Compact, High Speed and Low power Decoders for Future Generation System Building

Prameela Kumari N.
Department of Electronics and Communication
REVA Institute of Technology and Management,
Bangalore, India

K.S. Gurumurthy
Department of Electronics and Communication
REVA Institute of Technology and Management
Bangalore, India

ABSTRACT
Electronic system building has become highly competitive from the point of reducing area, power and increasing the speed. To address these issues, many technologies are being tried. Quantum Dot Cellular Automation is one such technology. This technology is in its infant state as far as its physical implementation and verification are concerned. However the researchers have come out with theoretical models and proposed many compact, fast and low power dissipating digital blocks. Decoders are one of the standard combinational modules used as the basic building blocks for efficient digital system design. Here in this paper we implement decoders in QCA using different techniques and analyze them with respect to area, time and energy. From the implementation and simulation results obtained using QCADESIGNER version 2.0.3 we have observed that, the 2:4 decoders implemented with and without the enable input using both single layer and multilayer techniques utilize minimum area, time and energy.

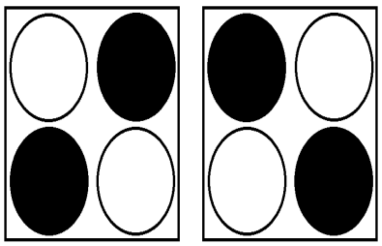
Keywords
Quantum Dot Cellular Automata (QCA), Decoder, Coincident Decoding, Tree Decoding, Universal Module.

1. INTRODUCTION
Quantum Dot Cellular Automation is one of the emerging nanotechnologies suitable for low power, high performance and ultra dense circuit designs operating at Terra Hertz [1]. Many universal structures like Majority-Not, Nand-Nor, And-Or, Multiplexers etc., are proposed to realize the circuits. Major works are also carried out utilizing these universal structures. However decoders along with an 'Or' gate which forms an Universal module is hardly explored. Hence in this paper we present the design of decoders in QCA using different techniques.

A few research works carried out with respect to QCA decoders are available in the literature. A 4:16 decoder implemented using 3-input majority gate was first proposed and designed in QCA to address a serial write, parallel read memory by Peskin et.al [2]. But the work mainly focused on the memory design and decoder was just a part of the memory access. Ottavi et.al, [3] also proposed a 4:16 decoder. This 4:16 decoder was split into two 2:4 row and column sub-decoders. These sub-decoders enabled the row and column lines, which in turn activated a particular cell in the memory. Tung et.al, [4] have proposed the design of linear row and column decoders for CLBs. Kianpour et.al. [5] have proposed the implementation of modular decoders in QCA. Kondwani et.al, [6] have proposed the design of decoder structure using inverter chain. In this paper we mainly focus on the design and implementation of optimized and well structured 2:4, 3:8 and 4:16 size decoders and analyze them with respect to area and performance.

The organization of the paper is as follows: Section 2 presents an overview of QCA, section 3 presents a general discussion on QCA decoders. Here we design the decoders using single layer and multi layer methods and analyze them with respect to area and performance. In section 4 we present the conclusion, followed by the references in section 5.

2. OVERVIEW OF QCA
C.S.Lent.et.al., [7] first proposed and experimentally demonstrated the possibility of Quantum Dot Cellular Automata. The basic element of QCA is a Cell, which is used to realize a circuit. It consists of two quantum dots at the center to control the switching and four cells placed at the corners to realize the logic. Polarization of the electrons in these quantum dots represents the binary information as '0' and '1' as shown in Figure.1 [8].



(a) **(b)**

Fig. 1. QCA Cell Polarization: (a) Binary '1', (b) Binary '0'.

There are four clock signals in QCA, which are phase shifted by 90° with respect to each other. Further each clock is subdivided into four switching zones, namely release, relax, switch and hold.

In QCA, cells are responsible for both processing and transmitting the information [9]. A Majority gate [MG] is the basic processing element in QCA. Here an input cell accepts the inputs, computes their majority at the device cell and transmits the output to the output cell. As shown in the Figure.2 the inputs A and B are '1' while the input C is '0'. Hence their majority value '1' is computed at the device cell and transmitted to the output cell. In QCA, circuits can be realized using two different techniques: 1. Single layer using coplanar crossing, which requires 45'and 90'oriented cells. 2. Multi layer rossing which requires only 90'oriented cells with multiple layers to implement the crossover.

3. QCA DECODERS

Any combinational circuit that has 'N' binary inputs and '2^n' binary ouputs is an 'N' input binary decoder as shown in Figure 3. In a decoder, at any point of time only one of the output is '1' and the remaining are '0'. Hence a decoder acts as a code converter that converts a binary code into 1-out-of-2^n codes. An enable input 'En' can be used to control the output of a decoder. When the 'En' input is '0', irrespective of the input combination, all the outputs are '0'. Only when 'En' is '1' will the outputs follow their input combination. As a decoder output corresponds to a switching function whose value is '1' for exactly one input combination, it can also be represented by a minterm which can be specified as:

Inputs: $L = (L_{n-1}, \ldots\ldots\ldots, L_0), L_j \in \{0,1\}$

$En \in \{0, 1\}$

Outputs: $Y = (Y_{2^n-1}, \ldots\ldots\ldots, Y_0), Yi \in \{0, 1\}$

Function: $Y_i = En * m_i(L), i = 0, 1, 2, \ldots\ldots 2^n - 1.$

where $m_i(L)$ is the i^{th} min term of the 'N' variables of 'L'. To realize this we require 'N' NOT gates and '2^n' AND gates. Hence for a 2:4 decoder we require '2' NOT gates and '4' AND gates as shown in Figure.4.

