

# Design and implementation of two-dimensional crossbar switch scheduler for SoC using quantum dot cellular automata and system Verilog

Amita Asthana<sup>1</sup>, Anil Kumar<sup>2</sup>, Preeti Sharan<sup>3</sup>

<sup>1,2</sup>ASET Lucknow Campus, Amity University, Noida, India

<sup>3</sup>The Oxford College of Engineering, Bangalore

India

<sup>1</sup> researches555@gmail.com, <sup>2</sup> akumar3@lko.amity.edu, <sup>3</sup> sharanpreeta@gmail.com

---

## Abstract

Intelligent Arbiter circuit designed at quantum level is a solution for high-speed data switching circuits with non-blocking properties and miniaturization of the digital circuits at nano-level. In this paper a two-dimensional arbiter design has been proposed using quantum dot cellular automata. The circuit provides a congestion free network for data switching from any in-port to allocated out-port. In the paper a 3x3 arbiter circuit has been designed and analysed using QCA designer software. The paper also presents the design of same arbiter circuit using CMOS technology. To establish a logical discrimination highlighting the prominent features of low power requirements, very less area and high speed QCA circuits a comparison has been made between the two technologies. Also, the paper presents a more factual clock distribution to ease the practical realization (2-DW clocking) of QCA design of the presented grant allocate scheduler using QCA and results have been reported.

**Keywords:** Crossbar Switch, Quantum Dot Cellular Automata, System Verilog, Schedulers for DCX

## Introduction

The increasing demand of fast and compact System on Chip (SoC) has led to development of high density and low power circuits to be integrated on a single chip. In addition to the circuit design to incorporate various logic circuits on a single chip the interconnections between the various circuit blocks also plays a vital role for proper functioning of the fabricated design. The communication between different modules such as microprocessor, memory blocks, interfaced modules require exact algorithms to be developed and implemented [1]. The crossbar switching circuit has been given a due consideration for connecting multiple input and multiple output ports. The connection between the input and output ports depends on the assigned algorithm to the scheduler circuit. Basically, there are three main components of the crossbar switch circuits, (i) the physical connection provider circuit (ii) the queue buffer where the data/packets can be stored and (iii) the arbiter circuit. The arbiter circuit basically decides the connections to be made by the crossbar switch matrix at a time. Various algorithms have been proposed for scheduling the interconnections like round robin [2], parallel iterative matching [3], maximum size matching algorithm [4], iSLIP algorithm with a small change in round robin algorithm (RRA) [5] etc. The important parameter of any arbiter is to improve the contention between different input port to output ports. In recent years there has been an inclination to adopt the RRA architecture of schedulers and numerous designs have been recommended [6]. To curtail the size of the arbiters the new innovations have reached to nano scale where these arbiter performances can be compared to existing techniques. Using Feynman gate reversible quantum dot cellular automata, the nano-communication architecture has also been proposed and computational fidelity has been verified between 1K to 10 K temperature ranges [7]. Serial nano-communication module has also been designed using QCA, with serial to parallel, parallel to serial, hamming code parity checker, hamming generator and Hamming corrector [8]. The design of 4x4 nano router have also been presented along with the components such as demultiplexer, parallel to serial

converter and to achieve the non-blocking behaviour a quadratic crossbar has been proposed [9]. For efficient channel applications data routers have also been designed using QCA in which a data path selector cum router, for efficient channel utilization has been proposed to accelerate data transmission from various sources using pair of signals named as control and select for connection establishment between the source and the destination [10].

The paper is catalogued as follows. First, a quick foundation of QCA technology has been furnished with explanation of few advancements in this ultra-unique spectrum based on QCA. In section 3, basic cell of two-dimensional arbiter architecture has been explained along with the RTL schematic and simulation waveform. In module 4, the arbiter architecture has been extended to 3x3 input-output scheduler circuit. Section 5, the design has been implemented using system Verilog, the RTL schematic and the simulation waveforms have been explained. The same design for the cell scheduler and 3x3 digital cross connects scheduler has been implemented using QCA Designer and the results have been obtained and explained in section 6. Section 7, the power calculation of QCA implementation has been explained. Section 8, represents the cost function analysis of QCA implementation for the schedulers. Results obtained by Verilog implementation and QCA implementation have been explained in section 9 and section 10 respectively. Section 10 also compares the results obtained with the two technologies in addition to the 2-DW clocking scheme. s

## Quantum Dot Cellular Automata (QCA Technology)

The QCA circuits are made by the fundamental unit cell in which there are four quantum dots are positioned at the four corners providing four possible accommodations to two electrons. For minimum energy condition the two electrons occupy the diagonal positions but they are capable to make quantum tunnels into their positions. The two possible states can be represented as logic 0 and logic 1 by encrypting as  $p(-1)$  and  $p(+1)$ , thus the QCA cells can be a great tool to implement the Boolean function. The array of cells is the QCA wires and QCA gates are inverters and majority gates. The function of majority gate is defined as  $M(p, q, r) = pq + qr + pr$ . Four phase clocking (switch, hold, release and relax) is used for designing QCA circuits with two types of clocking floorplans (columnar region and zone region). For physical implementation the columnar approach is more practical. In semiconductor QCA circuit development there are two crossing options, first is the multilayer crossovers and second one is coplanar crossings. Crossovers of two separate wires is a challenge to design QCA circuits, as if coplanar crossing is preferred then it requires a meticulous alignment during fabrication process. To implement non coplanar crossovers two silicon layers inside a SiGe device was proposed. The cost of multilayer crossover is greater than the coplanar crossover cost, given by the following relationship,

