

# Fault Models and Test Approaches in Reversible Logic Circuits



H. M. Gaur, T. N. Sasamal, A. K. Singh and A. Mohan

**Abstract** The operations in reversible circuits are fully controllable and observable due to their bijective property which provides cursive testing. Testing can be categorized into two behavioral schemes: (i) Online testing, where the detection of faults within the circuit is carried out during its operation, (ii) Offline testing, where test vectors are applied after extracting the circuit out from its normal operation and the correct output values are known. Test data minimization for a specific kind of fault model such as stuck-at, bridging, missing gate, cross-point, and cell faults, is an important factor in this type of testing using meta-heuristic algorithms and circuit modification methodologies. Diverse varieties of fault families in reversible logic circuits and an exclusive study of testable design advances for these faults are portrayed in this chapter. Plentiful approaches were projected under two extensive classifications to meet the challenge. The methodologies are alleged to coat almost all the faults and their sub kind by exploiting the properties of reversible gates and circuits. The objective is to minimize testing overhead, which can be achieved by reducing the cost metrics utilized for testability.

---

H. M. Gaur (✉)

Department of Electronics & Communication Engineering, ABES Institute of Technology, Ghaziabad (Delhi NCR) 201009, India  
e-mail: [leoharimohan84@gmail.com](mailto:leoharimohan84@gmail.com)

T. N. Sasamal

Department of Electronics and Communication Engineering, NIT, Kurukshetra, India  
e-mail: [sasamal.trailokyanath@gmail.com](mailto:sasamal.trailokyanath@gmail.com)

A. K. Singh

Department of Computer Applications, NIT, Kurukshetra, India  
e-mail: [ashutosh@nitkr.ac.in](mailto:ashutosh@nitkr.ac.in)

A. Mohan

Department of Electronics Engineering, IIT-BHU, Varanasi, India  
e-mail: [profanandmohan@gmail.com](mailto:profanandmohan@gmail.com)

© Springer Nature Singapore Pte Ltd. 2020

A. K. Singh et al. (eds.), *Design and Testing of Reversible Logic*, Lecture Notes in Electrical Engineering 577, [https://doi.org/10.1007/978-981-13-8821-7\\_9](https://doi.org/10.1007/978-981-13-8821-7_9)

## 1 Introduction

In quantum circuits, de-coherence enforces the quantum states of bits to decay. It results in loss of information that causes faults. Due to this phenomenon, these circuits are more prone to faults than conventional circuits [1]. As reversible circuits have direct relations to quantum circuits, they are largely prone to several transient and permanent faults which cause single and multiple point failures. Testing assures the correct operations of logic circuits which show its untamed necessity and their excellence. The operations in reversible circuits are fully controllable and observable due to their bijective property which provides cursive testing. It has also been extensively studied since the last decade by exploiting this property for the identification of several types of fault models. These fault models are stuck-at, bridging, missing gate, cross-point, and cell faults. Testing can be categorized in two behavioral schemes. First is online testing, where the detection of faults within the circuit is carried out during its operation [2, 3]. It provides a built-in self-testable environment through design methodology and circuit modification for the detection of fault models in terms of single and multiple-bit-flip faults. Second is offline testing, where test vectors are applied after extracting the circuit out from its normal operation and the correct output values are known. Test data minimization for a specific kind of fault model is an important factor in this type of testing using meta-heuristic algorithms and circuit modification methodologies.

Several novel paradigms have been presented in both the area of testing reversible logic circuits. Built-in testable environment are provided over pristine fundamental (MCT and MCF) and new gates-based design methodologies [4–11] and modification principles [12–16]. Test data minimization is achieved over new deterministic [17–27], randomized test pattern generation algorithms [28–31] and modification techniques [32–37] for respective faults. It is noticed that, the technique of parity checking [38] is dominating for the recognition of single/multiple bit faults in online testing, whereas the bijectivity property of reversible logic circuits is utilized in case of offline testing. The reduction of operating cost has been achieved to some needful extent in all the proposed approaches with respect to prior ones to narrow the compensation with overall testing overheads. Fault tolerance is also the architectural attribute of a digital system that maintains proper functioning of a logic machine during various kinds of failures [39, 40]. The inclusion of fault tolerance abilities during the design and synthesis process is also in the development phase [41–46].

Testing impetus a dramatic increment in the utilization of resources in terms of cost and power requirements. In electronic circuits, it leads to a large increment in operating costs from their original circuits like gates and wires, which drastically increases size and power. It accounts 30–60% of the cost of manufacturing electronic devices by consuming extra hardware and resources [47]. During the construction of built-in testable reversible circuits over design methodology or modification, the operating cost increases in terms of gates, wires, ancilla input, and garbage output.

Test data minimization over meta-heuristic algorithms consume excess time and a separate hardware to produce test sets. Test data minimization over modification includes incorporation of additional hardware as well as time to run test vectors for the respective type of fault. Hence, necessary actions should be taken to reduce excess usage of hardware and time to lower the overall testing overheads and narrow the compensation with power.

