A Novel Design of a random Generator Circuit in QCA

AndishehKeikha
Department of Electrical Engineering,
K.N. Toosi University of Technology
Tehran, Iran

ChitraDadkhah
Department of Electrical Engineering,
K.N. Toosi University of Technology
Tehran, Iran

Mohammad Tehrani
Nanotechnology and Quantum Computing Lab.
ShahidBeheshti University,
Tehran, Iran

KeivanNavi
Nanotechnology and Quantum Computing Lab.
ShahidBeheshti University,
Tehran, Iran

ABSTRACT
CMOS miniaturization limits have improved research on other advanced alternative technologies. Quantum Cellular Automata (QCA) is a nanometer scale technology that is one of these alternatives. Two principals have been introducedin the concept of Random Number Generator which are related to some physical occurrences or computational algorithms. In this paper we introduce a QCA circuit which is the first true random generator.

Keywords
Random Generator, Quantum Cellular Automata, Novel Design, Nanoelectronic Circuit.

1. INTRODUCTION
In recent years devices based on CMOS technology have reached their limits [1] and quantum-mechanical principles have made a huge vicissitude in speed, power consumption,circuit size and logic gates. Among the proposed technologies, QCA(Quantum-Dot Cellular Automata) is very powerful as it offers a new methodology of computation.It was set up by Lent in a configurationbuilt up of cells including four quantum dots at each corner of a square array coupled by tunnel barriers and occupied by two electrons. Due to Coulomb repulsion the two electrons always occupy opposite dotsdiagonally. The logic "0" and "1" are assigned to these two ground state polarizations. Electrons can tunnel between the dots, however cannot leave the cell. [2]
Many gates such as majority gate, inverter and wire have been produced based on QCA; however there are still some other operators that have not been considered in this domain. In this paper we propose an implementation of a QCA-based random generator.
Two principal methods are introduced for random number generation. One quantifies some physical occurrences that arelikely to exist. The other produces long series of random results with the aim of computational algorithms and are determined by an initial value called seed or key. The latter type is often known as pseudorandom number generators. Either of the methods cannot often achieve the goal of being truly random. What we have aimed in this paper is to attempt to reach this aim.
Brief introductions to QCA are in section 2. Our design and its experimental results are proposed in section 3 and section 4 presents the material and methods that we used in this approach and section 5 concludes our paper.

2. WHAT IS QCA
As it was previously mentioned, a QCA cell consists of four quantum dots occupied by two electrons. The electrons acts due to Coulomb repulsion formula and occupy the opposite corners of the cell. These arrangements results in two different polarizations; -1 and +1 which are aligned to the logics 0 and 1 respectively. The polarization of a QCA cell reverses when the two electrons tunnelbetween neighboring dots in the cell. When the inter-dot barrier is raised, a QCA cell will keep its current polarization and do not change due to the changes in neighbors' polarizations. [2-4]
As it is shown in Figure 1, it is possible to produce a wire using the attributes of a quantum cell due to the Coulomb repulsion. It is called a wire as the input polarizations are maintained through the output. The order of cells in an inverter and a majority gate are also shown in Figure 1,the polarizations of input and output cells in an inverter are the opposite, and in a majority gate we can assume that the polarization of the output is the majority polarization of the inputs. If one of the inputs of the majority gate is determined to be fixed, it is possible to reach "AND" and "OR" gates depending on the fixed input. If the fixed input is 0 then the majority gate will act as the "AND" gate for the other two inputs. Also putting the fixed input to 1 would lead to the "OR" gate. [5]

3. QCA BASED RANDOM GENERATOR
Our initial design of a QCA based random generator is shown in Figure 2 (a), where cells A, B, and C and D are input cells and cell O is the output. It is designed by QCADesigner, where all the cells are in the same clock phase and in just one layer. The simulation results are shown in Table 1 and as it can be seen, the results are not set for two of the cases of inputs that are 0011, 1100 for ABCD respectively. It is because in those cases, the result is different in each simulation to the next one. Also there are four cases that the output is null, which means that the polarization of the output in those cases was 0.
The simulation was run 50 times and the output value for the cases 1100 and 0011 were recorded. Table 2 shows

the results of 50 simulations for each set of these inputs. As it can be seen, when the input is set to 1100 for ABCD the result is a fair random number. It means that the probability of the output being 0 is equal to the probability of being 1. The result of one of the simulations in case showed in Figure 2 (a) is shown in Figure 3. The results are just different in the points that ABCD are 0011 or 1100 respectively and are the same in other points.

