Complete Test Set Generation for Control Flipping Faults in Reversible Circuits

Mousum Handique, Amrit Prasad, and Hiren Kumar Deva Sarma

Abstract The newly developed fabrication process reduces the size of the chips and increases the speed of a system. Simultaneously, power dissipation is also increased, which causes a major issue in semiconductor devices. Therefore, reversible computation is gaining interest in the low-power circuit design and the fast computation system. Moreover, the mechanism of reversible computation is widely used in quantum computing nanotechnology and also in digital communication systems. These inspire the researchers to give more attention to the reversible circuit. Thus, reversible circuit becomes more alternative than the conventional circuit. Testing these circuits is an essential aspect of ensuring these circuits' high reliability and integrity performance. This paper introduced a new fault model labeled as the control flipping fault model (CFF) in a Multiple-Control Toffoli (MCT)-based reversible circuit. Specifically, we target the positive control flipping faults (PCFFs) under the proposed control flipping fault model. The reported works present a scheme to detect fault to construct a complete test set (CTS), which is capable of detecting all the PCFFs in a reversible circuit. The paper also presents an experimental evaluation in order to show the efficiency of the CTS that covers 100% faults.

Keywords Reversible circuit · Control flipping fault model · Positive control flipping fault · MCT-based circuit

1 Introduction

Reducing energy dissipation and in turn energy efficiency is one of the challenging objectives in recent technologies under development, as the size of chips is becoming smaller and faster, and dissipate more energy in the form of heat. In 1961, Landauer [1] showed that irreversible circuits dissipate at least $kTln2$ Joules of energy because of

M. Handique (✉) · A. Prasad
Assam University, Silchar, Assam 788011, India

H. K. D. Sarma
Sikkim Manipal Institute of Technology, Rangpo, Sikkim 737136, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

H. K. D. Sarma et al. (eds.), *Contemporary Issues in Communication, Cloud and Big Data Analytics*, Lecture Notes in Networks and Systems 281,
https://doi.org/10.1007/978-981-16-4244-9_29

loss of information per bit. k signifies the Boltzmann constant, and T represents the temperature prevailing in the system. In 1973, Charles H. Bennett [2] observed the benefits of reversible computation in terms of information lossless, while reversible computation is executed. More precisely, the reversible logic computation can retain the information when operations are executed, and the system runs in a backward direction. Based on these properties, reversible logic is used as a circuit design alternative. The reversible gates are used to implement the reversible logic operation, and only the linear sequence formation of reversible gates is the structure of reversible circuits. The reversible circuit operations are bijective, and also, there is no direct concept of fan-out and feedback connections in reversible circuit [3].

Fault represents the incorrect state during the computation that leads to the functional error of a system. The presence of a single fault may affect the output performance in every computing device. In testing, fault detection is the first phase that detects all possible faults, and the next phase determines the exact location of faults. With the importance of the reversible circuit, testing is also an important parameter for the integrity performance of the reversible circuits. The physical description of faults is described by the fault models. The fault model is the mathematical description that helps the designer predict the system's erroneous state and efficiently construct the test pattern for evaluating the faults. The various fault models have been introduced in reversible circuits. An elaboration of these fault models may be found in [4].

The test generation process is required to determine the faults in a faulty circuit. A test set may be defined as a set of test sequences. A complete test set (CTS) can detect various possible faults available in a presented circuit [5]. The generation of the test set is straightforward in reversible circuits as compared to the traditional circuit because the reversible circuit maintains high controllability and observability [5].

In this paper, we have presented a proposed fault model labeled as control flipping fault model (CFF), which is more relevant in k-CNOT or Multiple-Control Toffoli (MCT)-based circuit structure. A fault detection scheme in order to generate the CTS to detect various possible positive control flipping faults in the presented reversible circuit has been proposed. At last, the experimental results are demonstrated based on the performance evaluation of MCT or k-CNOT-based reversible circuits to verify the proposed fault detection scheme.

The paper has been organized as mentioned below. Section 2 includes the introduction of logic function and gates in the reversible circuit, MCT, or k-CNOT-based circuits. This section also discussed structural fault models that are applied in reversible circuits and covers some of the prior works relevant to the proposed work. The description of the newly introduced fault model and the CTS generation process for detecting PCFFs is explained in Sect. 3. The experimental results based on the PCFFs and concluding remarks are presented in Sects. 4 and 5, respectively.

