Signal Processing and Renewable Energy
March 2020, (pp. 23-36)
ISSN: 2588-7327
eISSN: 2588-7335

Fault-Tolerant Techniques for Quantum-dot Cellular Automata Circuits and Systems

Razieh Farazkish[1*], Mani Zarei[2]

[1] Department of Computer Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran.
[2] Department of Computer Engineering, Shahr-e-Qods Branch, Islamic Azad University, Tehran, Iran.

Received: 03-Jan-2020, Accepted: 20-Jan-2020.

Abstract

This paper explains fault tolerance techniques for Quantum-dot cellular automata which offer remarkable robustness to implement QCA arithmetic circuits. It begins with a study of QCA based design. A classification for fault types is presented and some fault tolerance techniques are examined and their relevance for QCA circuits is evaluated. Finally, it is concluded that a combination of two or more hardware redundancy techniques is needed for tolerating faults in QCA circuits and systems. The proper functionality of the presented design is checked by computer simulations using the QCADesigner tool. Simulation results confirm our claims and their usefulness in designing robust digital circuits.

Keywords: Quantum-dot cellular automata, Nanoscale circuits, Fault tolerance, Hardware redundancy.

1. INTRODUCTION

As CMOS technology approaches its fundamental limits, recent years have found out alternative technologies. Among emerging technologies, Quantum-dot cellular automata (QCA) relies on new design models. QCA is a promising novel technology, which offers a robust design and has displayed extra low power and extra high speed [1-3].

The superior features of QCA over CMOS devices with the feasibility of designing logic gates and circuits point out the potential of QCA as a promising novel computing model [4].

*Corresponding Author"s Email: r.farazkish@srbiau.ac.ir

There are, however, several obstacles to the practical realization of QCA and exploiting the full potential of this new model [5]. Major hurdles that need to overcome in QCA are fault-tolerance [6]. The small size of QCA cells is subject to manufacturing defects and faults.

In this letter, we will focus on an approach to analyze some fault tolerance techniques and evaluate their efficiency to the demands of future nanoelectronic circuits.

Fault tolerance is the ability of a system to provide a service complying with the specification despite faults [7]. The fault-tolerant design of QCA is necessary for the characterization of defective behavior of QCA circuits. In recent years the fault-tolerance properties of QCA circuits have been demonstrated by many researchers [8-26].

This letter studies some aspects of designing and analyzing fault tolerance techniques and evaluates their efficiency to the demands of future 24 anoelectronics circuits.

2. MATERIALS AND METHODS

2.1. A Brief Outline of QCA

Quantum-dot cellular automata is a new device architecture, which is proper for the nanoscale systems. This technology, like other modern technologies such as vehicular ad hoc networks [27-30], has attracted the attention of many researchers. The principle of QCA proposed by Lent and Toagaw [31]. The basic computational element in QCA is a quantum cell. A quantum cell consists of four charge dots positioned at the corner of a square. Also, a quantum cell consists of two extra mobile electrons, which are allowed to tunnel between neighboring sites. The Coulomb repulsion between the electrons will force them to occupy corners in the square. The two possible polarization states represent a logic "0" and "1" (Fig. 1(a)).

As shown in Fig. 1(b), an ordinary QCA majority gate needs only five QCA cells; three inputs labeled A, B and C, the device cell and the output. The logic function of the majority gate is:

$$M(A,B,C) = AB+AC+BC \quad (1)$$

and in Fig. 1(c) a QCA inverter is shown.

2.2. Advantages and Difficulties of QCA-Based Design

QCA offers several distinct advantages over traditional technologies: (1) This schema inherently allows for small feature size and thus high computational density. (2) Because

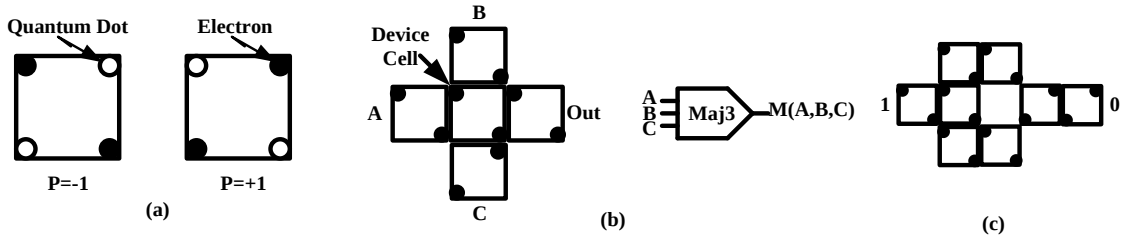

Fig. 1. (a) Basic QCA cell and binary encoding, (b) A QCA majority gate, (c) A QCA inverter.