## 2 Fault Models

Faults are a type of deficiency in a circuit which reflects in the imperfect behavior and functional abilities of a system long lastingly or for a limited period of time. They can be occurred due to any human and environmental issues [2] and termed as permanent and nonpermanent faults respectively. A fault model depicts the category of fault occurrence in a circuit and guides in identifying the target for testing. Numerous fault models are projected in the past along with respective categorization in reversible circuits as labeled out in Fig. 1. Following is the short description of these fault models given in this figure.

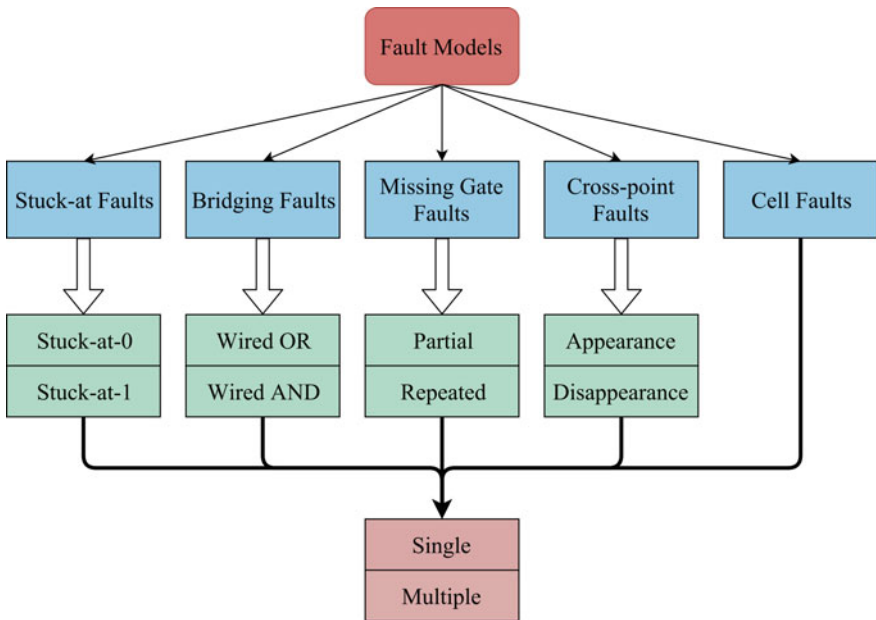
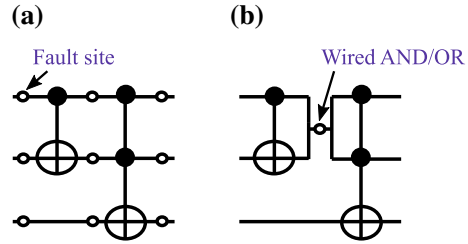


Fig. 1 Faults in reversible circuits

**Fig. 2** Stuck-at and bridging faults



### Stuck-At Fault

Alike conventional logic circuits, this fault occurred when any wire in a circuit get settled on a single logic bits 0 or 1 are termed as stuck-at-0 (S-a-0) or stuck-at-1 (S-a-1) faults respectively. The faults can be aroused on single or multiple nodes at a single time which can be either of same or of both types. In reversible circuits, these faults occurred at the input of the gates and the input/output wires of the circuit. The total number of stuck-at faults (S-a-0 and S-a-1) in the circuit is given by  $\{2 \times (\sum_{i=1}^G g_i + n)\}$  where,  $G$  are the number of gates contained by the  $n$  wires circuit and  $g_i$  is the gate size of  $i$ th gate. For example, there are nine possible sites for this type of fault occurrence are shown by tiny circles in Fig. 2a.

### Bridging Fault

This category of fault arises when two or more adjacent wires of a circuit get physically come in contact or linked and resemble the abilities of wired AND or OR interconnections that consequence into erroneous functionality. The illustration is provided in Fig. 2b which, these faults may occur on a couple of wires or on multiple wires at respective levels of the circuit which are termed as single intra-level and multiple intra-level bridging faults. The number of levels of the circuit is governed by number of gates. For instance, a circuit containing  $G$  gates there will be  $G + 1$  distinct levels which include the input of each gate and the final output. The number of single intra-level faults between a couple of wires are given by  ${}^n C_2$  and all single intra-level faults between a couple of wires in the circuit is given by  $\{(G + 1) \times {}^n C_2\}$ . The circuit represented in Fig. 2b has three levels and number of single intra-level faults between a couple of wires are 9.

### Missing Gate Fault

When a gate in a circuit fully fails to perform its characteristics or act completely disappeared from the circuit is called a single missing-gate fault (SMGF), as illustrated in Fig. 3a. As a result, the output changes to a faulty value as illustrated in the figure where the fault-free/faulty logic values are written on every level of the circuit.