At this point we want to see the effect of an extra diagonally input I close to cell X and comparing the results while extending the distance between the new input cell I and cell X. When cell I is connected to the cell X (Figure2 (b)), it results in a fix output at the points where the inputs are set to 00110, 00111, 11000, 11001 for ABCDI respectively. Running simulation in these cases, the results are fix polarizations at these points in all the simulations. In Other points the results were the same as the previous design with no input cell I.As we set input I further apart by distances of 1, 2 or 3 cells (Figure 2 (c), (d) and (e)) from cell X, the noise disappears and we reach our aim of generating different output results for those special inputs in multiple simulations.

The simulation was run 50 times by assuming the distance between cells I and X are equal to 1, 2 and 3. In each simulation, the value of output was recorded for the four inputs that were mentioned above. Table 3, 4 and 5 shows the result of 50 simulations for these four inputs. As it can be seen, this design results in an unfair random generator that in design of Figure 2 (c) the results are more likely to be 0 when I is 0 and also more likely to be 1 when I is 1. In the designs with more than 1 distance between I and X (Figure 2 (d) and (e)) results gravitate towards 0 which means that there is a higher probability of 0 occurring than the probability of 1 occurring in the output cell. It seems that its inequality reduces as the distance between Input I and the whole design increases.

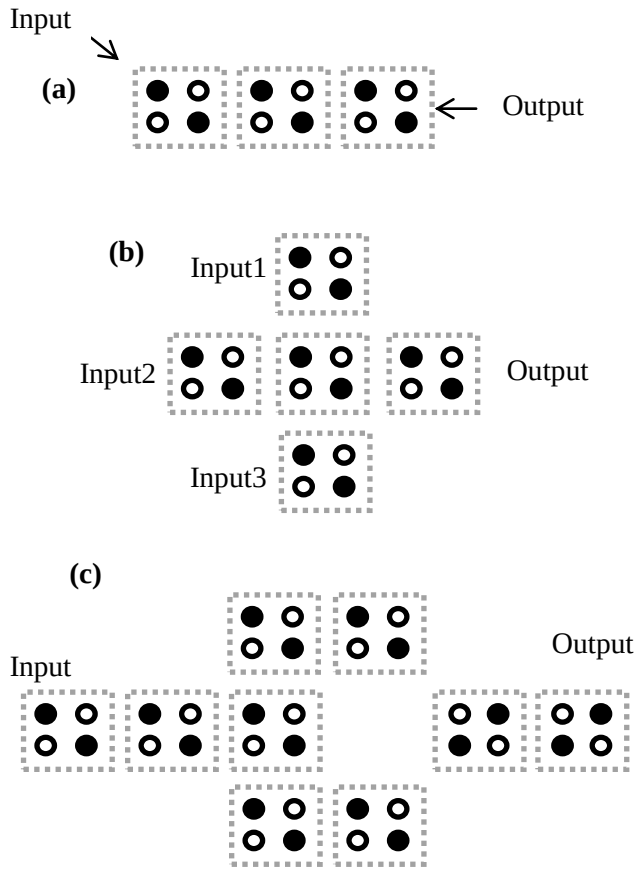

Figure1: a: Wire. b: Majority gate. c: Inverter

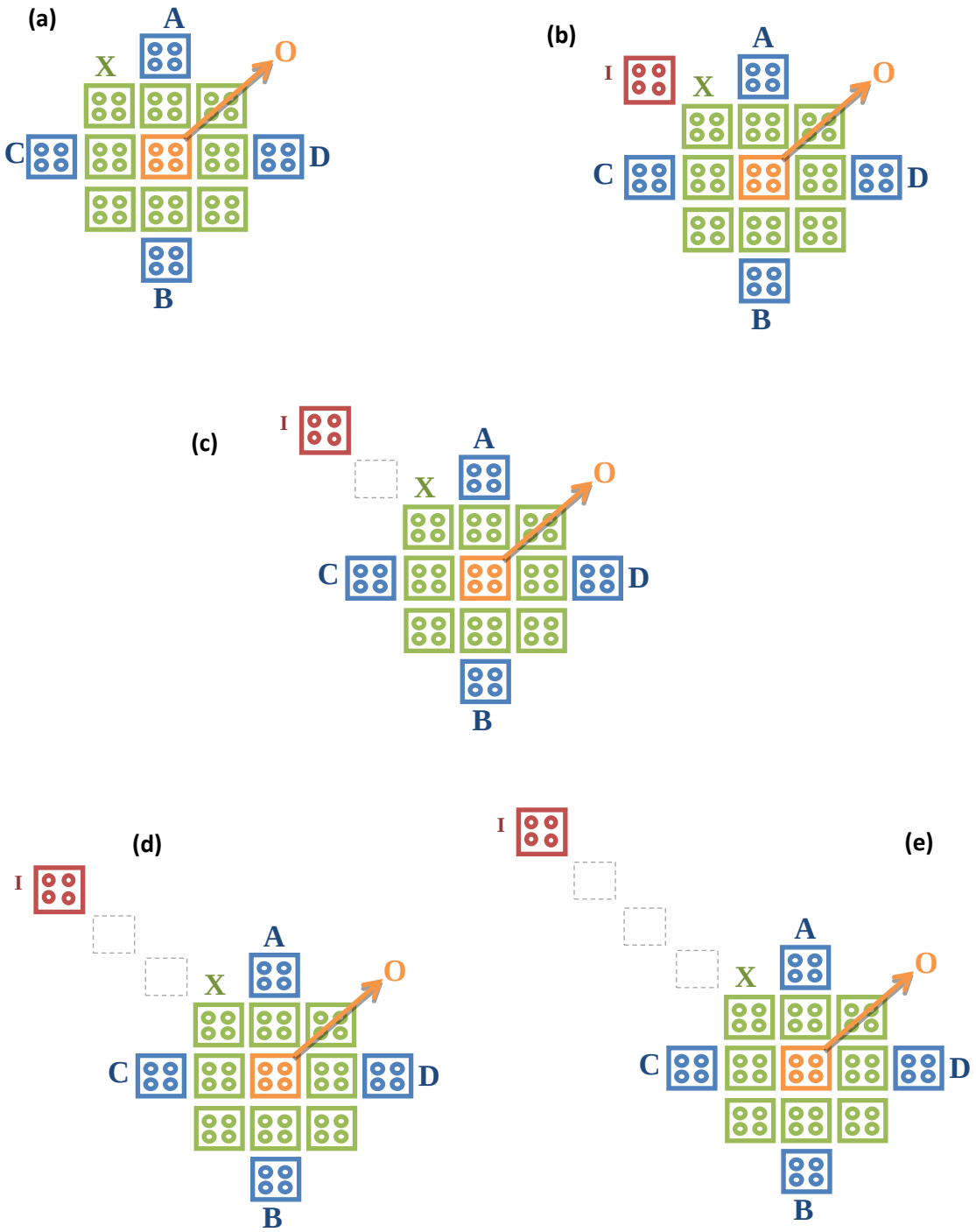

Figure 2:a: The initial design. b: Put an extra output I with distance equal to zero to the whole design. c: Put I in the distance = 1. d: Put I in the distance = 2. e: Put I in distance = 3.