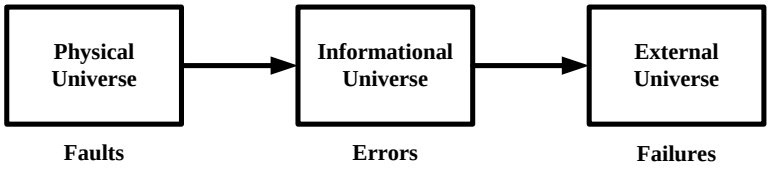

Fig. 2. Three universe model representing the cause and effect relationship between faults, errors, and failures.

current does not flow through QCA-based circuits, these designs can perform at low power levels. This low power cost is vital to being able to achieve the device densities. (3) QCA design support massively parallels computational architectures, which can allow for more efficient information processing.

Many obstacles must be overcome before QCA-based circuits are available as a viable technology: (1) quantum cells must be small, on the order of 18nm, to be efficient. Currently, the technology does not exist to reliably manufacture quantum cells of this size and assemble them into particular structures. Fortunately, much time and effort are being spent on these scale-related issues. (2) As with any technology on this scale, it is difficult to create interfaces between the computational circuits and I/O devices such as monitors and keyboards that would allow the user to interact with the computer. Also, this limitation is faced by other technologies. (3) QCA structures exhibit propagation delays. This delay can be attributed to the finite amount of time that it takes for the electrons in a cell to tunnel to their new position [9]. (4) In addition to the robustness capabilities of any future QCA device, another difficulty in its practical implementation is patterning a circuit. That is, if a simple gate is used within a QCA circuit, a high degree of accuracy is needed for proper alignment of cells. With today's technology, it is very hard to assemble a specific pattern, let alone making it precise. This issue should also be considered in the context of another problem associated with the manufacturing of massive arithmetic circuits. It is believed that QCA architectures could eventually be implemented with self-assembled molecules, although there are no candidates as yet and there are questions to whether molecular assemblies would give enough control over cell positioning [9, 23]. This suggested that while a great QCA array with a very large number of cells can be implemented, the exact position of cells would be hard to control. In other words, practical implementation QCA array represent a high degree of fault in cell positioning.

This has motivated us to investigate the design of QCA devices from a different perspective. In fact, instead of analyzing the behavior of a single cell, we have analyzed the behavior of two-dimensional arrays of cells for designing fault-tolerant QCA devices [11-13].

2.3. Faults and Fault Tolerance

A fault is a physical defect, imperfection, or flaw that occurs within some hardware or software components. There is a cause and effect relationship between faults, errors, and failures. Specifically, faults are the cause of

errors, and errors are the cause of failures. Fig. 2 illustrates the relationships implied in the three-universe model. In summary, faults are physical events that occur in the physical universe. Faults can result in errors in the information universe, and errors can ultimately lead to failures that are witnessed in the external universe of the system [7].

There are three primary techniques for attempting to improve a system's normal performance in an environment where faults concern: fault avoidance, fault-masking, and fault tolerance.

Fault avoidance is any technique that is used to prevent faults in the first place. Fault avoidance can include such things as design reviews, testing, and any other quality control methods.

Fault masking is any process that prevents faults in a system from introducing errors into the informational structure of that system.

Fault tolerance is the ability of a system to continue to perform its tasks after the occurrence of faults. The ultimate goal of fault tolerance is to prevent system failures from ever occurring. Fault tolerance can be achieved through many techniques.

When fault tolerance is required, some form of redundancy is also required. Redundancy is simply the addition of information, resources, or time beyond what is needed for normal system operation [7]. The redundancy can be in four forms: 1. Hardware redundancy, 2. Software redundancy, 3. Information redundancy, and 4. Time redundancy. But, redundancy can have a very important impact on a system's performance, size, weight, power consumption, and reliability. So, we must use proper redundancy techniques for designing fault-tolerant QCA circuits and systems. In this note, we will use hardware redundancy techniques for QCA based design.

2.4. Hardware Redundancy

The physical replication of hardware is the most common form of redundancy used in digital systems today. As QCA cells are small and inexpensive, the concept of hardware redundancy becomes more practical.

There are three basic forms of hardware redundancy; passive, active and hybrid [7]. Passive techniques use the concept of fault masking to hide the occurrence of faults. The active approach achieves fault tolerance by detecting the existence of faults and performing some action to remove the faulty hardware from the system. In other words, active techniques require the system to be reconfigured to tolerate faults. Hybrid techniques combine the attractive features of both passive and active approaches. Fault masking and system reconfiguration are used in hybrid systems.

In the next section, we have attempted to propose some kind of hardware redundancy techniques for QCA circuits; these methods can be used for designing fault-tolerant QCA circuits with situations of single or multiple faults.

3. FAULT-TOLERANT TECHNIQUES FOR QCA BASED DESIGN

Two fundamental units of QCA-based design are majority and inverter gates [4, 26]; hence, this note investigates fault tolerance techniques for QCA majority gate that offers remarkable robustness with respect to single

or multiple faults. Improving the robustness of the majority gate leads to the efficient designing of many fault-tolerant arithmetic circuits and systems.

3.1. Passive Hardware Redundancy for QCA Based Design

Passive hardware redundancy relies on voting mechanisms to mask the occurrence of faults. Most passive approaches are developed around the concept of majority voting.