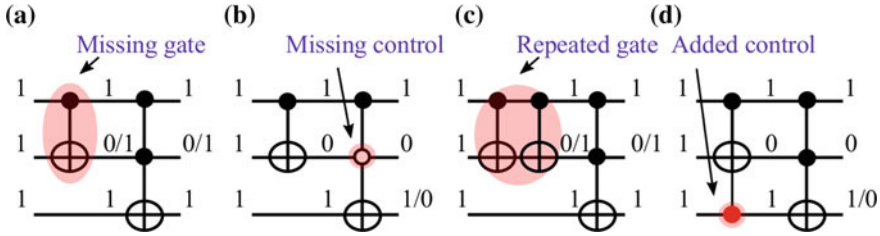


Fig. 3 Missing gate and cross-point faults

Multiple missing gate fault (MMGF) is the disappearance of two or more successive gates. Maximum occurrence of SMGF in the circuit is equal to number of gates ( $G$ ) contained by the circuit and number of MMGF is given by  $\{G(G + 1)\} \setminus 2 - G$ . Hence, the total number MGF (SMGF and MMGF) in the circuit can be calculated by  $\{G(G + 1)\} \setminus 2$ . For instance, number of MGF in the circuit for Fig. 3a is 3.

The of missing gate faults are also labeled as partial missing gate fault (PMGF) and repeated gate fault (RGF) by the researchers of this area. PMGF occurred when any control point is disappeared from a gate and RGF is a replication of operation by a single gate for multiple instances. The illustrations of these faults can be acknowledged in Fig. 3b and c respectively. There is no effect of RGF on the circuit if the instances are even in number. For odd number of instances, RGF will result as that of SMGF in the circuit.

**Cross-Point Fault**

This fault is associated to the nonfunctioning and inclusion of control points of reversible gates in a circuit. These faults are referred to as appearance (AF) and disappearance (DF) types of faults in the circuit. The behavior of these faults can be acknowledged from the Figs. 3d and b respectively. Disappearance faults show similar tendencies as that of PMGFs, the difference seems in the names only according to the researchers from the past. Total number of single AF in the circuit can be calculated by  $(nG - \sum_{i=1}^G g_i)$  and number of single DF can be given by  $(\sum_{i=1}^G g_i - G)$ .

**Cell Fault**

Any inappropriate operation of a gate in a circuit which results in an incorrect output is termed as cell faults (CF). Here the gates are referred to as a cell. The foundation of these faults is belongs to fault modeling in cellular logic arrays and therefore these faults can be simply called by cell faults. There are multiple anonymous ways of the occurrence of this kind of fault in the circuit, hence the calculation of its number is redundant.

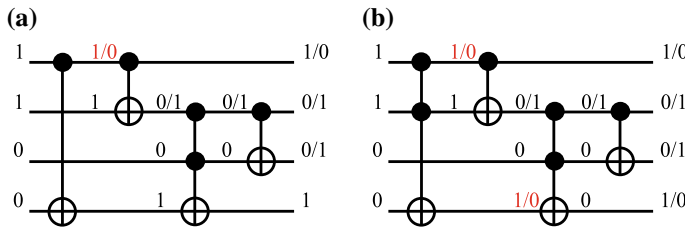


Fig. 4 Bit-flip faults (Source [48])

It can be noted that the occurrence of a type of fault in a circuit will result in flipping or inversion of bit values at the nodes after the gate where a fault occurrence has been taken place in the circuit. The alteration of these bits will affect single or multiple values at subsequent stages of the circuit and obviously cultivated toward the output. This kind of situation is referred to as a bit-flip fault can be termed as bit faults. When the value of a single bit is distorted, it can be a single-bit fault and if multiple values are flipped, it can be called as multiple bit faults. Figure 4a and b depict the patterns of changes of bit values because of respective faults in the circuit. Where, the propagation S-a-0 fault in red color for better understanding. The un-faulty/faulty bit values can be seen against each wire where the exactly these faults have affected.

It also can be concluded that the bit faults are meant for online testing as these faults detection will result in the detection of all types of fault models. The number of single-bit faults can be given by  $(\sum_{i=1}^G g_i + n)$ . It diminishes the requirements of designing a separate hardware/method for the detection of a given type of fault. The design for test complexities can be reduced as a consequence.

### 3 Fault Identification

Every fault model has its own role to change the behavior of the circuit. Their identification is based on the type of gates used and the test vector which changes the input–output logic values. For instance, if an input vector is not able to interrupt the functioning of a gate in the circuit, it cannot be used for identification of any faults. An applied test vector to the inputs of the circuit alters one or more bit logic values of the input wires of the gates and subsequent levels contained by it. The identification procedures for the different type of faults in MCT and MCF gates are explained as follows:

### 3.1 Fault Identification for MCT Gates

Considering an  $n$  wire MCT gate having  $(k_1, k_2, \dots, k_m)$  control inputs and target input  $T$ . Note that, there cannot be any bridging and cross-point faults in a NOT gate. Respective deterministic methodologies for the identifications of faults are explained below.

#### SAF Identification

Single stuck-at faults in MCT circuit is given by  $\{n + \sum_{i=1}^{GC} (k_{m_i}) + 2GC\}$ , where,  $GC$  is number of gates and  $k_{m_i}$  is control inputs of  $i$ th gate of the circuit. Assuming logic 0 and 1 values at all the fault sites, the  $n$  dimensional test vector of size 2 given by  $(0, 0, \dots, 0)(1, 1, \dots, 0)$  defines a test vector for the detection of all single and multiple type stuck-at faults of an MCT gate.