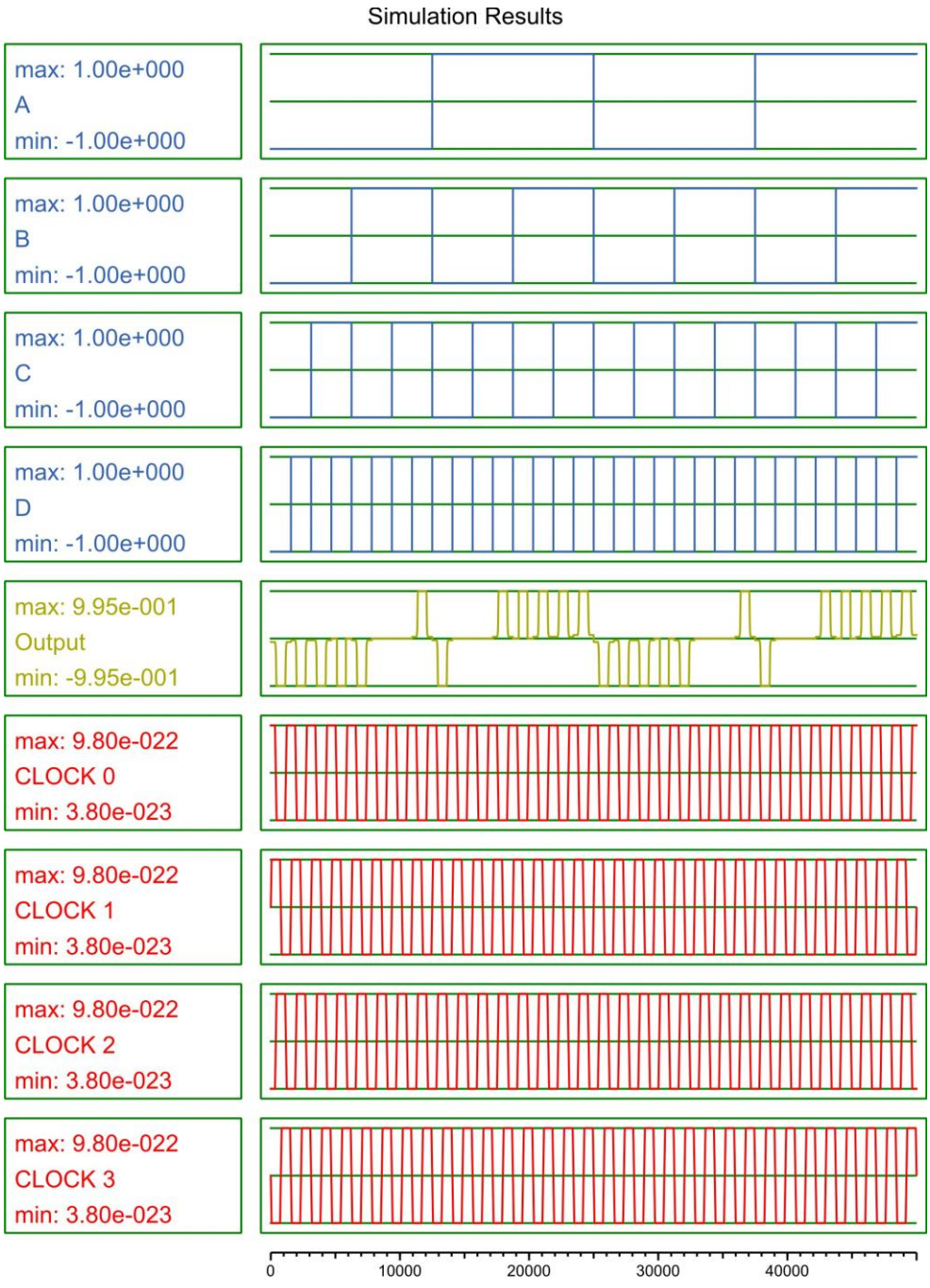

Figure 3: The result of simulation for the initial design showed in figure 2(a)

Table 1. Output results for the initial design

A	B	C	D	O
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	
0	1	0	0	0
0	1	0	1	Null
0	1	1	0	Null
0	1	1	1	1
1	0	0	0	0
1	0	0	1	Null
1	0	1	0	Null
1	0	1	1	1
1	1	0	0	
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Table 2. The probability of occurrence zero or 1 in output for theinitial design

ABCD	O = 0	O = 1
0011	34%	66%
1100	50%	50%

Table 3. The probability of occurrence zero or 1 in output when the distance between I and the whole design=1

ABCDI	O = 0	O = 1
00110	30%	70%
00111	66%	34%
11000	33%	67%
11001	54%	46%

Table 4. The probability of occurrence zero or 1 in output when the distance between I and the whole design=2

ABCDI	O = 0	O = 1
00110	60%	40%
00111	62%	38%
11000	54%	46%
11001	54%	46%

Table 5. The probability of occurance zero or 1 in output when the distance between I and the whole design=3

ABCDI	O = 0	O = 1
00110	52%	48%
00111	62%	38%
11000	54%	46%
11001	56%	44%

4. MATERIALS & METHODS

As it was mentioned in section 3, we used QCADesigner tool which was initially developed at the ATIPS Laboratory, University of Calgary for designing our approach. This product creates a fast and accurate simulation for QCA. It is written in C/C++ and utilizes a comprehensive area of open-source software such as the GTK graphics library and is claimed to be under the GNU public license for open source software.