2 Preliminaries

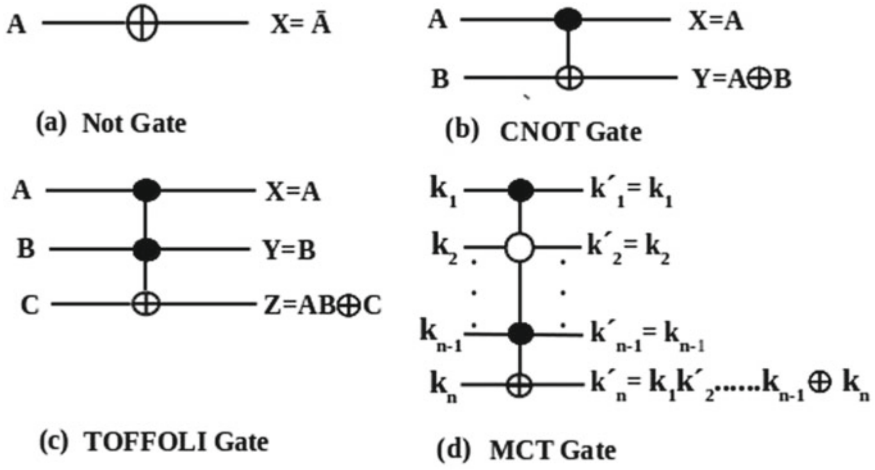
2.1 Logic Function and Gates in Reversible Circuits

Reversible function $f : \mathbb{B}^n \Rightarrow \mathbb{B}^m$ is bijective in nature (i.e., one-to-one and onto mapping), and it maintains the equal inputs (n) and outputs (m) where each possible output vector can uniquely retrievable an input vector. Therefore, the reversible logic function can uniquely determine the inputs and outputs from each other.

Reversible logic operations are used to implement the reversible logic function, and these operations are considered as gate operations to construct the circuit. Each of the reversible gates has an n number of inputs that establish one-to-one mapping with the n number of outputs. In general, NOT gate, FEYNMAN or Controlled-NOT (CNOT) gate [6], TOFFOLI gate [7], and Multiple-Control TOFFOLI gate or k-CNOT gate [7] are the classical gates, and these gates are most commonly used to design the reversible circuits. The operation of the NOT gate in a reversible circuit is similar to a conventional circuit's operation. NOT gate constructed with a single input and output and input A simply inverts the output $X{=}\overline{A}$ as depicted in Fig. 1a. CNOT gate constructs with 2-input × 2-output lines. The first input A is connected to the positive control connection (•), and the output remains the same as it input $X{=}A$. The second input B is connected to the target connection (⊕) and the output $Y{=}A \oplus B$. The input line B inverts if the logic value 1 is set to the input line A. The symbolic representation and gate operations are illustrated in Fig. 1b. TOFFOLI gate consists of 3-input and 3-output. The control lines' output X and Y are the same as the input of the control lines A and B, respectively. The output of the target connection Z inverts the input of the target connection C if both control input lines A and B are set as logic value 1, shown in Fig. 1c. The MCT gate consists of k number of control lines $k_1, k_2, \ldots, k_{n-1}$ where some of the control connection may be positive (•)/negative (○) and one target connection k_n as demonstrated in Fig. 1d. In MCT gate, the target connection's output is only inverted if the logic value is 1/0 for the positive/negative control connections.

2.2 Reversible Circuits

In the reversible circuit, the gates are arranged in a linear cascade manner [8], and the structure of the reversible circuit is not directly allowed any fan-out and feedback connection [3]. Figure 2 illustrates the MCT or k-CNOT-based reversible circuit. Here, the initial input $\langle 0\ 1\ 0\ 1 \rangle$ propagates to primary output $\langle 1\ 1\ 1\ 1 \rangle$. The output vector $\langle 1\ 1\ 0\ 1 \rangle$ is generated by the gate G_1 lies between the levels L_0 and L_1 after applying the initial input $\langle 0\ 1\ 0\ 1 \rangle$ in level L_0. If any intermediate level of the gates generates other test vector, then the primary output vector also changes instead of $\langle 1\ 1\ 1\ 1 \rangle$. Therefore, each reversible gate produces a unique test vector of the circuit.