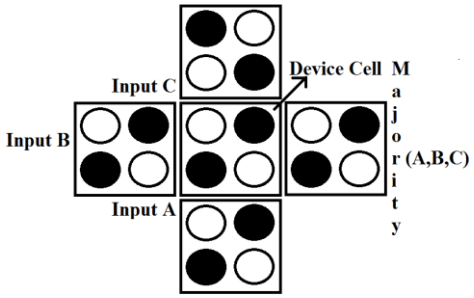


Fig. 2. QCA Majority Gate.

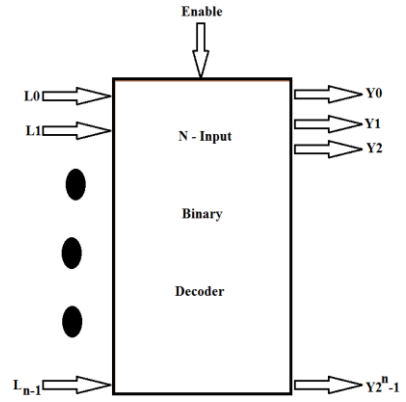


Fig. 3. N - Input Binary Decoder.

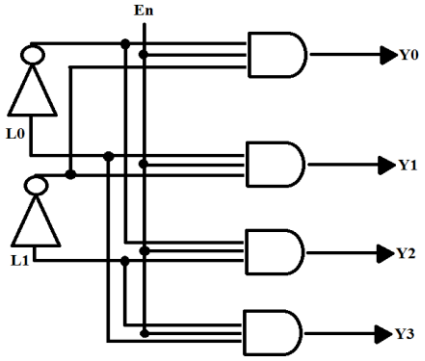


Fig. 4. Gate Level Realization of 2:4 Decoder

Figure.5 shows the QCA layout of a 2:4 decoder implemented using the single layer coplanar crossover method. In the first step the inputs 'a' and 'b' are transmitted using the inverting QCA cell chain when clock '0' is active. In an inverting cell chain we use 45° oriented QCA cells. Hence every cell complements its previous cell value. Which means if the first cell has a value '0'then the second cell will have a '1', the third will have a '0' and so on. In order to extract the value from the inverting chain, we have placed a 90° oriented cell between an even and an odd cell and to extract a complemented value the cell is placed between an odd and an even cell. In the second step we have implemented the four AND gates using the majority logic to obtain a′b′, a′b, ab′ and ab outputs respectively representing the four decoder outputs when the clock 1 is active. Overall this design requires 101 cells and utilises only 0.5 clock cycles (i.e. clock 0 and clock 1) to obtain all the 4 outputs simultaneously. Hence the proposed 2:4 single layer decoder is the smallest with the least area and also the fastest with least number of clock cycles[5][6][10].With an enable input this 2:4 decoder requires 103 cells and 1 clock cycle (i.e. clock 0, clock 1, clock 2, clock 3) as shown in Figure.6. It hardly takes 2 extra cells and hence the smallest in comparision with the avaliable decoders in the literature[5][6][10] which are implemented without the enable input . Figure.7 shows the QCA layout of a 3:8 decoder with enable input implemented using 247 cells with 1.75 clock cycles. To implement a 4:16 decoder we require 614 cells and 2.75 clock cycles as shown in Figure.8.

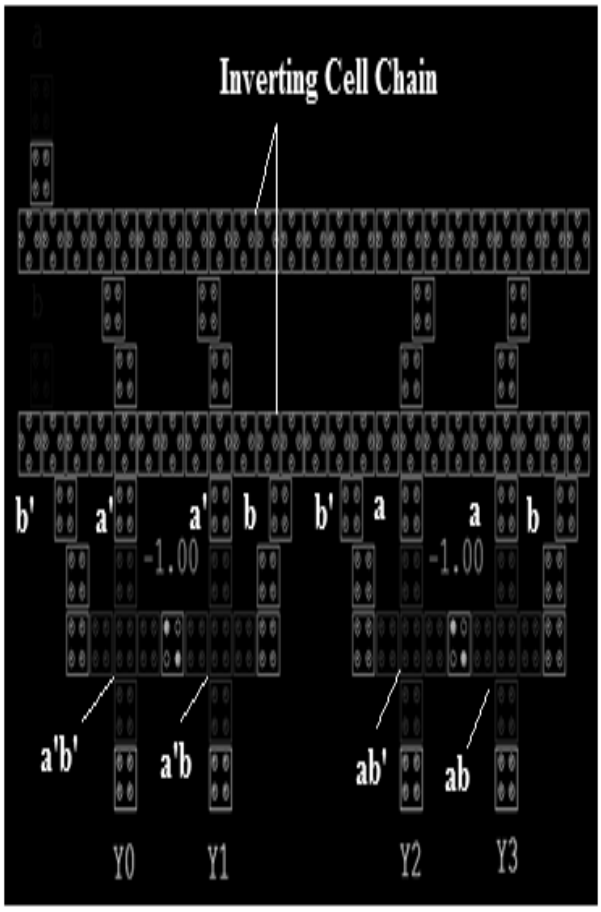


Fig. 5. QCA layout of a 2:4 decoder implemented in a single layer using coplanar crossing.