The most common form of passive hardware redundancy is triple modular redundancy (TMR) [7]. As illustrated in Fig. 3 (a), the basic concept of TMR is to triplicate the hardware and perform a majority vote to determine the output of the system. If one of the modules becomes faulty, the two remaining fault-free modules mask the results of the faulty module when the majority vote is performed.

One of the attractive features of QCA based design is having a majority gate that inherently performs the voting operation. So, in the TMR structure, the majority gate can be used as a voter when it is needed (Fig. 3 (b)).

The primary difficulty with TMR is the voter if the voter fails the complete system failures. In other words, the voter is a single point of failure (SPF) in the TMR structure.

One approach to overcome susceptibility to voter failure is to triplicate voters and provide three independent outputs, as shown in Fig. 4.

Another approach to overcome SPF to voter failure is using fault-tolerant three-input majority gate as shown in Fig. 5 [10].

Also, we can use multiple-stage TMR to correct errors before being passed to a subsequent module (Fig. 6).

A more generalized passive hardware redundancy realization is n-modular redundancy (NMR), which means that there are n redundant modules and a voter. The concept of NMR is shown in Fig. 7. The advantage of using n module is this fact that the more module faults can be tolerated. This structure is capable of detecting $\lfloor (n-1)/2 \rfloor$ errors.

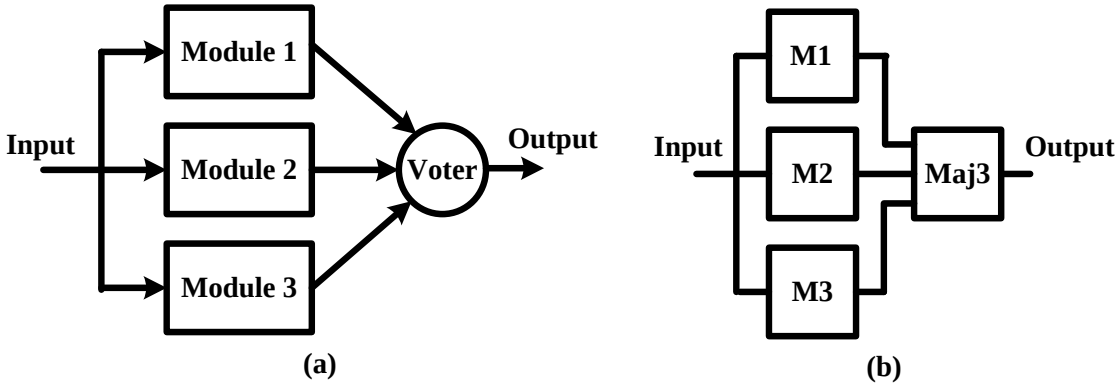

Fig. 3. (a) TMR structure, (b) Using Majority gate as a voter in QCA structure.

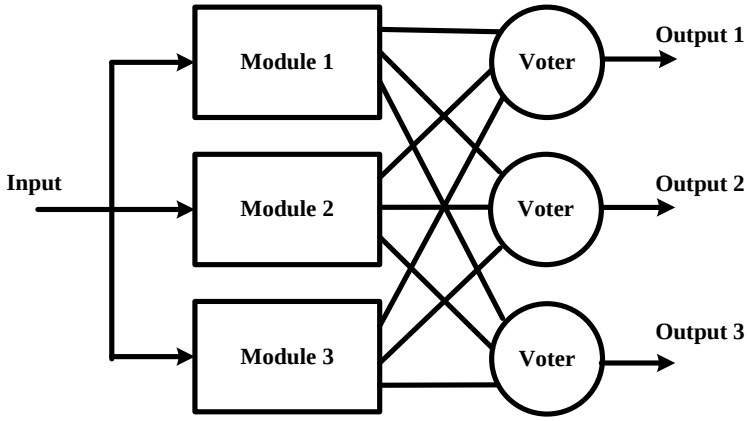

Fig. 4. TMR with triplicate voters.

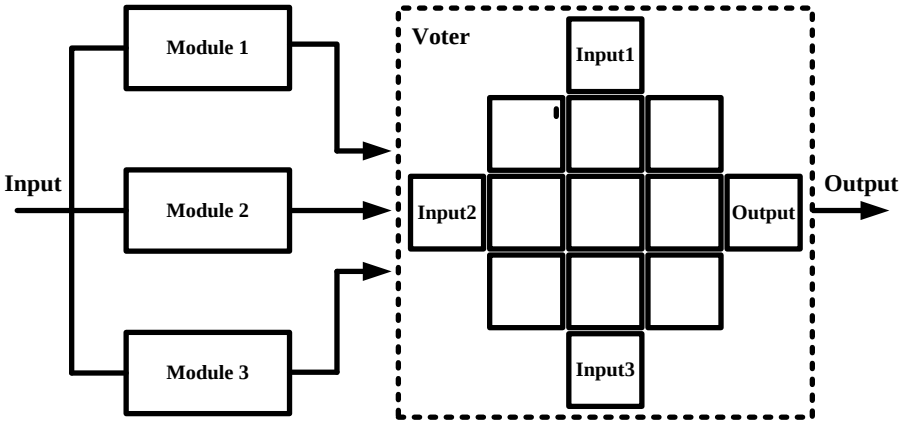

Fig. 5. Using fault tolerant three-input majority gate as a voter for TMR structure.