#### BrF Identification

Bridging faults is dependent on total number of wires in the circuit. All single intra-level bridging fault for an MCT gate is given by  $\{2(GC + 1) \times^n C_2\}$ . The detection is done by assuming complementary values between the two wires at every level of the circuit. The  $n$  dimensional test vector of size  $n$ , produced by shifting 0 value from first wire to next until it reaches to the end of test vector, given as  $(0, 1, \dots, 1), (1, 0, \dots, 1), \dots, (1, 1, \dots, 0)$  is complete for all single intra-level bridging faults of an MCT gate of a circuit.

#### MGF Identification

Single missing gate faults is equal to the gates present in the circuit. The detection principle for this type of fault is to assign logic values 1 to the control inputs and 1 (or 0) value to the target input of the gate in the circuit. The  $n$ -dimensional single test vector given as  $\{k_1, k_2, \dots, k_m, T\} = \{(1, 1, \dots, 1, 1)\}$  (or  $\{(1, 1, \dots, 1, 0)\}$ ) is complete for its detection.

#### CPF Identification

The number of single cross-point faults, either appearance or disappearance type, occurred in the circuit is given by  $N(n - 1)$ . The detection is achieved by assigning the combination of  $n$  test vectors keeping logic 1 to the  $m - 1$  control inputs and 1 value to target input of each gate at distinct levels of the circuit. Assuming logic 0 to the control input where the fault has been occurred, making rest all control at logic 1

and 1 (or 0) value to target input of the gate. The  $n$  dimensional test vector of size  $(n - 1)$  given as  $\{k_1, k_2, \dots, k_m, T\} = \{(0, 1, \dots, 1, 1), (1, 0, \dots, 1, 1), \dots, (1, 1, \dots, 0, 1)\}$  is complete for the detection of all CPF of the gate for  $n \geq 2$ .

### CF Identification

The  $2^n$  greedy permutation fix the detection of this type of fault and all  $n$  dimensional test vectors of size  $2^n$  are required to detect all the cell faults in the circuit.

### 3.2 Fault Identification for MCF Gates

Considering an  $n$  wire MCF gate having  $(k_1, k_2, \dots, k_m)$  control inputs and target inputs  $T_1$  and  $T_2$ . Note that, there will be no cross-point faults in a swap gate and no single wire MCF gate is available. Also, the test principles used for traditional and MCT based logic circuits can be applied for the detection of stuck-at faults, single intra-level bridging faults, and cell faults. Respective deterministic methodologies are explained below.

#### MGF Identification

Single missing gate faults in MCF circuit is equal to the sum of gates available in the circuit. The detection principle for this type of fault is to assign logic values 1 to the control inputs and complementary values to the two target inputs of the gate in the circuit. The  $n$  dimensional single test vector given as  $\{k_1, k_2, \dots, k_m, T_1, T_2\} = \{(1, 1, \dots, 1, 0, 1)\}$  is complete for its detection.

#### CPF Identification

The number of single cross-point faults, either appearance or disappearance type, occurred in the circuit is given by  $N(n - 2)$ . The detection is achieved by assigning the combination of  $n$  test vectors keeping logic 1 to the  $m - 2$  control inputs and complementary values to target input of each gate at every level of the circuit. Assuming logic 0 to the control input where the fault has been occurred, making rest all control at logic 1 and complementary values to the target inputs of the gate. The  $n$  dimensional test vector of size  $(n - 2)$  given as  $\{k_1, k_2, \dots, k_m, T_1, T_2\} = \{(0, 1, \dots, 1, 0, 1), (1, 0, \dots, 1, 0, 1), \dots, (1, 1, \dots, 0, 0, 1)\}$  is complete for the detection of all CPF of the gate for  $n \geq 3$ .