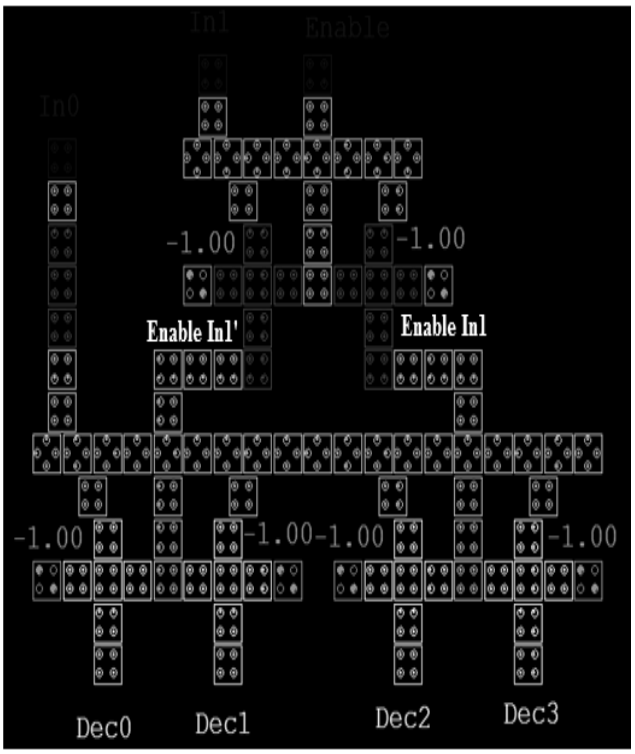


Fig. 6. QCA layout of a 2:4 decoder with enable input implemented in a single layer using coplanar crossing.

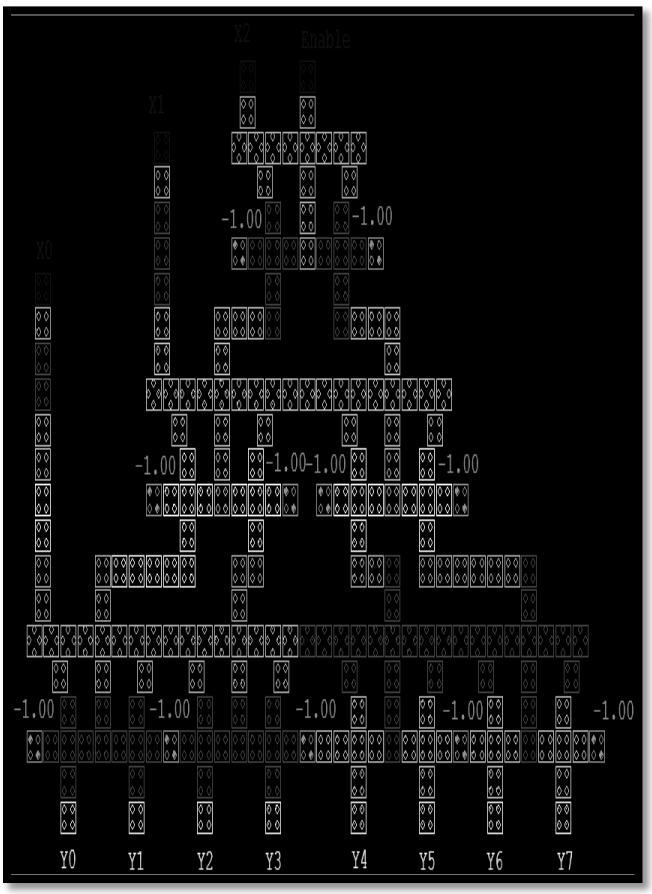


Fig. 7. QCA layout of a 3:8 decoder with enable input implemented in a single layer using coplanar crossing.

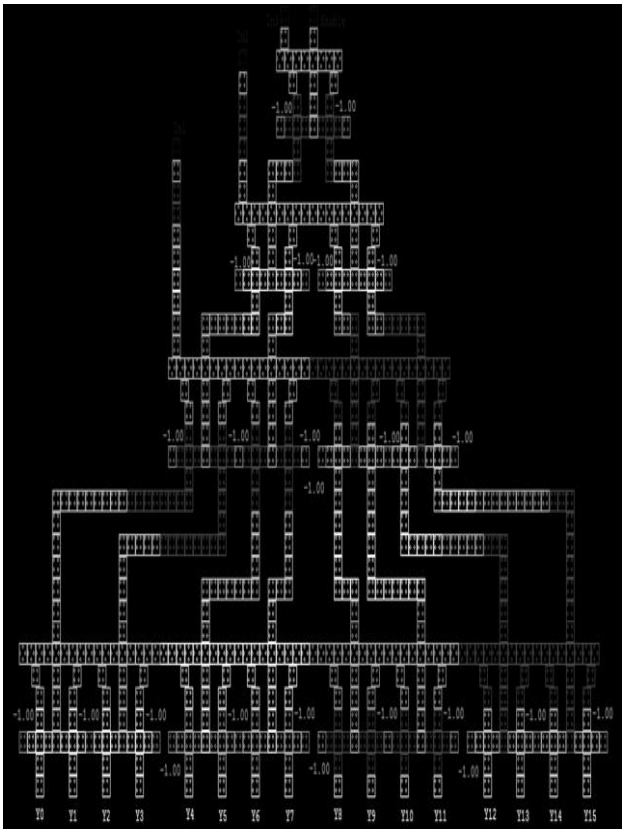


Fig. 8. QCA layout of a 4:16 decoder with enable in a single layer using coplanar crossing.