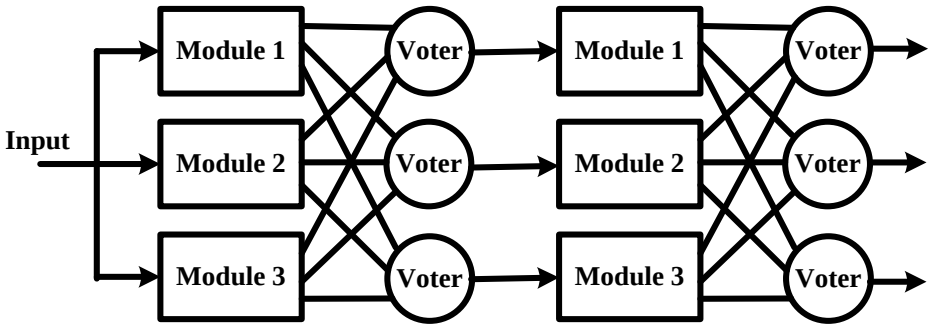

Fig. 6. Multiple-stage TMR.

As mentioned before, the primary element of NMR systems is the voter; in QCA structures, the majority gate can be used as a voter. For example, for NMR with 5 modules (5MR), a five-input majority gate can be used as a voter [9, 11-13].

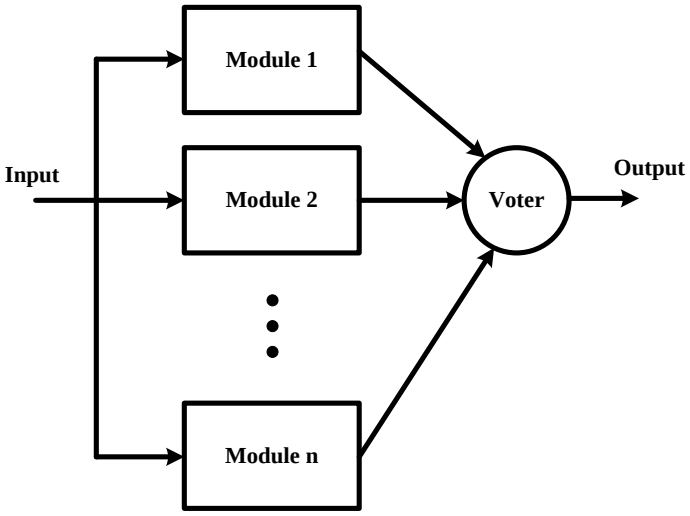

Fig. 7. NMR structure.

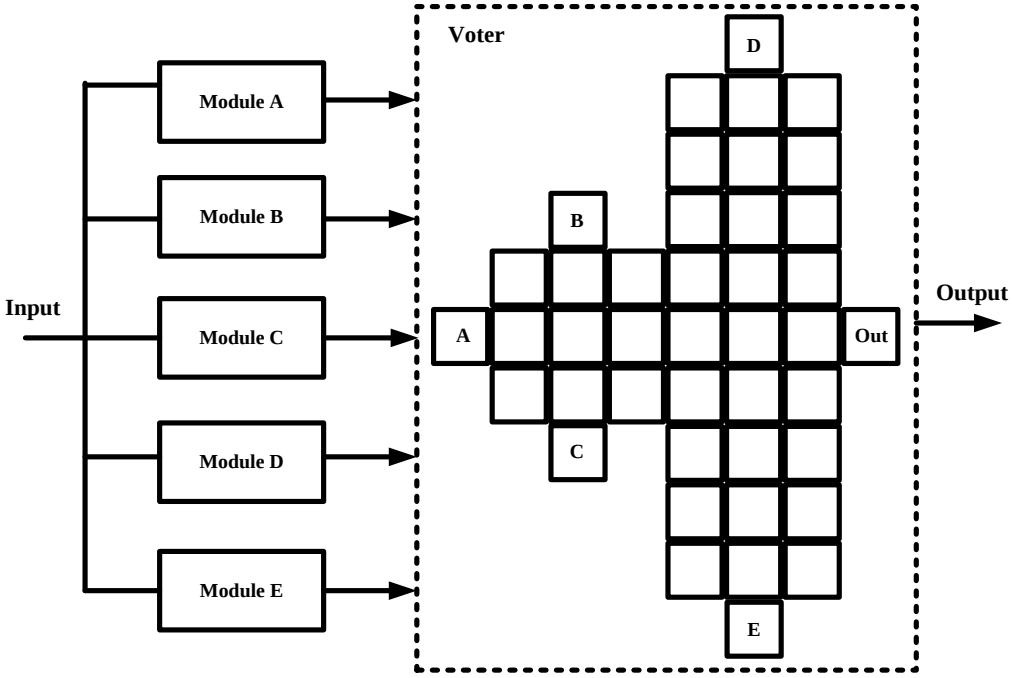

Fig. 8. Using fault tolerant five-input majority gate as a voter for 5MR structure.