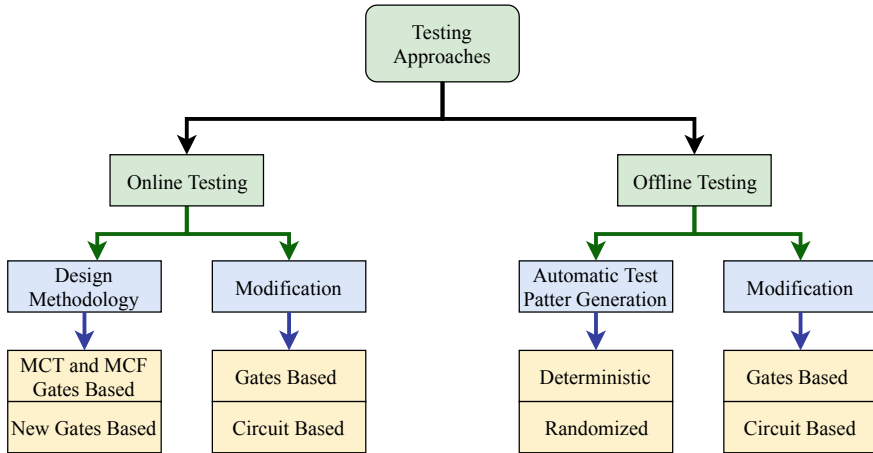


Fig. 5 Classification of testing approaches

## 4 Testing Approaches

Testing of the reversible circuit gaining grounds since more than a decade, where the researchers have been forwarded about the kind of faults that can be happed and developing their testing strategies in this emerging field. A collective information of testing approaches from the literature is provided in this section with respect to cost metrics and test complexity. The possible categorization of testing approaches for reversible circuits that have been proposed so far [49–51], are shown in Fig. 5. A brief about the contribution in these approaches are described in this section. It can be noted that mostly all the online testing approaches utilize parity checking technique. However, offline approaches exploit the bijective mapping property of reversible function. Number of inputs, gates, quantum cost, ancillas, and garbage are considered to evaluate the performance of respective testing approaches. Three more measures are included for the analysis of testing approaches: test size ( $s$ ), execution time ( $t$ ), and fault coverage ( $FC$ ) [51]. However, test size and execution time are used in case of ATPG approaches.

### 4.1 MCT and MCF Gates-Based Design Methodologies for Online Testing

The MCT and MCF are the fundamental gates and the final quantum decomposition is based on them. The design complexity of testable and quantum circuits are proven lesser due to this reason [52]. The design methodologies which provide built-in testability features includes a twofold MCT gate placement, design with MCF gates

and design with mixed MCTF gates methodologies were proposed [11, 53]. These methodologies depict a novel design paradigm that relies on the placement of MCT and MCF gates to generate a desired Boolean function. The placement methodology produce parity preserving circuits and are meant for single bit fault detection, which are called as soft errors in a broad sense.

#### ***4.2 New Gates Based Design Methodologies for Online Testing***

The objective of proposing testable gates (new gates) is to incorporate some additional input and output to achieve testability. Using minimal operating costs is the prime targets for their formulation. It can be noted that all the methods are meant single bit faults detection inside a circuit. These new gates are  $4 \times 4$  R1 and R2 whose combination formulates a testable block TB [4, 5] and testable gate OTG which can be used to produce testable cell CTSG. Two self-testable dual-rail coding scheme based gates are also presented [8] to remove the requirements of additional dual-rail checkers in prior methodologies. The gates LIG and FIG are introduced to establish testable information at the output for MGF faults without using parity checking [9]. The method for multi-bit errors is also formulated using concurrent error detection scheme [10].

#### ***4.3 Gate-Based Modification Methodologies for Online Testing***

The gates of the circuit are modified to achieve testability in these type of approaches. A modification of a reversible gate procedure to make them testable form is first adopted by the use of ETG (Extended Toffoli Gate) [54] is adopted. However, ETGs are formulated over photonics, the methodology utilizes two additional CNOT gates per MCT gate. Another method exploits the technique of cascading several stages of an identity gate to renovate a gate into a testable reversible cell TRC. Both of the methodologies exploits the phenomenon of parity preservation and generation for single bit fault detection in reversible circuits. A method of gates conversion for testability is presented utilizing the property of parity preserving gates rather converting for the same [48]. A methodology for the modification of Toffoli and Peres gates in corresponding testable form is also presented which ensures the nearly all multiple-bit faults detection [16].

#### ***4.4 Circuit-Based Modification Methodologies for Online Testing***

However, the circuit is modified when the modified gates are used for their formulation. In this particular approach complete circuit is circuit using the same testable viewpoints at all levels or some levels of the circuit. In this regard, a method of gates cascading was proposed for the modification of MCT circuits [55]. In this method, a gate of same size is cascaded after each gate with same control inputs and target on a new wire. This modification produces parity preserving reversible circuits where the detection of single bit faults can be achieved using two arrays of CNOT gates. The method is also proven better in performance when implemented using quantum cellular automata (QCA) platform.

#### ***4.5 Deterministic ATPG Approaches for Offline Testing***

Utilizing the concepts described in Sect. 3 for various types of faults detection, several algorithms are proposed. Complete test sets for detection of stuck-at and cell faults are formulated using practical heuristic over an integer level program (ILP) using binary variables [17, 18]. The test sets by obtained in lesser execution time and proven for minimal test size. An exact ATPG is developed over the emerging ion-trap quantum computing technology for obtaining minimal test set for missing gate faults [19]. Principle of comparison of change in output due to missing gates based and the concepts of Boolean difference method, algorithms are also developed for single and multiple missing gate faults detection respectively [20, 21]. The test generation algorithm is formulated for the detection of bridging faults on the basis of proposed block division method [22]. An algorithm which produces a constant universal test vector (of size  $n$ ) is also presented for single and multiple input bridging faults [23]. It is noted that, nearly all the presented methodologies are meant for MCT circuit.