$$C_{ml} = m \cdot C_{cp}$$

where,  $C_{ml}$  and  $C_{cp}$  are the costs of coplanar and multilayer crossings respectively. The delay of any nano-circuit implemented by QCA must be comprised with delay factor. In CMOS technology the delay of the circuit is important and calculated as one complete cycle, similarly in QCA circuit minimum delay is a clocking zone delay and calculated as one fourth (1/4) of a clock cycle.

## Ripple Carry Scheduler (RCS)

The scheduler for cross bar switch as shown in figure consists of three AND gates and two NOT gates. Whenever, the request for a particular connection is generated the scheduler serves the request by making the allocate output as high. The inverted output of allocate signal makes the R2 and R4 as low. As far as the allocate output is low the output R2 will be exactly equal to R1 and R4 output to be same as R3. This

explains the fundamental working of the basic cell of the scheduler circuit which has been simulated. The truth table for the basic cell circuit can be explained as follows as per table 1,

Table 1 Truth table of basic module of the scheduler

Request	R1	R3	Allocate	R2	R4
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	0	0	0
1	0	1	0	0	1
1	1	0	0	1	0
1	1	1	1	0	0

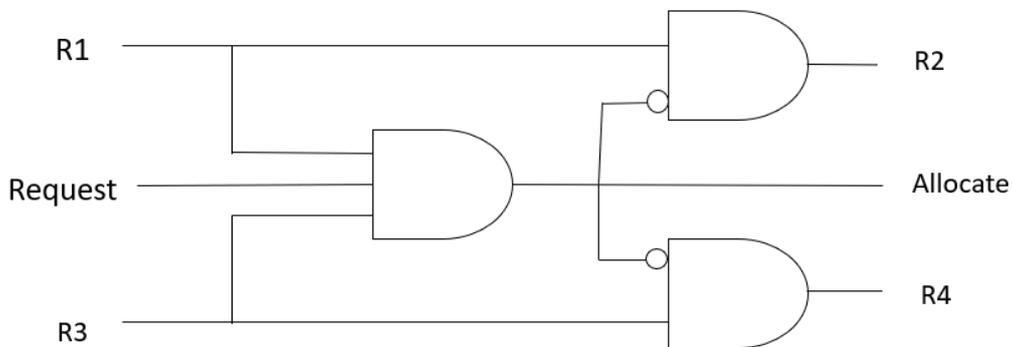


Figure1. Design of single scheduler cell

Figure1 presents the basic cell module of the chip scheduler which comprises of three AND gates, two double inputs and one for three inputs. The two inverters are placed in such position to make R2 and R4 to be zero whenever the allocate goes to logic 1.

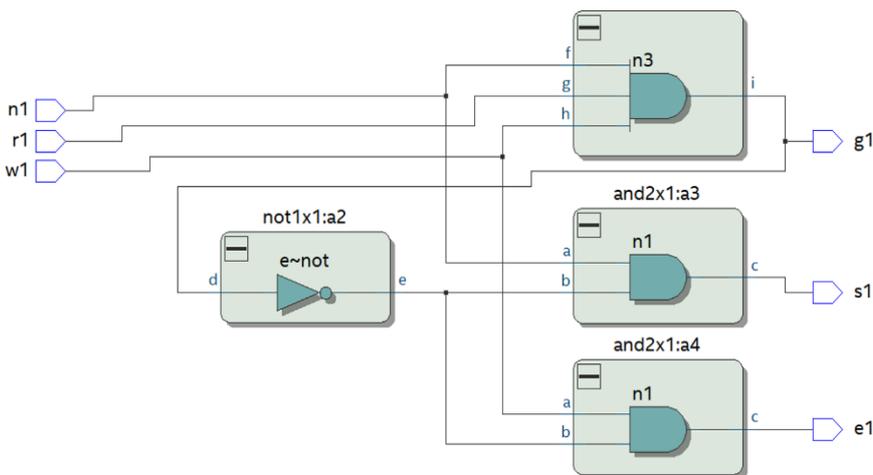


Figure 2. Scheduler cell RTL schematic

The RTL schematic of cell scheduler has been presented in figure 2. Here the allocate output has been represented by s1 and request to be represented as r1. The n1, g1, w1 and e1 have been represented by R1, R2, R3 and R4 respectively.