Also, to overcome susceptibility to voter failure, we can use fault-tolerant n-input majority gate (Fig. 8) [9].

3.2. Active Hardware Redundancy for QCA Based Design

Active hardware redundancy techniques attempt to achieve fault tolerance by fault detection, fault location, and fault recovery. This approach does not prevent faults from producing errors within the system. Consequently, active approaches are most

common in applications that can tolerate temporary, erroneous results as long as the system reconfigures and regains its operational status in a satisfactory length of time.

One common form of active redundancy is duplication with a comparison scheme shown in Fig. 9. The basic concept of duplication with the comparison is to develop two pieces of hardwares, which make them perform the same computations in parallel, and compare the results of those computations. In the event of disagreement, an error message is generated. In this form, the duplication concept can only detect the existence of faults, not tolerate them because there is no method for determining which of the two modules is faulty. However, this method can be used as the fundamental fault detection technique in an active redundancy approach.

In QCA- based design, a comparison unit can be performed using two-input XOR gates. The two bits are compared toprovide inputs to the XOR gate, and the output of the gate is '1' if the two bits disagree and '0' if they agree.

3.3. Hybrid Hardware Redundancy for QCA Based Design

The fundamental concept of hybrid hardware redundancy is to combine the features of both active and passive approaches. Fault masking is used to prevent the system from producing erroneous results, fault detection and fault location. Furthermore, fault recovery is used to reconfigure the system in the event of faults. It is obvious that hybrid redundancy is very expensive in terms of the amount of hardware required to implement a system. Although, in QCA circuits, because of the low cost and availability of cells, this method is appropriate.

One approach to hybrid redundancy is NMR with spares. This idea provides a basic core of n modules in voting or a form of voting and configuration. In addition, spares are provided to replace failed units in the NMR core. The benefit of NMR with spares is that a voting configuration can be restored after a fault has occurred.

The NMR with spares technique is illustrated in Fig. 10. The system remains in the basic NMR configuration until the disagreement detector determines that a faulty unit exits. A module that disagrees with the majority is labeled as faulty and removed from the NMR core. A spare unit is then switched in to replace the faulty module.

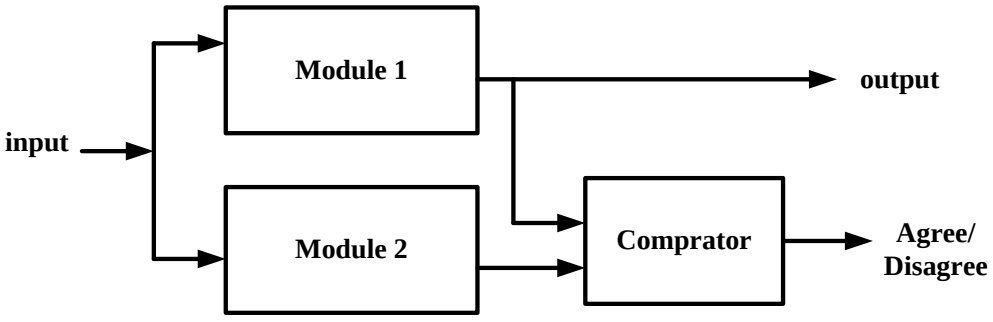

Fig. 9. Duplication with comparison technique.

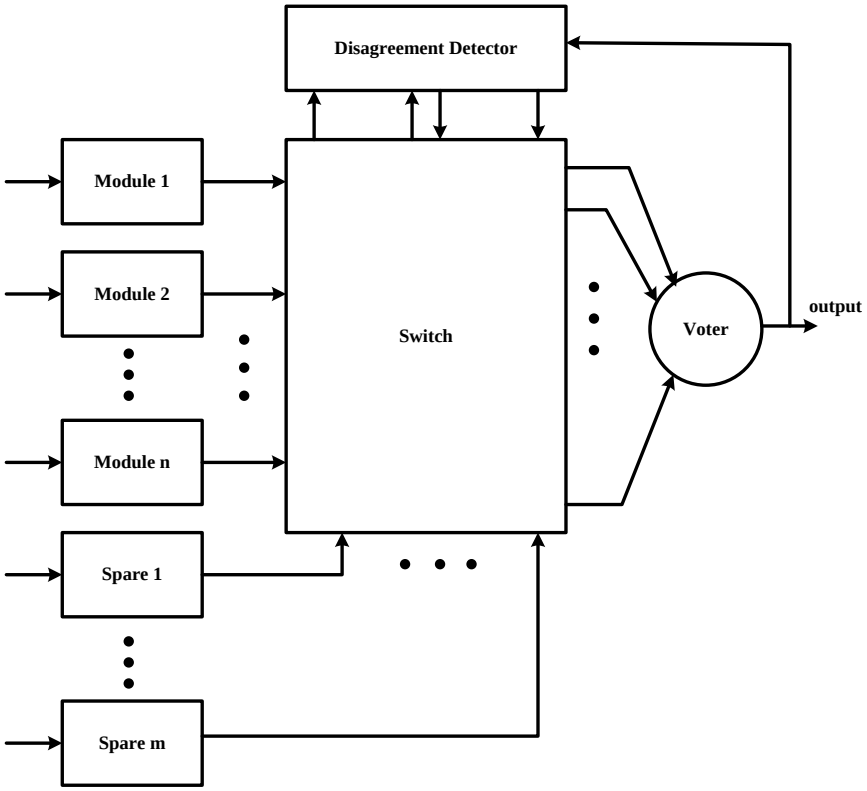

Fig. 10. NMR with spares structure.