Using the second technique i.e. multi layer crossover, inorder to implement a 2:4 decoder without the enable input we require 148 cells and 1 clock cycle or 4 clocking zones to obtain all the 4 outputs simultaneously as shown in Figure.9. With the enable input it requires 228 cells and 1.5 clock cycles or 6 clock zones as shown in Figure.10. A 3:8 decoder requires 720 cells and 2.5 clock cycles or 10 clock zones as shown in Figure.11 and a 4:16 decoder requires 1769 cells and 5.25 clock cycles or 21 clock zones as shown in Figure.12. Figure.13 shows the simulation result of 2:4 decoder without the enable input. Results follow the expected truth table output as shown in Table 1.

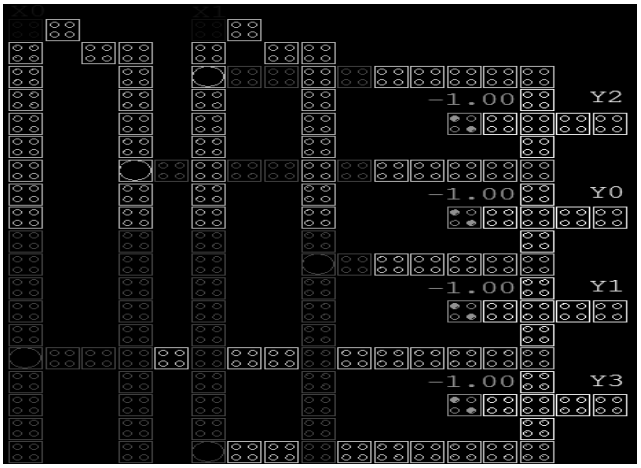


Fig. 9. QCA layout of a 2:4 decoder implemented using multilayer crossover.

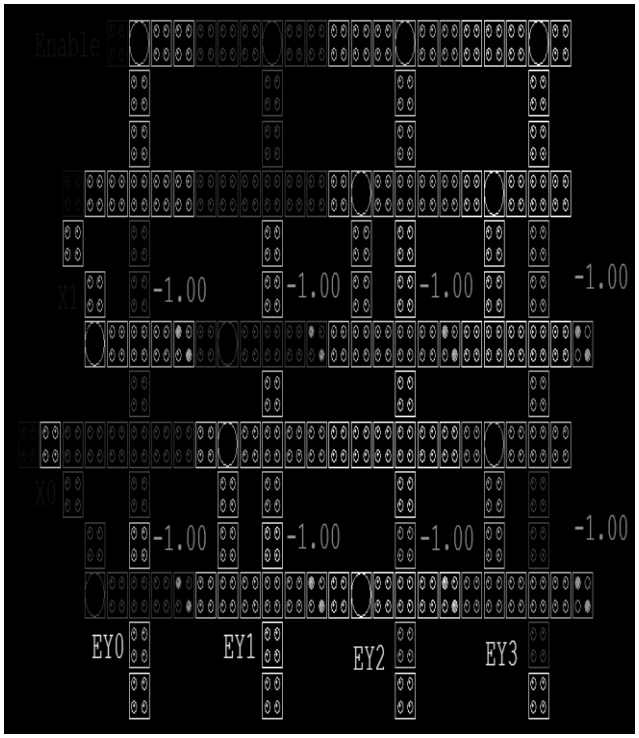


Fig. 10. QCA layout of a 2:4 decoder with enable implemented using multilayer crossover.

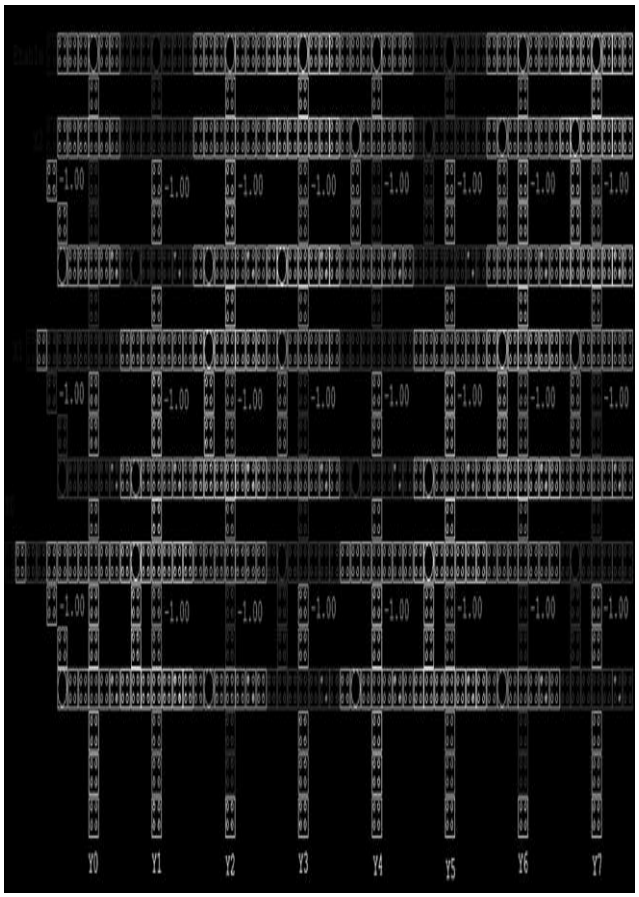


Fig. 11. QCA layout of a 3:8 decoder with enable implemented using multilayer crossover.