The current version of QCADesigner includes three different simulation engines.The first considers cells to have either zero or full polarization (-1 and 1) which is a digital logic simulator. The second is a nonlinear approximation engine, which uses the nonlinear cell-to-cell interactions to drive the stable state of the cells within a design. The third uses a two-state Hamiltonian to find an approximation of the full quantum mechanical model of such a system.[6]

In our design we used QCADesigner Ver. 2.0.3 In the bistable approximation, the following parameters were used:

- Number of samples = 50000,
- Convergence tolerance = 0.001,
- Radius of effect = 200nm,
- Relative permittivity = 12.9 (for GaAs),
- Clock high = 9.8e-022,
- Clock low = 3.8e-023,
- Clock shift = 0,
- Clock amplitude factor = 2,
- Maximum iteration per sample = 100,
- Simulation type = Exhaustive.

5. EXPERIMENTAL RESULTS

We made a comparison table of power, bit rate and area in our approach and in approaches proposed in [7-19] respectively.(Table 6)

Lent has mentioned in [20] that the power consumption in QCA arrays is 10^{-10}W per input bit, so in the designs with 4 and 5 input bits, the power consumption will be 4×10^{-4} µW and 5×10^{-4} µW respectively.

"The QCA pipeline clocks are assigned to the cells so that the noise in crossovers can be tolerated and the signal flowing in the gates can be synchronized." Kim [21] set the clock rate at 1THz, to achieve the bit rate of 1Tb/s.

Table 6. Comparison of power, bit rate and area

Approaches	Bit Rate	Power	Area
Our approach with 4 inputs	1Tb/s	4×10^{-4} µW	.01 μm^2
Our approach with 5 inputs	1Tb/s	5×10^{-4} µW	.02-.03 μm^2
[7]	1.4Mb/s	3.9mW	1.5 mm^2
[12]	10Mb/s	2.3mW	.0016 mm^2
[13]	200kb/s	50µW	.009 mm^2
[14]	40Mb/s	29mW	.518 mm^2
[11]	200kb/s	1mW	.036 mm^2
[8](DC)	500b/s	2.92µW	.031 mm^2
[8](DC)	5kb/s	9.39 µW	.031 mm^2
[8](FIR)	50kb/s	180 µW	1.49 mm^2

[9](Without pad, eight stages pipeline)	40Mb/s	29mW	.234 mm^2
[10]	20Mb/s	1mW	
[15]	40kb/s	1.04 µW	.05 mm^2
[16]	11kb/s	2mW	.0012 mm2
[17]	125 Mbps	0.095 W	0.052 mm2
[18]	40kb/s	1 µW	0.0512 mm2
[19]	2.4Gb/s	7mW	about 0.006 mm2

As it can be seen, the results in our approach are drastically better than conventional approaches based on CMOS structures. To show the comparison of the results more precise, we compared the bit rates in figure 4. As the differences between bit rates were major, we scaled the horizontal axis in logarithm of base 10. Although the bit rate values are scaled there is a big difference in bit rate value in our approach compare to others. Also there is another comparison in figure 5 on the power (again in logarithmic scale) and it can be seen that there in our approach the power consumption is far less than the power consumptions in other proposed works. The last comparison is showed on the area consumption in figure 6 which shows how the circuit results of our approach requires much less area than the others.

