Efficient Design of Check Circuit to detect Multiple Cell Errors in Reversible Logic Synthesis

Sajib Kumar Mitra, Shahed Anwar and Ahsan Raja Chowdhury
Department of Computer Science and Engineering
University of Dhaka
Dhaka- 1000, Bangladesh
{sajibmitra.csedu, shahedanwar01}@yahoo.com and farhan717@univdhaka.edu

***Abstract*—Online Testability is used to detect bit error of Reversible Circuit at runtime using Check circuit to propagate all errors into single line. In this paper, we propose an improved design of check circuit, Modified Test Cell (MTC) can detect error in short time and also able to identify affected cell(s) simultaneously. We also present a cost effective design of Online Testable Ripple Carry Adder (RCA) with proposed MTC. Proposed design of check circuit is able to enhance the reusability of circuit by replacing corresponding faulty cell(s).**

I. Introduction

Modern Computing is based on the nanotechnologies in every part of the electronics. But the underlaying foundation is irreversible which consumes more power and have no reliable system to detect the faults. Irreversible erasures of a bit in a system leads to generation of energy in the form of heat [1] and finally *kT**log2 joules heat is generated to reload information instead of single bit, where *k* is the Boltzmann's constant and *T* is absolute area temperature. Reversibility recovers bit loss by using unique mapping between input and output vectors. Alternatively reversibility disposes all the undesired intermediate results by retracing the step in backward order and energy dissipation per logical step is less than *kT* [2]. Reversible computation in a system can be performed if the system is composed of reversible gates, also has two major limitations:

I. Maximum fan-out always equal to one
II. Feed back is strictly restricted

Online testability uses only two testing output bits to verify alteration of bit at output lines of testable gate [3], [4] and can be used in Embedded Technologies.

In this paper, we have proposed the modified design of Test Cell (TC) [4] called Modified Test Cell (MTC) where the design mechanism is very simple but efficient to specify Multiple Faulty Cells (MFC) as well as the error in whole circuit.

Rest of the paper is organized as follows: Section II discusses about Reversible Logic about Online Testability. Section III provides an brief overview of existing Cell testing Logic. Proposed logic design of MTC is introduced in Section IV and Section V finishes with concluding remarks.

II. Background Study

A. Reversible Logic

Reversible logic is the unique mapping between input and output bit patterns where unit logic entities are represented as Gate.

Definition 1: A **Reversible Gate** is one that has equal number of input-output lines and preserves a unique mapping over any value of input to resultant output.

Let, I_v= (I_0, I_1, …, I_{n-1}) and O_v= (O_0, O_1, …, O_{n-1}) are input and output vectors respectively, so the relation is as follows:

$$I_v \leftrightarrow O_v$$

For example, Fig. 1 shows the input-output mapping of Feynman Gate (FG) [5] where input vector, I_v= (A, B) and corresponding output vector is O_v= (A, $A \oplus B$).

Many reversible gates are proposed in few decades. Most popular reversible gates are Feynman Double Gate (F2G) [6], Toffoli Gate (TG) [7], Peres Gate (PG) [8] and Fredkin Gate (FRG) [9] etc.

B. Runtime Error Detection

Online Testability is the shortcut method to detect any error in reversible circuit at runtime where the basic components are called Testable Reversible Cell (TRC) [4] or Cascading Block (CB) [3], [10], [11].

Definition 2: Any *n*-dimensional reversible gate can be online testable gate by adding an extra input and output line, which is also called **Deducible Reversible Gate (DRG)** [4] as shown in Fig. 2.a.