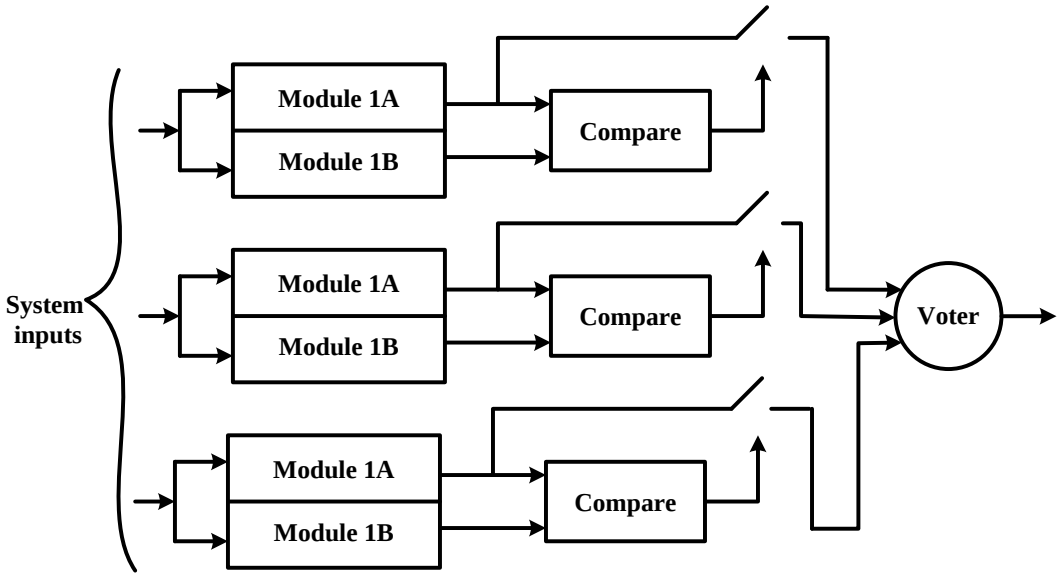

Fig. 11. The triple-duplex architecture.

Another hybrid redundancy method is called the triple-duplex architecture because it combines duplication with comparison and TMR. The use of TMR allows faults to be masked and continuous, and the use of duplication with comparison allows faults to be detected and faulty modules are removed from the process (Fig. 11).

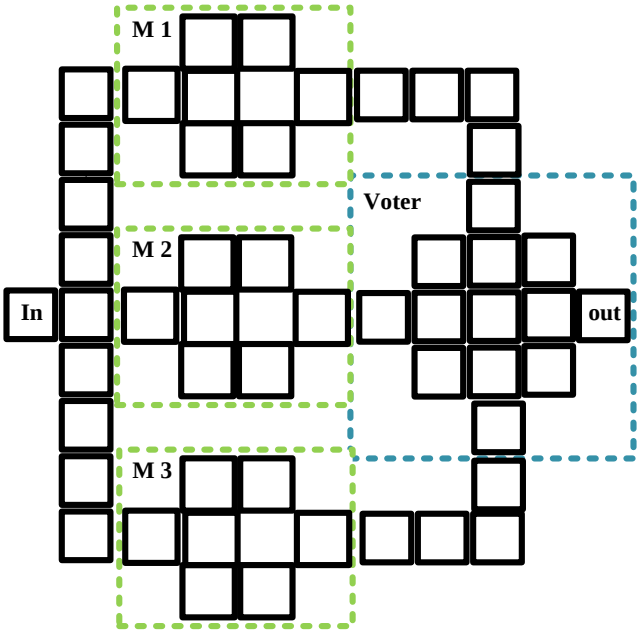

Fig. 12. Block diagram of the proposed inverter using the TMR structure with fault-tolerant three-input majority gate.

4. FAULT-TOLERANT QCA CIRCUITS

As already mentioned, two main logic elements for implementation with QCA are inverter and majority gate. In this section, a new design of fault-tolerant inverter is investigated (Fig. 12). The reason for the inverter is the simplicity of design and implementation. The proposed method can be used to design all QCA arithmetic circuits that are omitted due to lack of space.

In this scheme, the TMR structure with a fault-tolerant three-input majority gateis used. If one of the modules becomes faulty, the two remaining fault-free modules mask the result of the faulty module when the majority vote is performed. Also, by using fault-tolerant three-input majority gate as a voter, SPF will be overcome and this design can tolerate multi faults.

The presented scheme is justified based on simulation results.