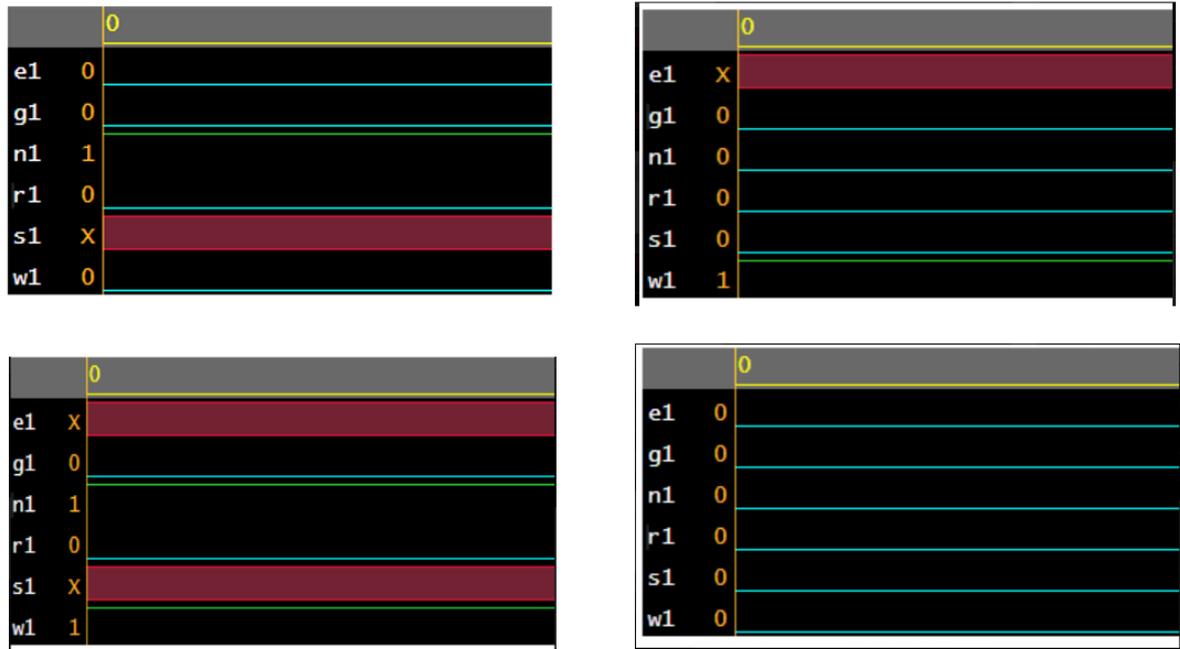


Figure 3. Simulation waveform of Scheduler cell

The simulation result for cell scheduler circuit comprising of simple three AND gates and one NOT gate using Verilog has been shown in figure 3.

### 3x3 Ripple carry scheduler

The scheduler circuit shown in figure consists of 9 cells, which allocate a particular cross point connection for establishment of data communication between the ports. The cell represented as (n, m) denotes that n<sup>th</sup> input port is allocated to the m<sup>th</sup> output port. The cells which are glowing with black and orange colour in figure 4, that all are demanding the connection establishment. It has been clearly shown in the diagram that only orange highlighted cells are approved to be connected and allocated. It is worth to mention that the cells (1,1) and (2,1) both needs to be connected to output port 1, but only (1,1) cell gets the approval for the same. If a particular cell has been assigned for the connection establishment, then the cells belonging to same column and the same row cannot get the approval to be connected to the demanded output port. Hence proper data communication can take place between input and output ports belonging to the same cell.