(a) Not Gate

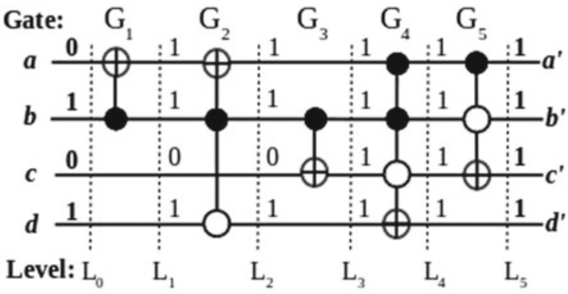
(b) CNOT Gate

(c) TOFFOLI Gate

(d) MCT Gate

Fig. 1 Classical reversible gate

Fig. 2 Illustration of MCT circuit of composing various *k*-CNOT gates



2.3 Fault Models in Reversible Circuits

Description of the various levels of abstraction at which the fault model can be grouped. In other words, fault models can be described at various levels of abstraction in the circuit design hierarchy. These levels of abstraction in circuits are behavioral, functional, structural, and geometrical [9]. The proposed works concentrate only on the fault models that describe the structural level of abstraction in circuits. There are some fault models such as stuck-at [5] and bridging [10] fault models are common to both conventional and reversible circuit testing. Specifically, missing gate [11, 12], partial missing gate [12], and the crosspoint [13] fault models are applied only in reversible circuits that describe the gate level faults.

2.4 Related Work

Several existing works on fault detection approaches of reversible circuits testing have been described in the literature. A deterministic test generation algorithm is proposed to construct the CTS in [5] to detect the stuck-at faults, and the construction of CTS is formulated using ILP. The work in [10] discussed a test generation problem for testing the bridging faults that used in the MCT circuit. The proposed approach showed the generated CTS cardinality is $(d \log_2 n)$ for $n \times n$ reversible circuit with levels d. The authors in [14] have proposed an exact ATPG algorithm with the concept of a set cover method for generating the minimal test set for detecting the bridging faults in a reversible circuit designed with Toffoli, Fredkin and Peres gates. The authors in [15] developed a testing scheme capable of generating the lesser number of test vectors for identifying the single and partial missing gate faults in mixed control MCT-based circuits. The work in [16] presented a fault detection and fault localization scheme for newly developed faults, namely as gate appearance and control appearance in MCT-based circuit. This work showed that only one test vector is needed to detect the gate appearance faults. For evaluating the control appearance fault, this scheme is required n test vectors with n-input lines available in the circuit.

In this literature review, we observed that several approximate and heuristic testing approaches are being proposed to identify the faults in reversible circuits. However, the physical implementation technologies for reversible circuits are still in progress. In this work, we proposed and investigated a control flipping fault model (CFF) based on the MCT-based circuit design structure. This work also presented the complete test generation scheme for evaluating the positive control flipping faults under the proposed fault model.

3 Proposed Model

In this section, the proposed control flipping fault model (CFF) has been introduced. Here, the control flipping fault model is considered as a permanent fault due to incorrect interconnection in designing can be prescribed. Generally, controls are of two types positive (●) and negative (○) control in MCT or k-CNOT-based circuit structure. Flipping of any control to another type can be termed as control flipping faults. Based on the k-CNOT circuit structure, CFF can be divided into three types: (i) Positive control flipping fault (PCFF), (ii) Negative control flipping fault (NCFF), and (iii) Mixed control flipping fault (MCFF). In this work, we specifically target only PCFF for constructing the CTS for all possible PCFFs.

3.1 Positive Control Flipping Fault

Positive control flipping is a fault that occurs in the control flipping fault model. Due to the designing error, when a positive control(s) is flipped to the negative control, such erroneous flipping type is called positive control flipping fault (PCFF). In the k-CNOT circuit, if the PCFF occurrence is involved with only one control, then it is termed as a single positive control flipping fault (SPCFF) as illustrates in Fig. 3a. If the PCFF is involved with two or more control connections, then it is termed multiple positive control flipping fault (MPCFF), which is shown in Fig. 3b.

3.2 Detection Logic for Positive Control Flipping Fault

In PCFF, one or more positive control is flipping to the negative control. The flipped control connection directly affects the target connection. Therefore, the detection logic of PCFF is performed such that the logic value of the target connection will be change after the control flipping. Here, unflipped control connection(s) is set with the logic value 1, and the logic value 1 or 0 (don't care) is applied to all other the affected flipped control connection(s), unconnected line(s), and the target connection. The proposed work applies the logic value 1 to all the flipped and unflipped connection(s) and unconnected connection(s) and the target connection.