#### ***4.6 Randomized ATPG Approaches***

Creation of and modification of existing ATPGs by including random variables is another method, where researchers also proposed several solutions for achieving testability in MCT circuits. Solving an NP-hard problem for obtaining minimal size test set stuck-at fault detection in NCT circuits is proposed [28]. Greedy heuristic and exact branch and bound algorithm are utilized to detect the missing gate faults [29]. Ping-pong method is proposed to generate a test set for missing gate type of faults [30]. The detection of cross-point faults is also generalized by proposing a randomized ATPG [31].

#### ***4.7 Gates Based Modification Methodologies for Offline Testing***

As per the authors knowledge and review of this area, transformation of a reversible to corresponding AND-EXOR irreversible circuit is the only method presented in the literature for the detection of single intra-level bridging faults [36]. MCT circuits are first decomposed in the irreversible PPRM AND-EXOR network. Faults are assumed between the wires of AND-EXOR circuit and detection strategy is applied at its different levels.

#### ***4.8 Circuit Based Modification Methodologies for Offline Testing***

Following the identification approaches of the faults in reversible circuits, the circuit is modified in such a method that the applied test vector propagates till the last level of the circuit in this type of methodology. Two circuit modification methodologies are formulated for single and multiple stuck-at faults detection in MCT circuits. A universal test set (UTS) and a complete test set (CTS) test sizes of 2 are proposed along with the methodologies. An efficient adding a gate methodology is also presented for the detection and location of these faults with a minimal test set for their detection [56]. Techniques are realized by the inclusion CNOT gates for missing gate fault detection by single test vector [17, 29] and universal test set of size  $(n + 1)$  [34, 35].

### **5 Summary**

Diverse varieties of fault families in reversible logic circuits and an exclusive study of testable design advances for these faults are portrayed in this chapter. As a new technological change, an outstanding awareness is depicted by the researchers of the area for finding a solution for the notation and detection of these faults. Plentiful approaches were projected under in two extensive classifications to meet the challenge. The methodologies are alleged to coat almost all the faults and their sub kind by exploiting the properties of reversible gates and circuits. The objective is to minimize testing overhead which can be achieved by reducing the cost metrics utilized for testability. Following are the key points discussed in this chapter which includes objectives, notations, and result analysis for design and testing of reversible logic circuits:

## References

1. Nielsen MA, Chuang IL (2011) Quantum computation and quantum information: 10th anniversary edition, 10th edn. Cambridge University Press, New York
2. Hurst SL (1998) VLSI Testing: digital and mixed analogue/digital techniques. The Institution of Electrical Engineers, London
3. Jha NK, Gupta S (2003) Testing of digital systems. Cambridge University Press, Cambridge
4. Vasudevan DP, Lala PK, Parkerson JP (2004) Online testable reversible logic circuit design using nand blocks. In: 19th IEEE international symposium on defect and fault tolerance in VLSI systems, 2004. DFT 2004. Proceedings, pp 324–331. <https://doi.org/10.1109/DFTVS.2004.1347856>
5. Vasudevan DP, Lala PK, Di J, Parkerson JP (2006) Reversible-logic design with online testability. IEEE Trans Instrum Meas 55(2):406–414. <https://doi.org/10.1109/TIM.2006.870319>
6. Hasan M, Islam AKMT, Chowdhury AR (2009) Design and analysis of online testability of reversible sequential circuits. In: 2009 12th international conference on computers and information technology, pp 180–185. <https://doi.org/10.1109/ICCIT.2009.5407143>
7. Thapliyal H, Vinod AP (2007) Designing efficient online testable reversible adders with new reversible gate. In: 2007 IEEE international symposium on circuits and systems, pp 1085–1088. <https://doi.org/10.1109/ISCAS.2007.378198>
8. Farazmand N, Zamani M, Tahoori MB (2010) Online fault testing of reversible logic using dual rail coding. In: 2010 IEEE 16th international on-line testing symposium, pp 204–205. <https://doi.org/10.1109/IOLTS.2010.5560205>
9. Zamani M, Tahoori MB (2011) Online missing/repeated gate faults detection in reversible circuits. In: 2011 IEEE international symposium on defect and fault tolerance in VLSI and nanotechnology systems, pp 435–442. <https://doi.org/10.1109/DFT.2011.56>
10. Thapliyal H, Ranganathan N (2010) Reversible logic based concurrent error detection methodology for emerging nanocircuits. In: 10th IEEE international conference on nanotechnology, pp 217–222. <https://doi.org/10.1109/NANO.2010.5697743>
11. Gaur HM, Singh AK (2016) Design of reversible circuits with high testability. Electron Lett 52(13):1102–1104. <https://doi.org/10.1049/el.2016.0161>
12. Nayeem NM, Rice JE (2011c) A simple approach for designing online testable reversible circuits. In: Proceedings of 2011 IEEE Pacific rim conference on communications, computers and signal processing, pp 85–90. <https://doi.org/10.1109/PACRIM.2011.6032872>
13. Nayeem NM, Rice JE (2011a) Online fault detection in reversible logic. In: 2011 IEEE international symposium on defect and fault tolerance in VLSI and nanotechnology systems, pp 426–434. <https://doi.org/10.1109/DFT.2011.55>
14. Nayeem NM, Rice JE (2013) Online testable approaches in reversible logic. J Electron Test 29(6):763–778. <https://doi.org/10.1007/s10836-013-5399-3>
15. Nayeem NM, Rice JE (2012) A new approach to online testing of tgfsoP-based ternary toffoli circuits. In: 2012 IEEE 42nd international symposium on multiple-valued logic, pp 315–321. <https://doi.org/10.1109/ISMVL.2012.57>
16. Sen B, Das J, Sikdar BK (2012) A DFT methodology targeting online testing of reversible circuit. In: 2012 international conference on devices, circuits and systems (ICDCS), pp 689–693. <https://doi.org/10.1109/ICDCSyst.2012.6188661>
17. Patel KN, Hayes JP, Markov IL (2003) Fault testing for reversible circuits. In: Proceedings of the 21st VLSI test symposium, 2003, pp 410–416. <https://doi.org/10.1109/VTEST.2003.1197682>
18. Patel KN, Hayes JP, Markov IL (2004) Fault testing for reversible circuits. IEEE Trans Comput-Aided Des Integr Circuits Syst 23(8):1220–1230. <https://doi.org/10.1109/TCAD.2004.831576>
19. Polian I, Fiehn T, Becker B, Hayes JP (2005) A family of logical fault models for reversible circuits. In: 14th Asian test symposium (ATS'05), pp 422–427. <https://doi.org/10.1109/ATS.2005.9>