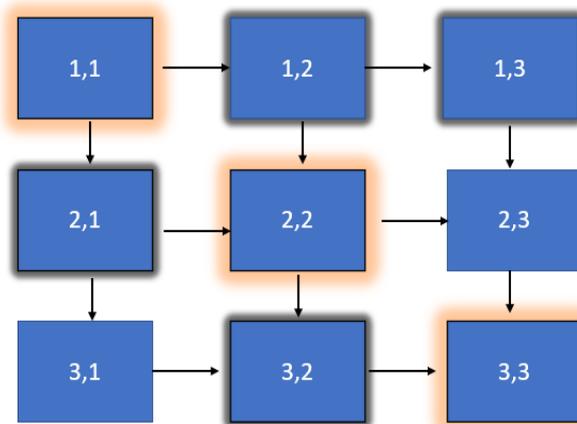


Figure 4. Block diagram of 3x3 crossbar arbiter

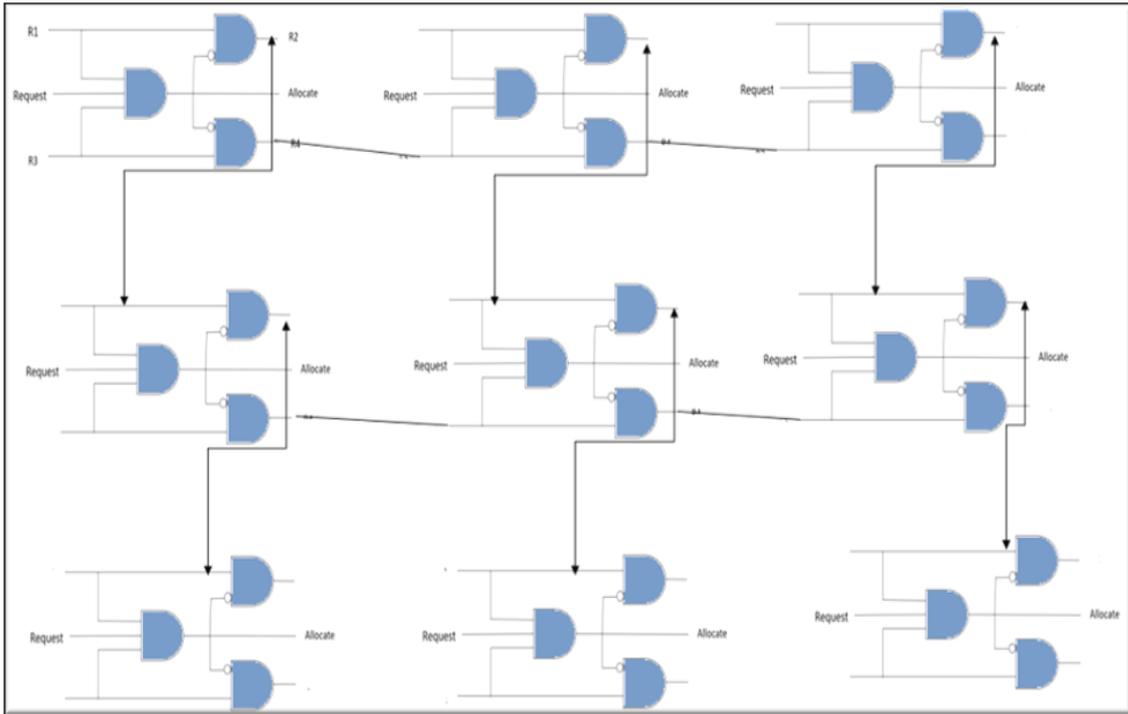


Figure 5. 3x3 Scheduler circuit

Figure 5 implements the 3x3 scheduler circuit in which all nine cells have been accommodated to support nine different input output combinations. The R2s of first row and second row are connected to the R1s of the bottom cells. Similarly, the R4s of the first and second columns are connected to their respective front R3s. All R1s of the first column and all R1s of the first row are at logic1. All the R4s of the third column and R2s of the last row are also floating. Hence for the 3x3 scheduler a total of ten connections are built internally whereas, others are waiting for the input or at the floating till the size of the scheduler gets increased.

### RTL schematic and simulation waveform of the 3x3 scheduler



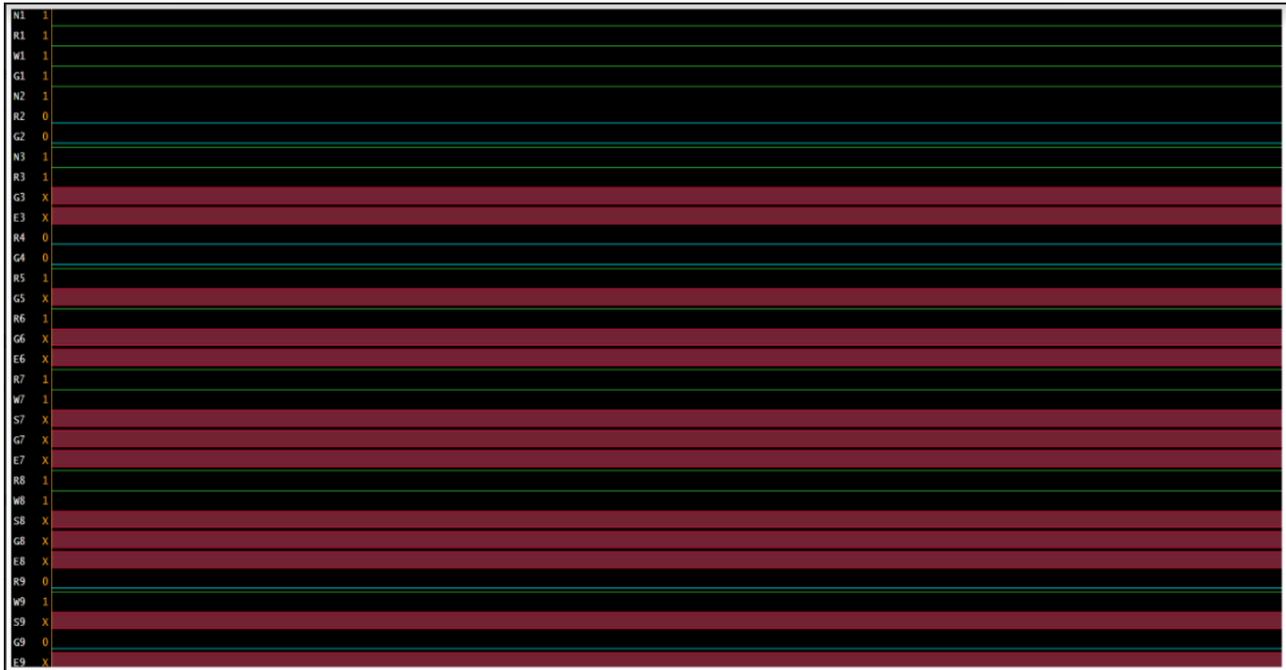


Figure 7. RTL schematic of 3x3 scheduler circuit

### QCA implementation of RCS

Figure 8 represents the quantum dot cellular automata realization of single scheduler cell. The QCA layout uses 35 cells. Whenever, the G goes acquires a logic 1, R3 and R4 goes to zero and hence blocking the front and lower cells to support the non-blocking architecture as explained previously. The simulation results have been verified as shown in figure 9. The layout area and number of cells have been presented in table 7.