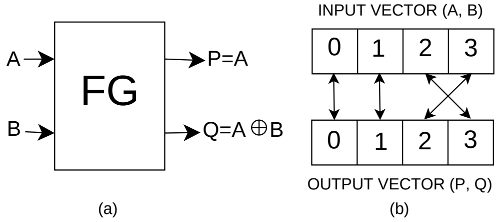

Fig. 1. (a) Feynman Gate (FG) (b) Input output mapping of FG

ISBN: 978-1-6123-3002-0

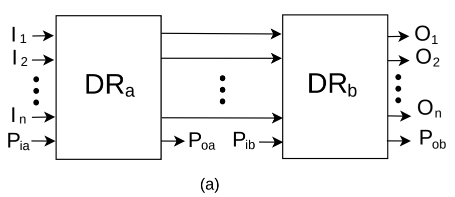

Fig. 2. (a) Cascading attachment of two DRGs (b) Testable Reversible Cell

Any (n+1) dimensional DRG with input-output vectors I_v and O_v can be online testable if it preserves the following property:

$$O_{n+1} = I_{n+1} \oplus O_0 \oplus O_1 \oplus O_2 \oplus \ldots \oplus O_n$$

But testable DRG is not enough to detect error within itself rather it needs another testable part to check bit errors occurrence. Two different bits are needed to compare that output is either correct or wrong.

The actual idea behind is: Any DRG (DRa) will always produce correct symptom by its testing output bit (P_{oa}) and another DRG (DRb) will generate another testing bit (P_{ob}) which will be affected by the changes of any output bits of prior DRG (DRa). Finally, the cascading attachment of two DRGs produces Testable Reversible Cell (TRC) which is fully testable at runtime (shown in Fig. 2).

III. Existing Check Circuit

Primarily, there is an extra Check circuit used as a co-operator of online testable circuit to detect error of whole circuit by using minimum number of testing bits because every Testable Reversible Cell produces their errors individually and is not possible to trace them without any additional check circuit.

A. Two Pair Rail Check Circuit

Definition 3: Rail check (RC) circuit is used to propagate errors of any Testable Reversible Cell (TRC) and reports the occurrence of error by using two testing bits of the overall circuit [10].

Rail Check circuit uses four testing output bits (x_0, y_0, x_1 and y_1) of two TRCs and produces two testing output bits (e_1, e_2) to detect error. Fig. 3.a shows a simple Rail Check circuit which is constructed by only using R3 [10], [11]. The error checking functions of the two pair rail check are as follows:

$$e_1 = x_0y_1 + y_0x_1$$
$$e_2 = x_0x_1 + y_0y_1$$

where the complementary output e_1 and e_2 ensures that no error is occurred in corresponding blocks only when the testing inputs of corresponding TRCs are also in complemented form.

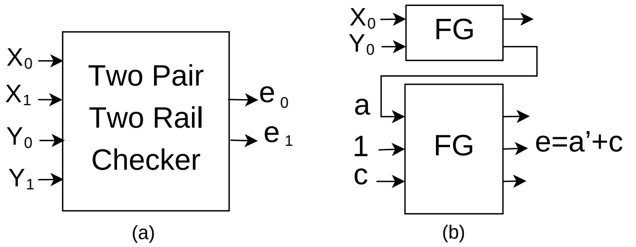

Fig. 3. (a) Block diagram of Rail Check and (b) Improved Rail Check

B. Improved Rail Check (IRC)

Definition 4: Improved Rail Check (IRC) is the superior version Rail Check consists of FG and FRG that detects and propagates error by single output which was proposed in [12].

IRC provides better performance over the previous Rail Check Circuit [12]. Fig. 3.b shows the organization of IRC which uses FG and FRG gates and acts on single TRC but propagate the error of previous IRC to the next with the self status.

The Rail Check circuit [10], [11] uses huge number of gates and garbage and also contains feedback. On the other hand, IRC [12] having no feedback and can handle single or multiple errors efficiently detection properties.

C. Test Cell

Definition 8: Test Cell (TC) is a straightforward check circuit (shown in Fig. 4) which can check error of N-TRC(s) with (2N+1)×(2N+1) dimension [4].