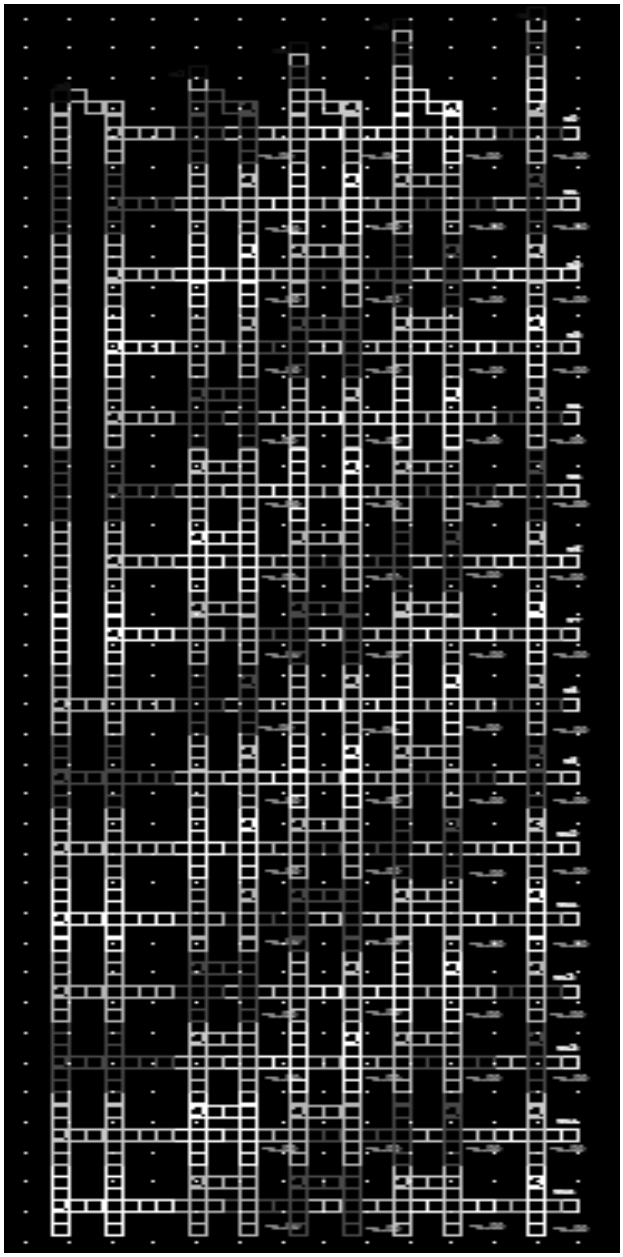


Fig. 12. Layout of a 4:16 decoder with enable implemented using multilayer crossover in QCA.

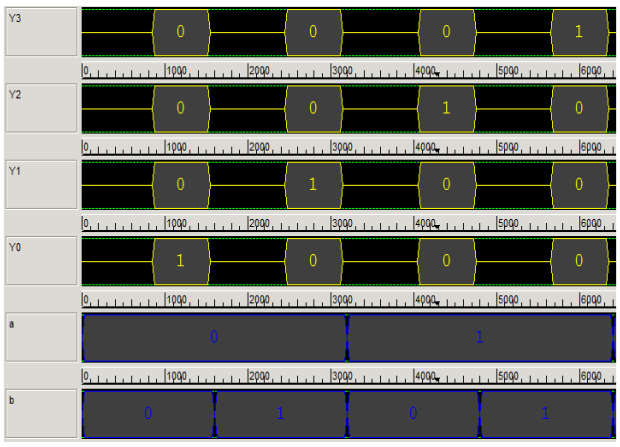


Fig. 13. Simulation Results of 2:4 Decoder without enable input.

TABLE 1. Truth table of 2:4 decoder without enable.

Inputs X1 X0		Output
0	0	Y0
0	1	Y1
1	0	Y2
1	1	Y3

Figure.14 shows the simulation results of 2:4 decoder with the enable input. Results follow the expected truth table output as shown in Table 2.

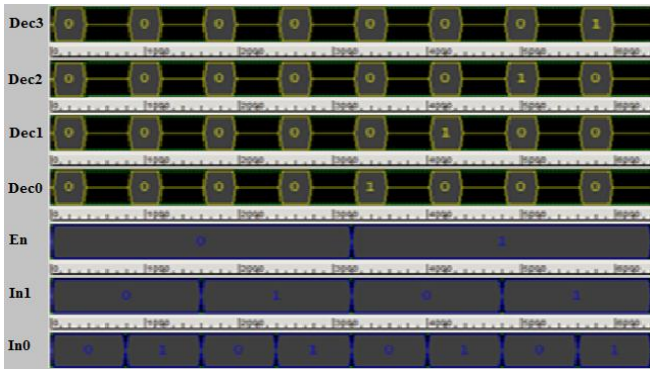


Fig. 14. Simulation Results of 2:4 Decoder with enable input.

TABLE 2. Truth table of 2:4 Decoder with Enable.

Inputs Enable X0	X1		Output
1	0	0	Y0
1	0	1	Y1
1	1	0	Y2
1	1	1	Y3

Figure. 15 shows the simulation results of 3:8 decoder with enable input. Results follow the expected truth table output as shown in Table 3.

TABLE 3. Truth table of 3:8 Decoder with Enable.