20. Kole DK, Rahaman H, Das DK, Bhattacharya BB (2011) Derivation of automatic test set for detection of missing gate faults in reversible circuits. In: 2011 international symposium on electronic system design (ISED). IEEE, pp 200–205
21. Mondal B, Kole DK, Das DK, Rahaman H (2014) Generator for test set construction of SMGF in reversible circuit by boolean difference method. In: 2014 IEEE 23rd Asian test symposium, pp 68–73. <https://doi.org/10.1109/ATS.2014.24>
22. Rahaman H, Kole DKKDK, Das DK, Bhattacharya BB (2007) Optimum test set for bridging fault detection in reversible circuits. In: 16th Asian test symposium (ATS 2007), pp 125–128. <https://doi.org/10.1109/ATS.2007.91>
23. Sarkar P, Chakrabarti S (2008) Universal test set for bridging fault detection in reversible circuit. In: 2008 3rd international design and test workshop, pp 51–56. <https://doi.org/10.1109/IDT.2008.4802464>
24. Babu HMH, Mia MS, Biswas AK (2017) Efficient techniques for fault detection and correction of reversible circuits. *J Electron Test*, pp 1–15. <https://doi.org/10.1007/s10836-017-5679-4>
25. Fujita M (2015) Detection of test patterns with unreachable states through efficient inductive-invariant identification. In: 2015 IEEE 24th Asian test symposium (ATS), pp 31–36. <https://doi.org/10.1109/ATS.2015.13>
26. Fujita M (2015) Automatic identification of assertions and invariants with small numbers of test vectors. In: 2015 33rd IEEE international conference on computer design (ICCD), pp 463–466. <https://doi.org/10.1109/ICCD.2015.7357149>
27. Burchard J, Erb D, Singh AD, Reddy SM, Becker B (2017) Fast and waveform-accurate hazard-aware sat-based TSOF ATPG. In: Design, automation test in Europe conference exhibition (DATE), 2017, pp 422–427. <https://doi.org/10.23919/DATE.2017.7927027>
28. Tabei K, Yamada T (2009) On generating test sets for reversible circuits. In: 2009 international conference on computer engineering systems, pp 94–99. <https://doi.org/10.1109/ICCES.2009.5383305>
29. Hayes JP, Polian I, Becker B (2004) Testing for missing-gate faults in reversible circuits. In: 13th Asian test symposium, pp 100–105. <https://doi.org/10.1109/ATS.2004.84>
30. Zamani M, Tahoori MB, Chakrabarty K (2012) Ping-pong test: Compact test vector generation for reversible circuits. In: 2012 IEEE 30th VLSI test symposium (VTS), pp 164–169. <https://doi.org/10.1109/VTS.2012.6231097>
31. Zhong J, Muzio JC (2006) Analyzing fault models for reversible logic circuits. In: 2006 IEEE international conference on evolutionary computation, pp 2422–2427. <https://doi.org/10.1109/CEC.2006.1688609>
32. Chakraborty A (2005) Synthesis of reversible circuits for testing with universal test set and c-testability of reversible iterative logic arrays. In: 18th international conference on VLSI design held jointly with 4th international conference on embedded systems design, pp 249–254. <https://doi.org/10.1109/ICVD.2005.158>
33. Ibrahim M, Chowdhury AR, Babu HMH (2008) Minimization of CTS of k-CNOT circuits for SSF and MSF model. In: 2008 IEEE international symposium on defect and fault tolerance of VLSI systems, pp 290–298. <https://doi.org/10.1109/DFT.2008.38>
34. Rahaman H, Kole DK, Das DK, Bhattacharya BB (2008) On the detection of missing-gate faults in reversible circuits by a universal test set. In: 21st international conference on VLSI design (VLSID 2008), pp 163–168. <https://doi.org/10.1109/VLSI.2008.106>
35. Rahaman H, Kole DK, Das DK, Bhattacharya BB (2011) Fault diagnosis in reversible circuits under missing-gate fault model. *Comput Electr Eng* 37(4):475–485. <https://doi.org/10.1016/j.compeleceng.2011.05.005>
36. Chakraborty A (2010) Testing of bridging faults in AND-EXOR based reversible logic circuits. CoRR [arXiv:abs/1009.5098](https://arxiv.org/abs/1009.5098)
37. Bubna M, Goyal N, Sengupta I (2007) A DFT methodology for detecting bridging faults in reversible logic circuits. In: TENCON 2007 - 2007 IEEE region 10 conference, pp 1–4. <https://doi.org/10.1109/TENCON.2007.4428915>
38. Parhami B (2006) Fault-tolerant reversible circuits. In: 2006 fortieth asilomar conference on signals, systems and computers, pp 1726–1729. <https://doi.org/10.1109/ACSSC.2006.355056>