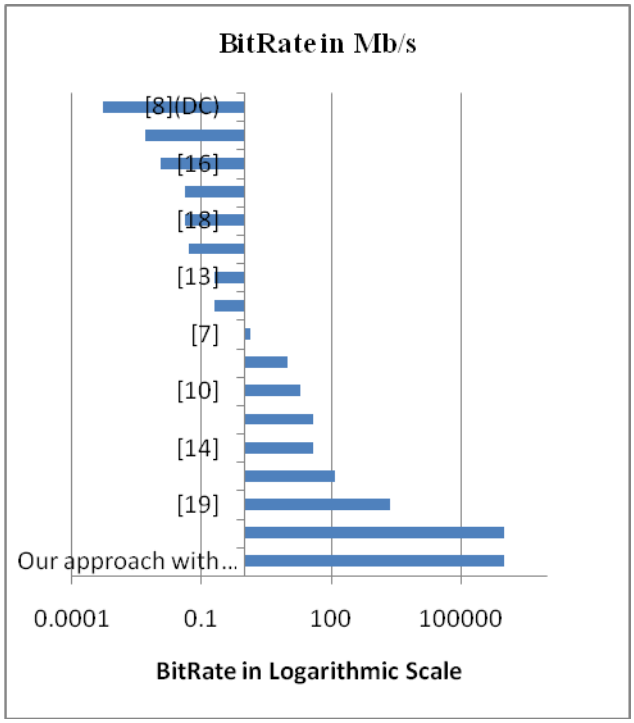

Figure 4: Bit rate comparison

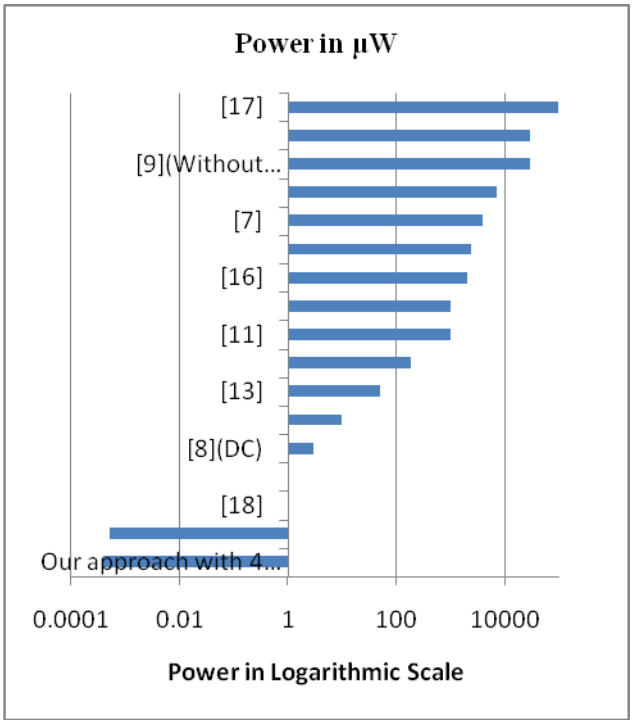

Figure 5: Power Comparison

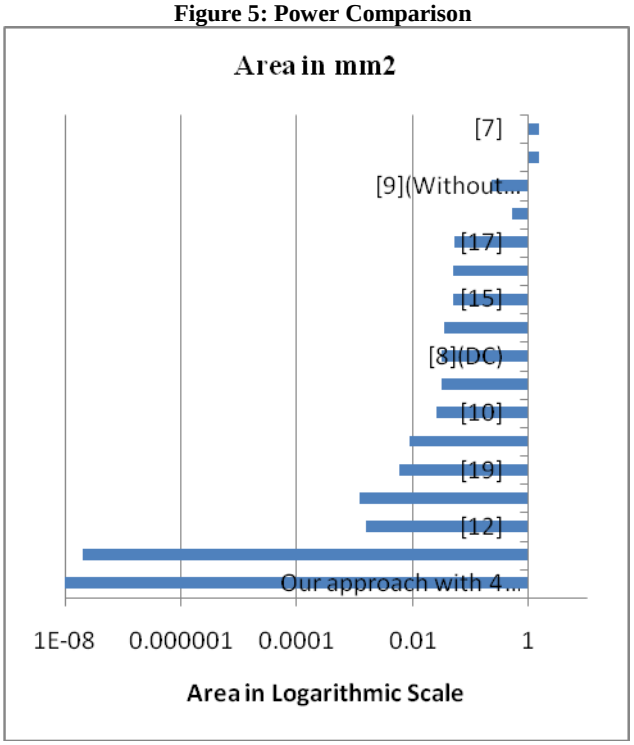

Figure 6: Area Comparison

6. CONCLUSION

In this paper we proposed a design of QCA that acts as a random generator. Two models were proposed, which showed that it can act as two different kind of random numbers generator: fair and unfair. By fair we mean that the probability of producing 0 and 1 in output is the same.

However in unfair generation type that is introduced, generating 0 in output is more likely than 1. By using this approach, for the first time, we succeeded in achieving a true random number generator that can be designed directly without using different hardware or algorithms' behavior.

7. REFERENCES

[1] A. Abdollahi, M. Pedram. Analysis and Synthesis of Quantum Circuits by Using Quantum Decision Diagrams. Proceedings of the conference on Design, automation and test in Europe (2006). pp. 1-6.

[2] M. R. Azghadi, O. Kavehei, and K. Navi. A Novel Design for Quantum-dot Cellular Automata Cells and Full Adders. Journal of Applied Sciences. 7, no. 22 (2007), pp. 3460-3468.

[3] M. A. Tehrani, and K. Navi. A Novel Quantum Dot Cellular Automata for Implementation of Multi-Valued Logic. Elsevier, Nanotoday Conference (2009).