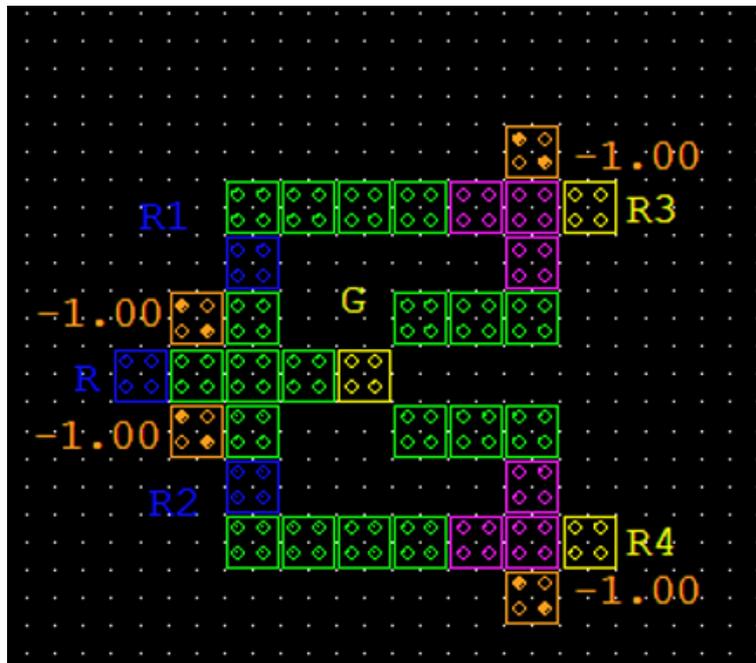


Figure 8. The QCA realization of single scheduler cell

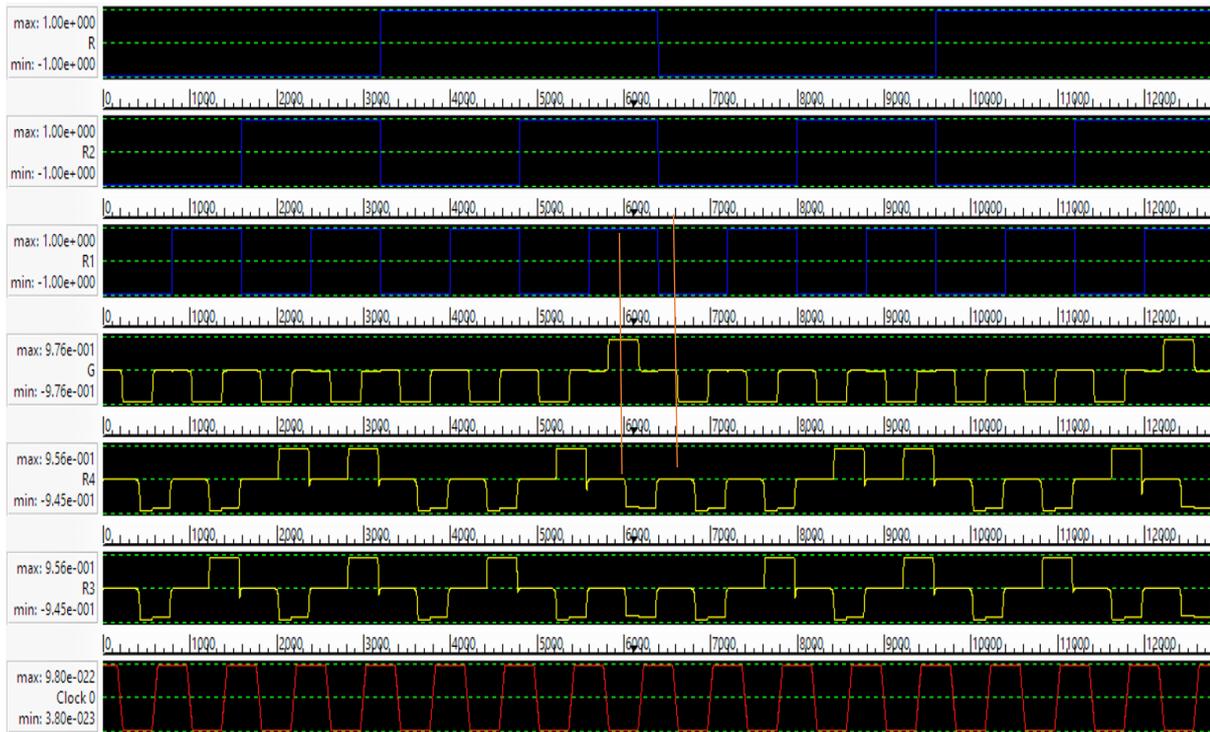


Figure 9. The simulation waveform of basic cell of the scheduler

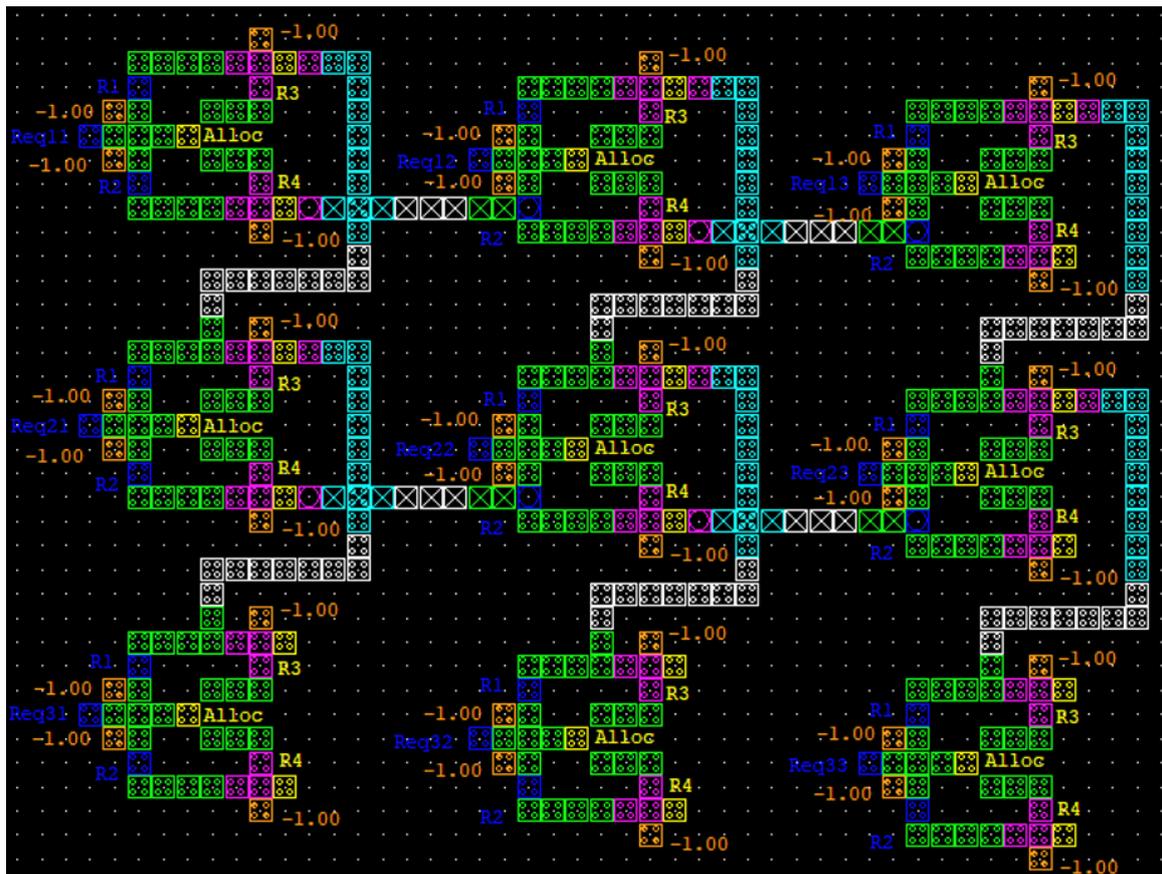


Figure 10. QCA implementation of 3x3 scheduler circuit

The 3x3 QCA realization has been presented in figure 10, in which four wire crossings and 27 majority gates have been incorporated. The realization parameters are given in table 6. The total layout area is 674726.65 nanometre squares.

### Power calculation and performance evaluation of QCA simulation

The power and energy calculation for the QCA layout of cell scheduler has been performed using QCA-Designer Pro E software. The power can be calculated by using the formula,

$$Ediss \leq \left[ \frac{2Y_{new}}{Ek} \left( \frac{Po}{Pold} Y_{old} - \frac{Pn}{Pnew} Y_{new} \right) + \frac{Ek P_{new}}{2} (Po - Pn) \right]$$

Where,  $E_k = 1506.75 \times 10^{-29}$  Joule

$P_n = P_o = 1, P_{old} = -1, P_{new} = 1$

By putting these values, we have evaluated the total energy dissipated for single cell scheduler architecture as  $Ediss = 2.22 \times 10^{-2}$  electron volts.

### The Cost Function Analysis of the QCA simulation

In addition to the area and number of cells used in QCA simulation the delay and energy consumption are also very important parameters to evaluate the total performance of the designed nano circuit.

The energy delay cost function can be calculated using the formula,

$$\text{Cost energy delay} = E^m T^n$$

Where E and T can be defined as the energy dissipation and the delay of the designed circuit (m and n are the weighting factors value can be taken as 2 at standard metric).

One more evaluation has been designed for calculation of cost function is using M, C, T and I as follows,

$$\text{COST OF QCA} = (C^x + I^1 + M^y) T^z$$

where, x, y, z are the weighting parameters which can be assigned for speed, fabrication cost on priority basis. Table 2 highlights the number of wire crossings, number of inverters and number of majority gates in both the single cell and 3x3 QCA implementations.