39. Avizienis A (1978) Fault-tolerance: the survival attribute of digital systems. *Proc IEEE* 66(10):1109–1125. <https://doi.org/10.1109/PROC.1978.11107>
40. Mathew J, Singh J, Taleb AA, Pradhan DK (2008) Fault tolerant reversible finite field arithmetic circuits. In: 2008 14th IEEE international on-line testing symposium, pp 188–189. <https://doi.org/10.1109/IOLTS.2008.35>
41. Vasudevan DP, Lala PK, Parkerson JP (2005a) The construction of a fault tolerant reversible gate for quantum computation. In: 5th IEEE conference on nanotechnology, 2005, vol 1, pp 112–115. <https://doi.org/10.1109/NANO.2005.1500705>
42. Vasudevan DP, Lala PK, Parkerson JP (2005) Fault tolerant quantum computation with new reversible gate. In: Technical proceedings of the 2005 NSTI nanotechnology conference and trade show, vol 3, pp 744–747
43. Boykin PO, Roychowdhury VP (2005) Reversible fault-tolerant logic. In: 2005 international conference on dependable systems and networks (DSN'05), pp 444–453. <https://doi.org/10.1109/DSN.2005.83>
44. Zamani M, Farazmand N, Tahoori MB (2011) Fault masking and diagnosis in reversible circuits. In: 2011 sixteenth IEEE European test symposium, pp 69–74. <https://doi.org/10.1109/ETS.2011.19>
45. Abdessaied N, Amy M, Soeken M, Drechsler R (2016) Technology mapping of reversible circuits to clifford-t quantum circuits. In: 2016 IEEE 46th international symposium on multiple-valued logic (ISMVL), pp 150–155. <https://doi.org/10.1109/ISMVL.2016.33>
46. Amy M, Maslov D, Mosca M, Roetteler M (2013) A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits. *IEEE Trans Comput-Aided Des Integr Circuits Syst* 32(6):818–830. <https://doi.org/10.1109/TCAD.2013.2244643>
47. iNEMI (2015) Roadmap executive summary highlights. International electronics manufacturing initiative
48. Gaur HM, Singh AK, Ghanekar U (2017) Testable design of reversible circuits using parity preserving gates. In: IEEE design and test
49. Gaur HM, Singh AK, Ghanekar U (2015) A review on online testability for reversible logic. *Procedia Comput Sci* 70:384–391
50. Gaur HM, Singh AK, Ghanekar U (2016a) A comprehensive and comparative study on online testability in reversible logic. *Pertanika J Sci Technol* 24(2):245–271
51. Gaur HM, Singh AK, Ghanekar U (2018) Offline testing of reversible logic circuits: an analysis. *Integration* 62:50–67. <https://doi.org/10.1016/j.vlsi.2018.01.004>, <http://www.sciencedirect.com/science/article/pii/S0167926016301341>
52. Gaur HM, Singh AK, Ghanekar U (2018b) In-depth comparative analysis of reversible gates for designing logic circuits. *Procedia Comput Sci* 125:810–817
53. Gaur HM, Singh AK, Ghanekar U (2018d) Reversible circuits with testability using quantum controlled not and swap gates. *Indian J Pure Appl Phys* 56(7):529–532
54. Mahammad SN, Veezhinathan K (2010) Constructing online testable circuits using reversible logic. *IEEE Trans Instrum Meas* 59(1):101–109. <https://doi.org/10.1109/TIM.2009.2022103>
55. Gaur HM, Singh AK, Ghanekar U (2016b) A new dft methodology for k-cnot reversible circuits and its implementation using quantum-dot cellular automata. *Optik - International Journal for Light and Electron Optics* 127(22):10593–10601. <https://doi.org/10.1016/j.ijleo.2016.08.072>, <http://www.sciencedirect.com/science/article/pii/S003040261630941X>
56. Gaur HM, Singh AK, Ghanekar U (2018a) Design for stuck-at faults testability in mct based reversible circuits. *Def Sci J* 68(4):381–387