Example 1 Figure 4a shows the fault-free $ham3\_tc$ benchmark reversible circuit. Here, we demonstrated the effect of SPCFF and MPCFF as a comparison with the fault-free circuit. Initially, the test vector $\langle 0\ 1\ 1\rangle$ is applied to the gate G_1 in level L_0, and it propagates to the gate G_5 at level L_5. The primary output vector would be $\langle 1\ 0\ 0\rangle$ in the fault-free circuit. In Fig. 4b, SPCFF occurs at control connection 'b' in gate G_1, and primary output is effected in the circuit due to the presence of the SPCFF. If we apply the same initial test vector $\langle 0\ 1\ 1\rangle$ at level L_0 to the circuit, then primary output gate G_5 in level L_5 generates the test vector $\langle 0\ 1\ 1\rangle$, which is faulty output of circuit. MPCFF occurs at both line 'b' and 'c' in gate G_1 which is depicted in Fig. 4c. The presence of MPCFF effects the primary output gate G_5 in level L_5 that produces the faulty test vector $\langle 0\ 1\ 1\rangle$.

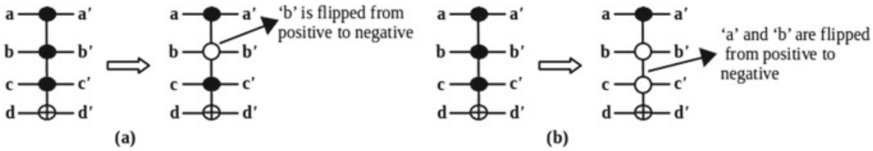

a
b
c
d
a'
b'
c'
d'
'b' is flipped from positive to negative
'a' and 'b' are flipped from positive to negative
(a)
(b)

Fig. 3 **a** SPCFF at line 'a,' **b** MPCFF at line 'b' and 'c'

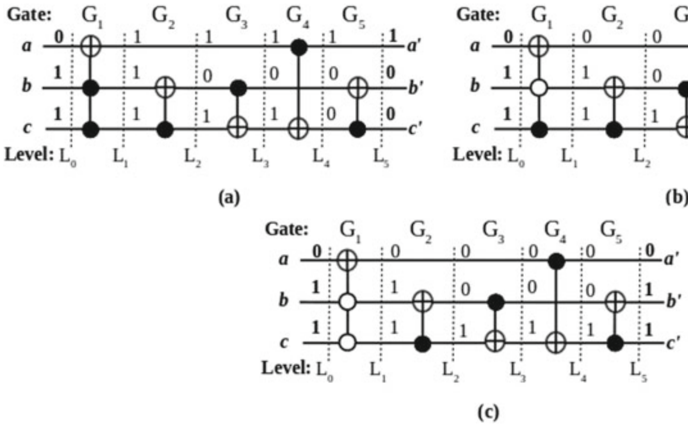


Fig. 4 $ham3\_tc$ reversible circuit: **a** fault-free circuit, **b** SPCFF at line ‘b’ in gate G_1, **c** MPCFF at line ‘b’ and ‘c’ in gate G_1

3.3 Generation of Complete Test Set for PCFF

To obtain the CTS in a presented MCT-based circuit, we consider the concept of fault detection logic for the PCFFs. Here, we consider the logic value 1 that is assigning to all the signal lines of the circuit, and this test sequence applies to each gate G_i. The possible test vector of the CTS retrieves from the initial input level by the backpropagation. Algorithm 1 presented the construction of CTS.

Example 2 The CTS generation algorithm is illustrated using $ham3\_tc$ benchmark circuit in Fig. 5. Here, gate G_j is denoted at the jth gate in the circuit, where $1 \leq j \leq N$, where N denotes the number of gates. The array $OutputList[\ ]$ initially empty and the array $TP[\ ]$ stores as $TP[000, 001, 010, 011, 100, 101, 110, 111]$ for n=3. The values of $RevReadGate[\ ]$ stores {‘c, b’, ‘a, c’, ‘b, c’, ‘c, b’, ‘b, c, a’} all information of controls and target connection that are present of gate G_j in reverse order. Let $RevReadGate[3]$ stores {‘b, c’} information of gate G_3. If we apply the $TP[6]$=$\langle 1\ 1\ 0 \rangle$ from primary output to the gate G_3 level, then $TP[7]$=$\langle 1\ 1\ 1 \rangle$ is generated. Now, the test pattern $TP[7]$=$\langle 1\ 1\ 1 \rangle$ propagates to the initial input level, and the $OutputList$ produces $TP[7]$=$\langle 1\ 1\ 0 \rangle$. Therefore, the test vector $\langle 1\ 1\ 0 \rangle$ is capable of detecting all PCFFs at gate G_3. In similar way, $OutputList$ is $\langle 0\ 1\ 1 \rangle$, $\langle 1\ 0\ 1 \rangle$, $\langle 1\ 0\ 1 \rangle$, and $\langle 1\ 0\ 0 \rangle$ for the gate G_1, G_2, G_4, and G_5, respectively. Finally, complete test set for the $ham3\_tc$ circuit is {011, 101, 110, 100} for all PCFFs.