4.1. Simulation and Practical Results

For the proposed circuit layout and functionality checking, a simulation tool for QCA circuits, QCADesigner version 2.0.3 [32], is used. The following parameters are used for bistable approximation: cell size =18 nm, number of samples =50000, convergence tolerance =0.0000100, radius of effect =65.000000 nm, relative permittivity =12.900000, clock high =9.800000e-022 J, clock low =3.800000e-023 J, clock shift =0, clock amplitude factor=2.000000, layer separation =11.500000 and maximum iterations per sample =100. Most of the mentioned parameters are default values in QCADesigner.

Figs. 13 and 14 show layout and simulation results of the proposed design.

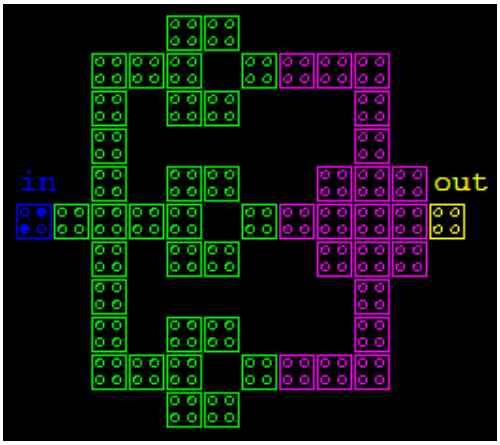

Fig.13. The layout of a proposed inverter using TMR structure with fault-tolerant three-input majority gate.

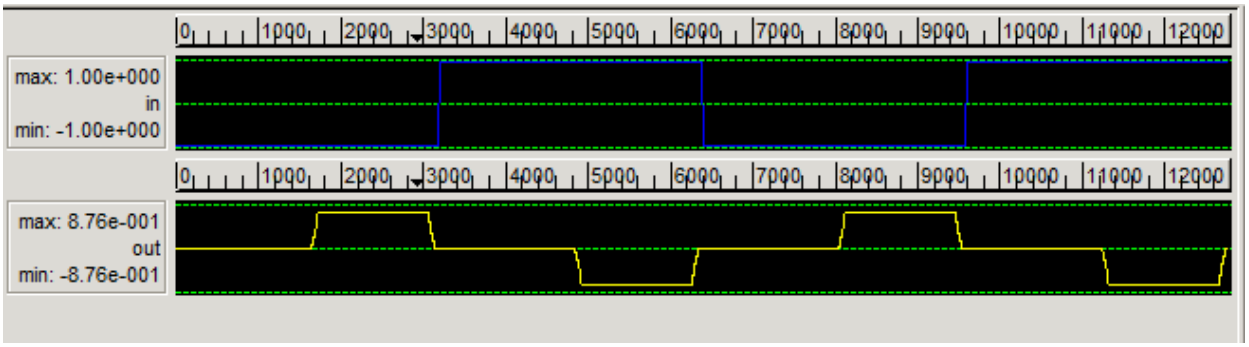

Fig.14. Simulation results for proposed inverter using TMR structure with fault-tolerant three-input majority gate.

Table 1. ***Comparison of QCA inverters.***

	Cell count	Fault tolerance	SPF free
Previous design (Fig. 1)	8	No	No
Proposed design 1 (Fig. 3b)	44	Yes	No
Proposed design 2 (Fig. 12)	52	Yes	Yes

The layout of previous designs and their simulation results are prepared and compared in this section. Simulation results reveal that the proposed design is more robust than the previous designs. Table 1 demonstrated the differences between the traditional QCA design with presented designs.

5. CONCLUSION

The probability of fault occurrences is growing as moving towards nanoscale. This paper demonstrated fault tolerance techniques for designing robust QCA digital circuits and systems by using hardware redundancy. In summary, it can be stated that

no single method is suitable for all kinds of faults and the best fault-tolerance can be gained by utilizing a variety of different fault tolerance techniques.

REFERENCES

[1] Ma X., and Lombardi F., "Fault-Tolerant Schemes for QCA Systems", IEEE International Symposium on Defect and Fault Tolerance of VLSI Systems, 236 – 244 (2008).

[2] Momenzadeh, Ottavi M., and Lombardi F.; "Modeling QCA defects at molecular-level in combinational circuits", 20th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems, 208 – 216 (2005).

[3] Hashemi S., Farazkish R., and Navi K., "New Quantum Dot Cellular Automata Cell Arrangements", Journal of Computational and Theoretical Nanoscience, Vol. 10, 798–809 (2013).

[4] Tougaw P. D., and Lent C. S., "Logical Devices Implemented Using Quantum Cellular Automata", Journal of Applied Physics 75, 1818-1825 (1994).

[5] Lehtonen T., Plosila J., and Isoaho J., "On fault tolerance techniques towards nanoscale circuits and systems", TUCS Technical Report, No. 708 (2005).

[6] Patitz Z. D., "Fault-Tolerant Quantum-dot Cellular Automata Majority Gate Design", Diss. Oklahoma State University (2006).

[7] Johnson B.W., "Design & analysis of fault-tolerant digital systems", Addison-Wesley Longman Publishing Co., ISBN: 0-201-07570-9 (1988).