Table 2 Values C, I, M and T for single cell and 3x3 scheduler to calculate the QCA cost

Parameters	Single cell scheduler circuit	3x3 scheduler circuit
C (wire crossings)	0	4
I (inverters)	2	18
M (majority gates)	3	27
T(delay)	0.25	0.25
Cost Function	0.6875	46.93

### The Verilog Implementation Results

The scheduler has been implemented by defining in Verilog Hardware Description Language (HDL). The simulation has been performed using Xilinx 13.2 and results have been obtained and shown in tables 3, 4 and 5.

Table 3 The design summary of the implemented cell circuit

Cell: in->out	Fanout	Gate Delay (ns)
IBUF: I ->O	2	1.218
LUT3: IO -> O	1	0.704
OBUF: I ->O		3.272
Total	6.236 ns	

Table 4 The delay summary of the implemented cell circuit

Source Pad	Destination Pad	Delay (ns)
n1	g1	6.119
n1	s1	4.827
r1	g1	6.169
w1	e1	4.794
w1	g1	6.371

Table 5 The power values for defined signal rate of the implemented cell circuit

Name	Power (watt)	Signal Rate	Fanout	Slice Fanout
e1_OBUF	0.00216	10000	2	2
g1_OBUF	0.00432	20000	1	1
r1_IBUF	0.00454	50000	1	1
s1_OBUF	0.00363	20000	2	2
Total Data Power (W) = 0.01464				

### Simulation using QCA

The QCA realization have been performed using coherence vector simulation engine with convergence tolerance of 0.001. The layer separation as 11.5 nanometres defining the single nano-cell area being 18x18 nanometre squares as in table 6. The 3x3 scheduler area as calculated 987.84x719.45 nm-squares with 487 cells. The area for single basic cell as calculated 246.57x179 nm-squares as tabulated in table 7.

Table 6 Simulation Parameters for QCA Simulation

Parameters	Value of the Parameter
No. of Samples	12800
Clock Low	$3.80000 \times 10^{-23}$
Radius of Effect (nm)	65.0000 nm
Relative Permittivity	12.900000
Clock High	$9.80000 \times 10^{-22}$
Diameter of electron Dot	5 nm
Maximum Iterations per Second	100
Clock Amplitude factor	2
Convergence Tolerance	0.001000
Layer Separation	11.5
One cell area	$18 \times 18 \text{ nm}^2$

Table 7 Design Analysis using QCA for basic cell of the scheduler and 3x3 scheduler

Name of the Circuit	Number of Cells	Area (nm <sup>2</sup> )
Single cell module of the scheduler	35	44136.84
3x3 Scheduler circuit	487	674726.65

Table 8. Differentiation between the two technologies

Proposed On-chip Scheduler		
Assessment Criteria	CMOS Technology	Quantum Dot Cellular Automata Technology
Power (milli-W)	81	88.8 x 10 <sup>-6</sup>

The findings for power analysis for the two technologies prove QCA to be better technology as depicted in table 8.

## 2 DW clock circuit

### Cell Scheduler

The realistic design of QCA for single cell scheduler uses only 23 cells as shown in figure 11. The number of cells has been reduced and the design presents more convenient clocking for the scheduler design. This is tremendous achievement of the presented work, where the simulation results are also verified as in figure 12.