Lemma 1 *The generated CTS covers all the possible PCFFs in MCT or k-CNOT-based circuit.*

Proof As per the operation of k-CNOT gate, the target connection output only inverts when all the controls are assigning the logic value 1, whereas the logic value 1 or

Algorithm 1: CTS generation for PCFF

```
Input: Generated tfc file using given circuit structure.
$TP[\,]$: stores binary encoding pattern of $n$ input lines.
$OutputList[\,]$: stores input variables with their corresponding binary value.
$RevReadGate\,[\,]$: stores the target and control information in reverse.
$G_j$: store the control(s) and target variable for each gate at a time.
$L_j$: cardinality of $G_j$.
Output: Complete test set for detecting all PCFF
$OutputList[\,] \leftarrow \varnothing$
for $i \leftarrow 0$ to $len(TP) - 1$ do
    $OutputList \leftarrow mapping(TP[i])$
    Flag=0
    for $j \leftarrow 0$ to $len(RevReadGate) - 1$ do
        $G_j \leftarrow RevReadGate[j]$
        $L_j \leftarrow len(G_j)$
        if All elements in $OutputList$.values is 1 AND $L_j >= 2$ then
            Flag=1
        if $L_j$==1 then
            $OutputList[G_j[L_{j-1}]]$ = NOT $OutputList[G_j[L_{j-1}]]$
        else
            Temp = $OutputList[G_j[0]]$
            for $k \leftarrow 1$ to $L_{j-2}$ do
                Temp = Temp AND $OutputList[G_j[k]]$
            $OutputList[G_j[L_{j-1}]]$ = $OutputList[G_j[L_{j-1}]]$ EXOR Temp
    if Flag==1 then
        Print($OutputList$.values)
```

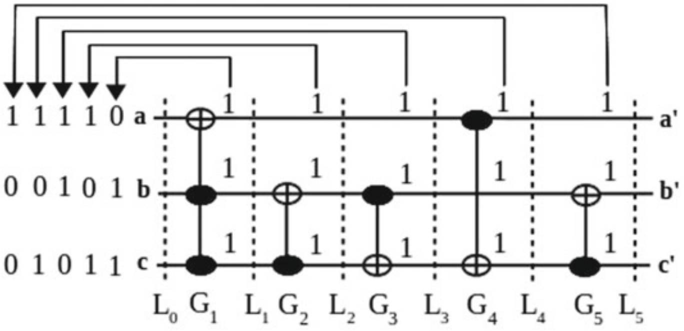


Fig. 5 Demonstration of Algorithm 1 for the *ham*3_*tc* circuit

0 (don't care) applies in target and unconnected lines. However, when one or more positive controls are affected by flipping control, then the logic value of flipping control inverts at the target connection. As a result, the output logic value of the target connection remains the same as the input. Therefore, one binary logic value

changes at the target connection when positive control is flipping. In this proposed method, the logic value 1 is assigning to the lines of each available gate G_j and stores in test pattern TP[i] and the backpropagation of TP[i] toward the input level to form the complete test set for each k-CNOT gate. Hence, a constructed test set at the initial input level is the CTS for all the PCFFs in an MCT- based circuit.

4 Experimental Results

The presented algorithm for PCFFs has been run on an Intel Pentium (R) CPU-2020 @ 2.40GHZ × 2 system running on Windows 10 (64-bit) with 4 GB RAM and implemented in Python 3.4. Several benchmark circuits [17, 18] are considered for analyzing the proposed CTS generation algorithm. Based on this analysis, the experimental results are shown in Table 1. Here, n and N are denoting as the number of input lines and gates presented in columns 2 and 3, respectively. Column 4 is presenting all possible faults for both SPCFFs and MPCFFs in a presented circuit. The cardinality of the generated CTS to detect the faults is shown in column 5. The last column of Table 1 mentioned the computational time for generating the CTS. From the experimental results, it is observed that in most of the circuits, the computational time increases when the number of gates is increasing as compared to the number of input lines in k-CNOT-based circuits.

