridge, MA, Sep. 1990, pp. 60–65.

[9] E. Fredkin and T. Toffoli, “Conservative logic,” Int. J. Theor. Phys., vol. 21, nos. 3–4, pp. 219–253, 1982.

[10] P. Kartschoke, “Implementation issues in conservative logic networks,” M.S. thesis, Dept. Electr. Eng., Univ. Virginia, Charlottesville, 1992.

[11] G. Swaminathan, “Concurrent error detection techniques using parity,” M.S. thesis, Dept. Electr. Eng., Univ. Virginia, Charlottesville, 1989.

[12] H. Thapliyal, M. B. Srinivas, and M. Zwolinski, “A beginning in the reversible logic synthesis of sequential circuits,” in Proc. Int. Conf. Military Aerosp. Program. Logic Devices, Washington, DC, Sep. 2005, pp. 1–5.

[13] S. Mahammad and K. Veezhinathan, “Constructing online testable circuits using reversible logic,” IEEE Trans. Instrum. Meas., vol. 59, no. 1, pp. 101–109, Jan. 2010.

[14] H. Thapliyal and N. Ranganathan, “Design of reversible sequential circuits optimizing quantum cost, delay and garbage outputs,” ACM J. Emerg. Technol. Comput. Syst., vol. 6, no. 4, pp. 14:1–14:35, Dec. 2010.

[15] M. Hasan, A. Islam, and A. Chowdhury, “Design and analysis of online testability of reversible sequential circuits,” in Proc. Int. Conf. Comput. Inf. Technol., Dec. 2009, pp. 180–185.

[16] M. Pedram, Q. Wu, and X. Wu, “A new design for double edge triggered flip-flops,” in Proc. Asia South Pacific Design Autom. Conf., 1998, pp. 417–421.

[17] X. Ma, J. Huang, C. Metra, and F. Lombardi, “Reversible gates and testability of one dimensional arrays of molecular QCA,” J. Electr. Test., vol. 24, nos. 1–3, pp. 1244–1245, Jan. 2008.

[18] M. Momenzadeh, M. Ottavi, and F. Lombardi, “Modeling QCA defects at molecular level in combinational

circuits," in Proc. DFT VLSI Syst., Monterey, CA, Oct. 2005, pp. 208–216.

[19] M. Ottavi, L. Schiano, and F. Lombardi, "HDLQ: A HDL environment for QCA design," ACM J. Emerg. Technol., vol. 2, no. 4, pp. 243–261, Oct. 2006.Fijany and B. Toomarian, "New design for quantum dots cellular automata to obtain fault tolerant logic gates," J. Nanoparticle Res., vol. 3, no. 1, pp. 27–37, 2001.

[20] M. Dalui, B. Sen, and B. K. Sikdar, "Fault tolerant QCA logic design with coupled majority-minority gate," Int. J. Comput. Appl., vol. 1, no. 29, pp. 81–87, Feb. 2010.