[4] M. A. Tehrani, F. Safaei, M. H. Moaiyer, and K. Navi. Design and Implementation of Multi-Stage Interconnection Networks Using Quantum-Dot Cellular Automata. Microelectronics Journal (2011).

[5] R. Tang, F. Zhang, and Y. B. Kim. Quantum-Dot Cellular Automata SPICE Macro Model. Proceedings of the 15th ACM Great Lakes symposium on VLSI - GLSVSLI (2005). pp. 108-111.

[6] K. Walus, T.J. Dysart, G. a Jullien, and R. a Budiman. QCADesigner: A Rapid Design and Simulation Tool for Quantum-Dot Cellular Automata. IEEE Transactions On Nanotechnology. 3 (March 2004), pp. 26-31.

[7] C.S. Petrie, and J. Connelly. A noise-based IC random number generator for applications in cryptography. IEEE Transactions on Circuits and Systems. 47 (May 2000), pp. 615-621.

[8] J. Holleman, S. Member, S. Bridges, B.P. Otis, and C. Diorio. A 3 W CMOS True Random Number Generator With Adaptive Floating-Gate Offset Cancellation. IEEE Journal of Solid-State Circuits. 43, no. 5 (May 2008), pp. 1324 - 1336.

[9] F. Pareschi, G. Setti, and R. Rovatti. Implementation and Testing of High-Speed CMOS True Random Number Generators Based on Chaotic Systems. IEEE Transactions on Circuits and Systems. 57, no. 12 (2010), pp. 3124 - 3137.

[10] F. Cao, and S. Li. Random numbers from an integrated CMOS double-scroll. IEICE Electronics Express. 7 (2010), pp. 1382-1387.

[11] C. Tokunaga, D. Blaauw, and T. Mudge. True Random Number Generator With a Metastability-Based Quality Control. IEEE Journal of Solid-State Circuits. 43 (January 2008), pp. 78-85.

[12] M. Bucci, L. Germani, R. Luzzi, A. Trifiletti, and M. Varanonuovo. A High-Speed Oscillator-Based Truly Random Number Source for Cryptographic Applications on a Smart Card IC. IEEE Transactions on Computers. 52 (2003), pp. 403-409.

[13] R. Brederlow, R. Prakash, C. Paulus, and R. Thewes. A low-power true random number generator using random telegraph noise of single oxide-traps. IEEE International Solid-State Circuits Conference(ISSCC). (2006), pp. 1666 - 1675.

[14] F. Pareschi, G. Setti, and R. Rovatti. A Fast Chaos-based True Random Number Generator for Cryptographic Applications. Proceedings of the 32nd European Solid-State Circuits Conference. (2006), pp. 130-133.

[15] W. Chen, W. Che, Z. Bi, J. Wang, N. Yan, X. Tan, J. Wang, H. Min, and J. Tan. A 1.04 μW Truly Random Number Generator for Gen2 RFID tag. IEEE Asian Solid-State Circuits Conference. (November 2009), pp. 117-120.

[16] N. Liu, N. Pinckney, S. Hanson, D. Sylvester, D. Blaauw. A True Random Number Generator using Time-Dependent Dielectric Breakdown. Symposium on VLSI Circuits Digest of Technical Papers (2011). p. 216-217.

[17] Ü. Güler, and S. Ergün. A high speed, fully digital IC random number generator. AEU - International Journal of Electronics and Communications. (2011).

[18] W. Chen, W. Che, N. Yan, X. Tan, and H. Min. Ultra-Low Power Truly Random Number Generator for RFID Tag. Wireless personal communication. 59, 1 (2011), pp. 85-94

[19] S. Srinivasan, S. Mathew, R. Ramanarayanan, F. Sheikh, M. Anders, H. Kaul, V. Erraguntla, R. Krishnamurthy, G. Taylor. 2 . 4GHz 7mW All-Digital PVT-Variation Tolerant True Random Number Generator in 45nm CMOS. IEEE Symposium on VLSI Circuits (2010), pp. 203-204.

[20] C.S. Lent, P.D. Tougaw, and W. Porod. Quantum cellular automata. Nanotechnology. 4 (1993), pp. 49-57.

[21] K. Kim, Quantum-Dot Cellular Automata Design Guideline. EICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences. 89, no. 6 (2006), pp. 1607-1614.