Inputs Enable X0	X2	X1		Output
1	0	0	0	Y0
1	0	0	1	Y1
1	0	1	0	Y2
1	0	1	1	Y3
1	1	0	0	Y4
1	1	0	1	Y5
1	1	1	0	Y6
1	1	1	1	Y7

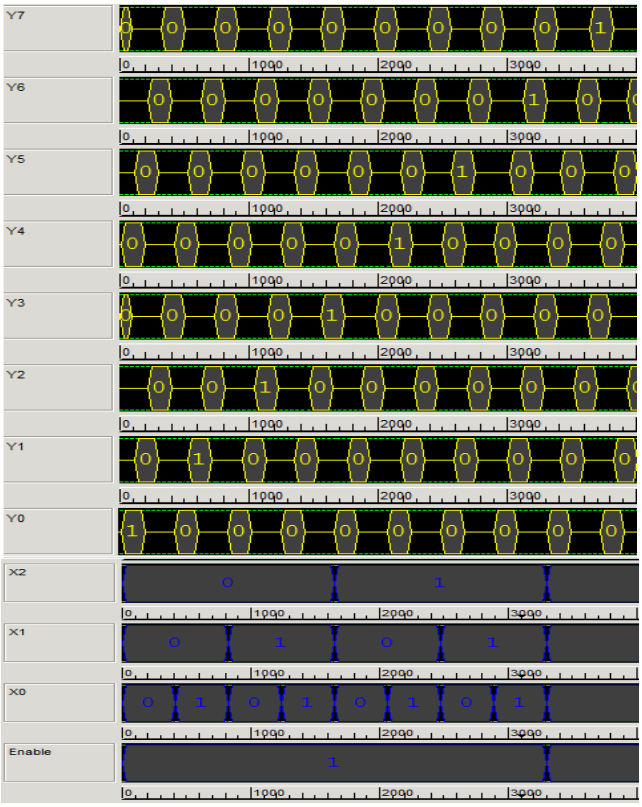


Fig. 15. Simulation Results of 3:8 Decoder with enable input.

Figure. 16 shows the simulation results of 4:16 decoder with enable input. Results follow the expected truth table output as shown in Table 4.

TABLE 4. Truth Table Of 4:16 Decoder With Enable

Inputs Enable X1	X3	X2		X0	Output
1	0	0	0	0	Y0
1	0	0	0	1	Y1
1	0	0	1	0	Y2
1	0	0	1	1	Y3
1	0	1	0	0	Y4
1	0	1	0	1	Y5
1	0	1	1	0	Y6
1	0	1	1	1	Y7
1	1	0	0	0	Y8
1	1	0	0	1	Y9
1	1	0	1	0	Y10
1	1	0	1	1	Y11
1	1	1	0	0	Y12
1	1	1	0	1	Y13
1	1	1	1	0	Y14
1	1	1	1	1	Y15

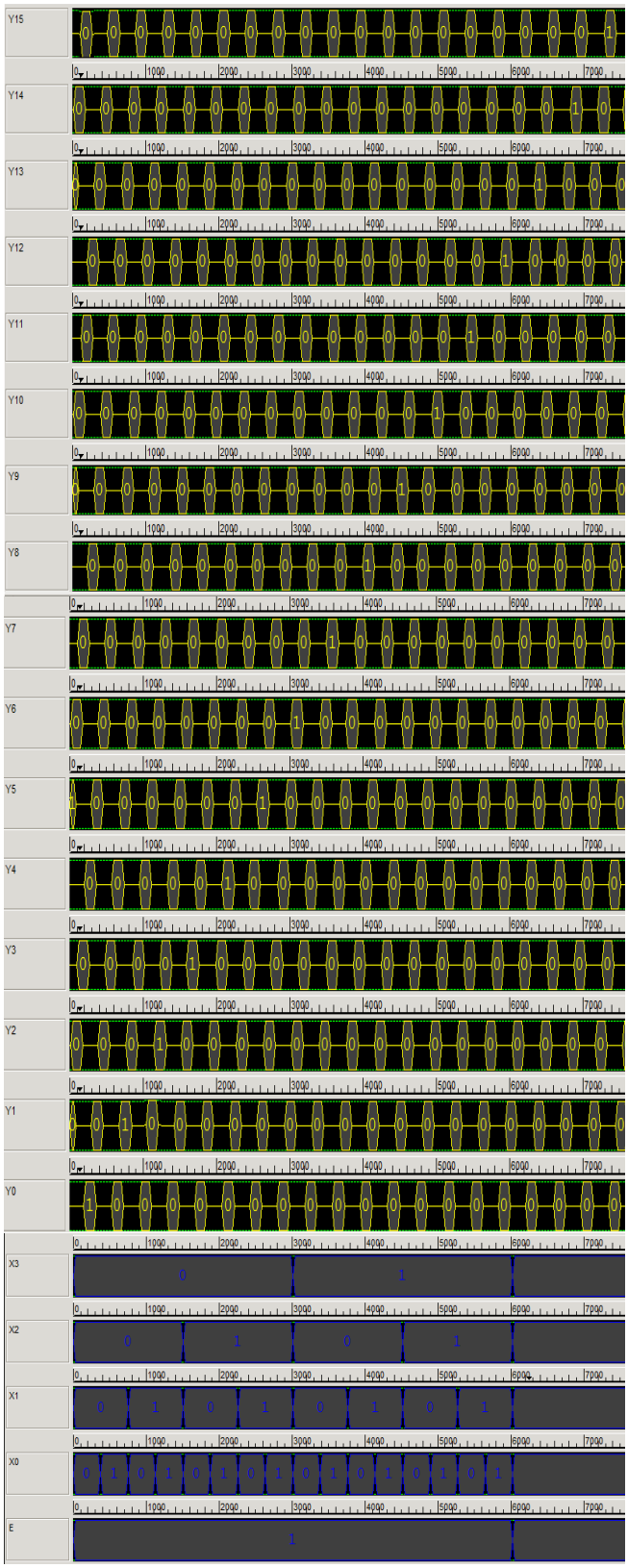


Fig. 16. Simulation Results of 4:16 Decoder with enable input.