[8] Farazkish R., Khodaparast F., Navi K., and Jalali A., "Design and Characterization of a Novel Inverter for Nanoelectronic Circuits", International Conference on Nanotechnology: Fundamentals and Applications 219 (2010).

[9] Farazkish R., and Navi K., "New Efficient Five-Input Majority Gate for Quantum-Dot Cellular Automata", Journal of Nanoparticle Research 14:1252 (2012a).

[10] Farazkish, R., Sayedsalehi, S., and Navi, K. "Novel design for quantum dots cellular automata to obtain fault-tolerant majority gate", Journal of Nanotechnology, doi: 10.1155/ 943406 (2012b)

[11] Farazkish R., "A New Quantum-Dot Cellular Automata Fault-Tolerant Five-Input Majority Gate", Journal of Nanoparticle Research 16:2259 (2014).

[12] Farazkish R., and Khodaparast F., "Design and characterization of a new fault-tolerant full-adder for quantum-dot cellular automata", Microprocessors and Microsystems J., doi:10.1016/j.micpro.2015.04.004 (2015).

[13] Farazkish R., "A new quantum-dot cellular automata fault-tolerant full-adder", J. Comput. Electr. 14,506–514 (2015).

[14] Navi K., Farazkish R., Sayedsalehi S., and Azghadi M. R., "A new quantum-dot cellular automata full-adder", Elsevier Microelectronics Journal, doi:10.1016/ j.mejo.2010.07.003 (2010a).

[15] Navi K., Sayedsalehi S., Farazkish R., and Azghadi M. R., "Five-Input Majority Gate a New Device for Quantum-Dot Cellular Automata", Journal of Computational and Theoretical Nanoscience 7, 1546-1553 (2010b).

[16] Farazkish R., "Novel efficient fault-tolerant full-adder for quantum-dot cellular Automata", Int J Nano Dimens 9(1):58–67 (2018).

[17] Roohi A., DeMara R. F., and Khoshavi N., "Design and evaluation of an ultra-area-efficient fault-tolerant QCA full adder", Microelectronics Journal, Vol. 46, 531–542 (2015).

[18] Farazkish R., "Fault-tolerant adder design in quantum-dot cellular automata", Int J Nano Dimens 8(1):40–48 (2017).

[19] Sen B., Dutta M., Mukherjee R., Kumar Nath R., Prakash Sinha R., and Sikdar B.K., "Towards the design of hybrid QCA tiles targeting high fault tolerance", Journal of Computational Electronics, Vol. 15, Issue 2, 429-445 (2016).

[20] Armstrong C. D., and Humphreys W. M., "The Development of Design Tools for Fault-Tolerant Quantum-Dot Cellular Automata Based Logic", 2nd Int'l Workshop on Quantum Dots for Quantum Computing and Classical Size Effect Circuits (2003a).

[21] Armstrong C. D., Humphreys W. M., and Fijany A., "The Design of Fault-Tolerant Quantum-Dot Cellular Automata Based Logic", 11th NASA Symposium on VLSI Design (2003b).

[22] Dalui M., Sen B., and Sikdar B. K., "Fault-Tolerant QCA Logic Design With Coupled Majority-Minority Gate", International Journal of Computer Applications 1 No. 29 (2010).

[23] Fijany A., and Toomarian B. N., "New Design for Quantum Dots Cellular Automata to Obtain Fault-Tolerant Logic Gates", Journal of Nanoparticle Research 3, 27-37 (2001).

[24] Sen B., Ganeriwal S., and Sikdar B. K., "Reversible Logic-Based Fault-Tolerant Nanocircuits in QCA", ISRN Electronics, Article ID 850267 (2013).

[25] Tahoori M. B., Momenzadeh M., Huang J., and Lombardi F., "Defects and faults in quantum cellular automata at the nanoscale" IEEE VLSI Test Symposium 4 (2004).

[26] Farazkish R., "Robust and reliable design of bio-nanorobotic systems", Microsyst Technol., https://doi.org/10.1007/s00542-018-4049-1, (2018).

[27] Zarei M., "Traffic-Centric Mesoscopic Analysis of Connectivity in Vehicular Adhoc Networks", *The Computer Journal*, DOI:10.1093/comjnl/bxz094, (2019).

[28] Zarei M., Rahmani A. M., "Analysis of vehicular mobility in a dynamic free-flow highway", *Vehicular Communications*, 7, 51-57, (2017).

[29] Zarei M., Rahmani A. M., "Renewal Process of Information Propagation in Delay Tolerant VANETs", *Wireless Personal Communications*, 89 (4), (2016), 1045-1063.

[30] Zarei M., Rahmani A. M., Samimi H., "Connectivity analysis for dynamic movement of vehicular ad hoc networks", *Wireless Networks*, 23 (3), (2016), 843-858.

[31] Lent C. S., and Tougaw P. D., "Lines of interacting quantum-dot cells: a binary wire", Journal of Applied Physics, 6227-6233 (1993).

[32] QCADesigner Home Page <www.atips.ca/projects/qcadesigner/>.