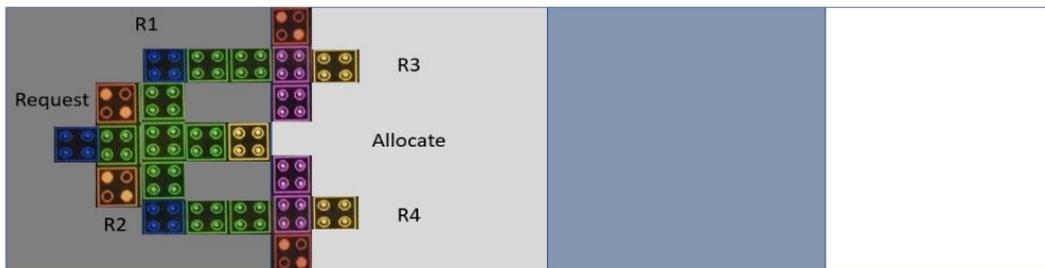


Figure 11 The 2-DW clocking scheme of cell scheduler



Figure 12 The simulation result of cell scheduler under 2-DW clocking scheme

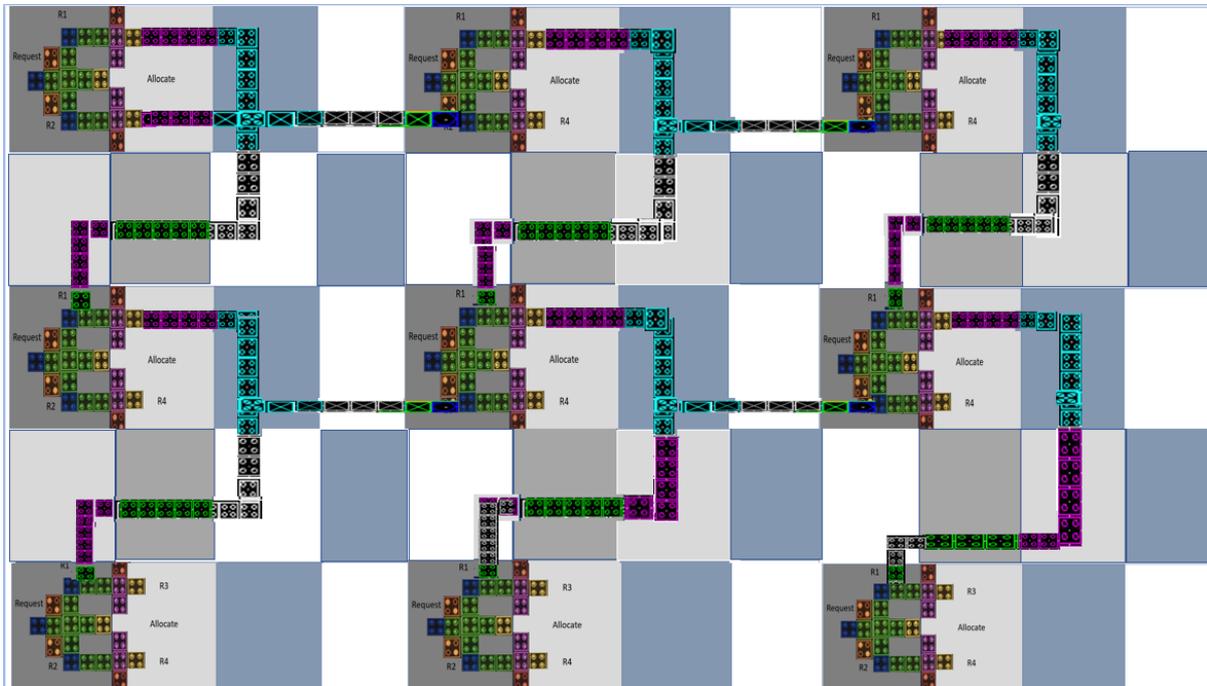


Figure 13 The 3x3 Nano Scheduler with 2 - DW clocking scheme

The 2 – DW clocking realization of 3x3 scheduler circuit has been presented in figure 13. All the four clocking zones have been utilized and arranged to achieve the physical realization of the chip scheduler design.

## Conclusion

The execution of data handling and communication of any switch architecture basically depends on the non-blocking capability. The arbiter is the main component of digital cross connect system which routes the data packets by selecting a path between input and output. In this paper we have implemented a scheduler using both Verilog (RTL Schematic tool: Quartus Prime, waveform: EDA playground) and QCA designer tool. The simulation results are verified using both tools. The power assessment results clearly indicates that QCA circuit design requires less power in comparison to the CMOS realization. The results have been verified for 2-DW clocking scheme using QCA which is a more realistic proposal for operational temperatures and Kink free operation.

## Future Scope

The work can be extended to design scheduler which can connect more inputs and outputs. Also, different architecture for enabling and disabling of the adjacent connections can be derived by the same cell architecture. The work can be extended to design the switch scheduler at nano level. Since in addition to the normal cell clocking scheme two-dimensional clocking switch design have been presented and verified in the paper, the same idea can be applied to design complex schedulers with practical clocking perspective.

## References

- Thakur, G., Sarvagya, M. and Sharan, P., Design and implementation of crossbar scheduler for system-on-chip network in quantum dot cellular automata technology, *Internet Technology Letters*, Wiley, 2018.
- Ugurdag, H. F., O. Baskirt , Fast parallel prefix logic circuits for n<sup>2</sup>n round robin arbitration, *Microelectronics Journal*, vol.8, 2012.
- Park, Y. K. and Lee, Y., Parallel iterative matching-based cell scheduling algorithm for high-performance ATM switches, *IEEE Transactions on Consumer Electronics*, Volume 47, 134-137, issue 1, Feb 2001.
- McKeown, N., Anamtharam, V. and Warland, J., "Achieving 100% throughput in an input-queued switch," *Proc. INFOCOM'96*, San Francisco, March 1996, pp. 296-302.
- McKeown, N., "The iSLIP scheduling algorithm for input-queued switches," *IEEE Transactions on Networking*, vol. 7, no. 2, pp. 188-201, April 1999.
- Helal, K. A., Attia, S. and Mostafa H., Priority-select arbiter: An efficient round-robin arbiter, *New Circuits and Systems Conference (NEWCAS)*, IEEE 13th International, 2015.
- Das, J. C., D. De, Computational fidelity in reversible quantum-dot cellular automata channel routing under thermal randomness, *Nano communication networks* 18 (2018) 17–26.
- Silva, D. S., L.H. Sardinha, M. A. Vieira, L. F. Vieira and Neto, O. P. V. ,Robust serial nano communication with qca, *IEEE Transactions on Nanotechnology* 14 (3) (2015) 464–472.
- Sardinha, L. H., Costa, A. M., Neto, O. P. V., Vieira, L. F. and Vieira, M. A. , Nano router: A quantum-dot cellular automata design, *IEEE Journal on Selected Areas in Communications* 31 (12) (2013) 825–834.
- Das, S., D. De, Nano communication using qca: a data path selector cum router for efficient channel utilization, in: *Radar, Communication and Computing (ICRCC)*, 2012 International 430 Conference on, IEEE, 2012, pp. 43–47