Figure.17 and Figure.18 shows the comparision of different size of decoders implemented using Single and Multi Layer crossover techniques. It is very clear from the graph that there is a drastic increase in the cell count by around 70% for every increase in the order of the decoder. Followed by Figure.19 and Figure.20 which shows their performance evaluation. From the graph it is understood that the required time doubles as the order of the decoder increases. Figure.21 and Figure.22 shows the comparision of energy consumption. It increases by 60% for every increase in the order of the decoder. From the above analysis it is clearly understood that we need to adopt different methods to decrease the number of cells and the required clock cycles as the order of the decoder increases.

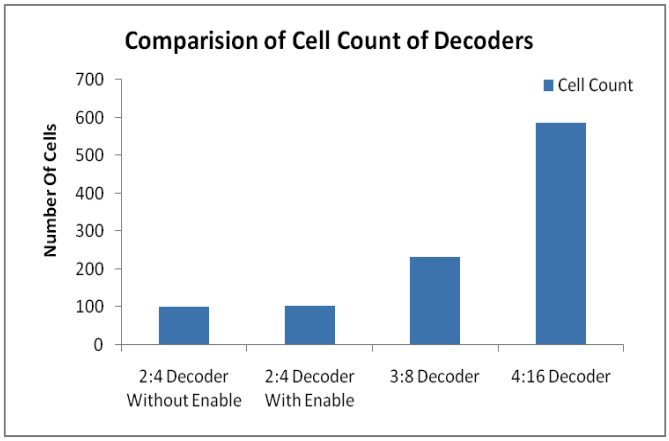


Fig. 17. Comparision of Cell Count for decoders implemented using Single Layer

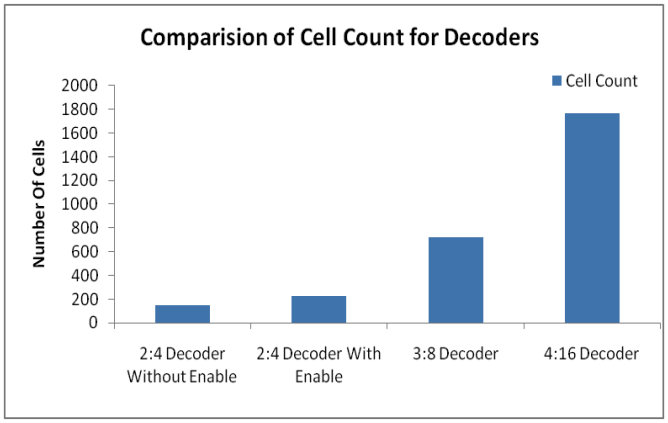


Fig. 18. Comparision of Cell Count for decoderimplemented using Multi Layer.

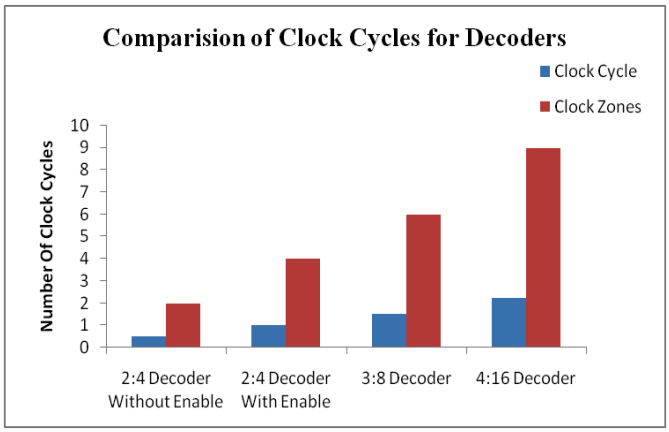


Fig. 19. Comparision of Clock Cycle for decoder implemented using Single Layer.

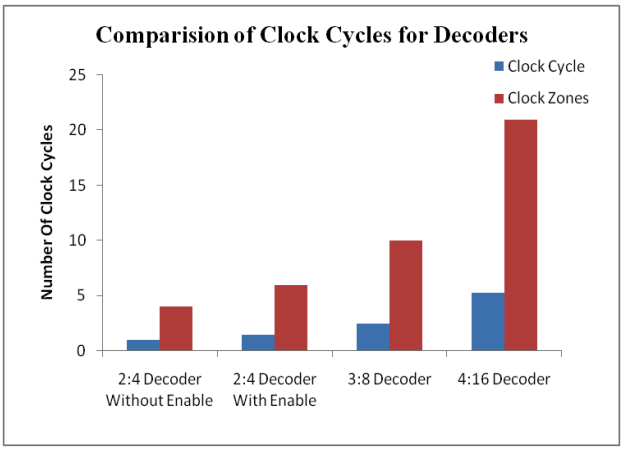


Fig. 20. Comparision of Clock Cycle for decoder implemented using Multi Layer.