Table 1 Experimental results for PCFFs detection and its computation time (s)

Benchmarks circuit	n	N	Total no. of faults	CTS	CPU time (s)
			(SPCFF+MPCFF)	(SPCFF+MPCFF)	
Peres_9	3	2	4	2	0.01562
3_17_14	3	6	9	4	0.01562
3_17_13	3	6	9	4	0.01563
ex-1-166	3	4	5	3	0.0162
fredkin_6	3	3	9	3	0.0155
ham3_tc	3	5	7	4	0.0163
4b15g_1	4	14	28	9	0.0161
4b15g_2	4	15	35	7	0.0158
hwb4-11-21	4	11	17	8	0.0156
hwb4d1	4	17	41	7	0.0176
mspk_4b15g_2	4	15	24	9	0.0156
mspk_hwb4_13	4	13	19	9	0.0165
ham15-70	15	70	278	53	11.325
ham15-109-214	15	109	427	94	14.418
ham15tc1	15	132	2816	73	24.392

5 Conclusion

This paper introduces a new fault model called a control flipping fault model (CFF), and the physical justification of this kind of fault model is considered a designing fault. In this method, we are efficiently extracting the test vectors for evaluating all single positive control flipping fault (SPCFF) and multiple positive control flipping fault (MPCFF) under the control flipping fault model. Our work may be extended to detect other faults like negative control and mixed control flipping faults in MCT-based circuits as future work. Moreover, this work may establish the correlation of other existing fault models of reversible circuits.

References

1. Landauer, R.: Irreversibility and heat generation in the computing process. IBM J. Res. Dev. **5**(8), 183–191 (1961)
2. Bennett, C.H.: Logical reversibility of computation. IBM J. Res. Dev. **17**(6), 525–532 (1973)
3. Nielson, M.A., Chuang, I.L: Quantum Computation and Quantum Information. Monograph Collection (Matt - Pseudo) (2000)
4. Rice, J.: An overview of fault models and testing approaches for reversible logic. In: 2013 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), pp. 125–130. IEEE (2013)
5. Patel, K.N., Hayes, J.P., Markov, I.L.: Fault testing for reversible circuits. IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **23**(8), 1220–1230 (2004)
6. Feynman, R.P.: Quantum mechanical computers. Found. Phys. **16**(6), 507–531 (1986)
7. Toffoli, T.: Reversible computing. In: International Colloquium on Automata, Language, and programming, pp. 632–644. Springer, Berlin (1980)
8. Maslov, D.: Reversible logic synthesis. University of New Brunswick, Ph.D. diss. (2003)
9. Jha, N.K., Gupta, S.: Testing of Digital Systems. Cambridge University Press (2003)
10. Rahaman, H., Kole, D.K., Das, D.K., Bhattacharya, B.B.: Optimum test set for bridging fault detection in reversible circuits. In: Asian Test Symposium, ATS'07. 16th, pp. 125–128. IEEE (2007)
11. Hayes, J.P., Polian, I., Becker, B.: Testing for missing-gate faults in reversible circuits. In: 13th Asian Test Symposium, vol. 2004, pp. 100–105. IEEE (2004)
12. Polian, I., Fiehn, T., Becker, B., Hays, J.P.: A family of logical fault models for reversible circuits. In: 14th Asian Test Symposium (ATS'05), vol. 2005, pp. 422–427. IEEE (2005)
13. Zhong, J., Muzio, C.J.: Analyzing fault models for reversible logic circuits. In: 2006 IEEE International Conference on Evolutionary Computation, pp. 2422–2427. IEEE (2006)
14. Nagamani, A., Abhishek, B., Agrawal, V.K.: Deterministic approach for bridging fault detection in peres-fredkin and toffoli based reversible circuits. In: 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCC), pp. 1–6 (2015)
15. Mondal, B., Bandyopadhyay, C., Rahaman, H.: A testing scheme for mixed-control based reversible circuits. In: 6th International Symposium on Embedded Computing and System Design (ISED), pp. 96–100. IEEE (2016)
16. Mondal, B., Bandyopadhyay, C., Rahaman, H.: Detection and localization of appearance faults in reversible circuits. In: 7th International Symposium on Embedded Computing and System Design (ISED), pp. 1–5. IEEE (2017)

17. Maslov, D.: Reversible logic synthesis benchmark page. http://webhome.cs.uvic.ca/dmaslov/ (2015)
18. Wille, R., Große, D., Teuber, L., Dueck, G.W., Drechsler, R.: Revlib: an online resource for reversible functions and reversible circuits. In: Proceedings of 38th International Symposium on Multiple Valued Logic (ismvl 2008), pp. 220–225