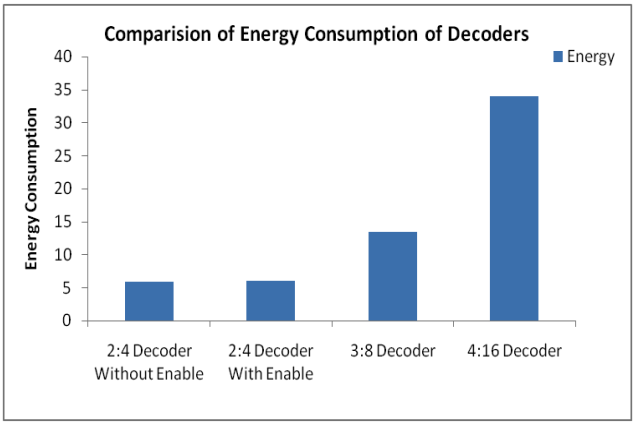


Fig. 21. Comparision of Energy Consumption for decoders implemented using Single Layer.

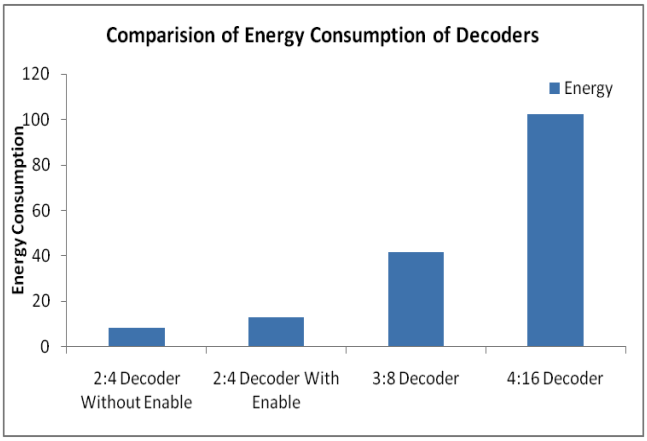


Fig. 22. Comparision of Energy Consumption for decoders implemented using Multi Layer.

4. CONCLUSION

In this paper we have realized 2:4, 3:8 and 4:16 size QCA decoders using single layer crossover and multilayer crossover techniques. The 2:4 QCA decoder implemented without the enable input in a single layer utilizes only 101 cells and 0.5 clock cycles. While the 2:4 decoder implemented with the enable input uses only 103 cells and 1 clock cycle. These two decoders utilize minimum area and time when compared to the decoders available in the literature as shown in Table V. From the obtained simulation results it is observed that area, time and energy increase by a factor of 70%, 50% and 60% respectively as the size of the decoder increases. Hence we will implement higher order decoders using different techniques and analyze them with respect to area time and energy as a further extension to this work.

TABLE 5. Functionality of a 2: 4 decoder

Decoder Type	Cells	Clock Cycles
Proposed 2:4 without Enable	101	0.5
Proposed 2:4 with Enable	103	1
[6] 2:4 Without Enable	139	1.25
[10] 2:4 Without Enable	110	0.75

5. REFERENCES

[1] International Technology Roadmap for Semiconductors (ITRS), http://www.itrs.net, (2007).

[2] T.D.Lantz, and E.R.Peskin, A QCA implementation of a configurable logic block for FPGA, in Proceedings on *Third International Conference on Reconfigurable Computing and FPGA's*, (2006), pp. 132 - 141.

[3] M. Ottavi, S.Pontarelli, V.Vankamamidi A.Salsano, F. Lombardi, QCA Memory with Parallel Read / Serial write: Design and Analysis, in IEEE Proceedings on *Circuits Devices and Systems*, vol. 153,(2006), pp. 199-206.

[4] C.C.Tung,R.B.Rungta,E.R.Peskin, Simulation of a QCA based CLB and a Multi - CLB Applications, (FPT 2009), pp. 62-69.

[5] M.Kianpour, R.Sabbaghi-Nadooshan, A Novel Modular Decoder Implementation in Quantum-Dot Cellular Automata (QCA), Paper presented at the International Conference on Nanoscience, Technology and Societal Implications (NSTSI), (2011), pp.1-5.

[6] Makanda, Kondwani, Jun-Cheol Jeon, Improvement of Quantum-Dot Cellular Automata Decoder Using Inverter Chain, Published in Advance sience and technology letters, vol.29,(2013), pp. 227-229.

[7] Craig. S. Lent and P. D. Tougaw, A Device Architecture for Computing with Quantum Dots, Proceedings of *IEEE International conference on Device Technology*, vol.85,(1997), pp. 541-557.

[8] Enrique P. Blair and Craig S. Lent, Quantum-Dot Cellular Automata: An Architecture for Molecular Computing, Paper presented at the IEEE conference on NanoElectronics,(2003),pp. 14-18.

[9] 0. Orlov, I. Amlani, G. Toth, Craig. S. Lent, G. H. Bernstein, and G. L. Snider, Experimental demonstration of a binary wire for quantum-dot cellular automata, Published in Applied Physics Letters, vol.74, (1999),pp. 2875.

[10] Zhou, Rigui, A Logic Circuit Design of 2-4 Decoder Using Quantum Cellular Automata, Published in Journal of Computational Information Systems, vol. 8, (2012),pp. 3463-3469.