Test Cell maps the input vector, I_v= (P_{oa1}, P_{ob1}, P_{oa2}, P_{ob2}, $\ldots$, P_{oaN}, P_{obN}, e) to output vector, O_v= (P_{oa1}, P_{ob1}, P_{oa2}, P_{ob2}, $\ldots$, P_{oaN}, P_{obN}, T) where T= $(P_{oa1} \oplus P_{ob1})(P_{oa2} \oplus P_{ob2}) \ldots (P_{oaN} \oplus P_{obN}) \oplus e$.

Huge number of IRCs circuitry is required for multiple errors detection of testable circuitry. For instance, N-TRCs need N-IRCs to detect error individually.

Test Cell [4] pulls through the density limitation of IRC by mapping circuit nodes into ports. But TC is not able to detect MFC and it works based on identical value logic.

In our proposed design we have used complementary logic i.e. 0, 1 as testing input and merge MFC detection logic. The following section has described in detail of proposed design.

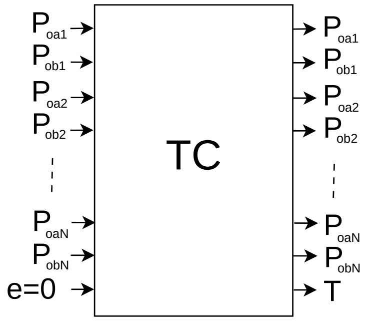

Fig. 4. (2N+1)×(2N+1) Test Cell

IV. Proposed Design

A. Modified Test Cell Design

The proposed design of Modified Test Cell is based on complementary logic (testing inputs of any TRCs use complement value which is not identical).

Fig. 5 shows the basic construction of proposed Modified Test Cell (MTC) where Input vector, I_v =(*A*, *B*, *C*) and Output vector, O_v = (*A*, *A* $\oplus$ *B*, *A* $\oplus$ *B* $\oplus$ *C*).

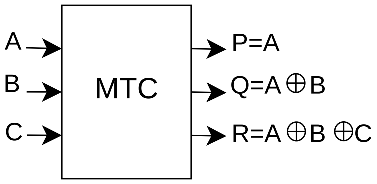

Fig. 5. Proposed Modified Test Cell (MTC)

The unique mapping of MTC can be expressed as Fig. 6 where binary values of input and output vectors are expressed as decimal numbers (0-7) and the arrows represent corresponding unique mapping.

By setting input C to HIGH, MTC produces result in complement form, ($\overline{A \oplus B}$) and a non-complement form, (*A* $\oplus$ *B*) which are represented by *e* and *T* respectively (as shown in Fig. 7). Actually *e* is used as error propagator and *T* as MFC detection.

The 3x3 MTC actually uses the testing output bits come from TRC as input and produces *e* as single testing output bit respect to that TRC. We make an initiative to extended the design of MTC along with better realization of fault occurrences. We have proposed one theorem and two algorithms for MTC to handle multiple TRCs and MFC detection.

Theorem 1: *N*-Testable Reversible Cell (TRC) can be realized by using (2*N*+1)×(2*N*+1) dimensional MTC.

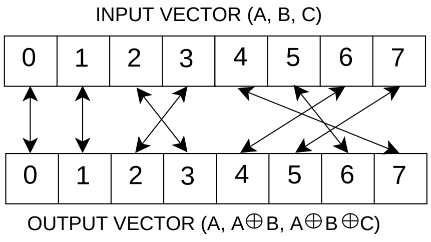

Fig. 6. Input and output mapping realization of Basic MTC

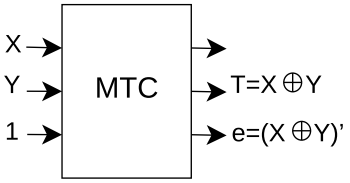

Fig. 7. Error Detection By using Single MTC

Proof: Lets, TRC generates two testing outputs P_{oa} and P_{ob} come from two different DRGs. Testing inputs (P_{ia}, P_{ib}) of the TRC use complementary value. So the testing result of single TRC can be defined as,

$$e = (\overline{P_{oa} \oplus P_{ob}})$$

MTC generates error (*e*) by setting *C*= 1 where the dimension of MTC is 3 and produces Single Cell Testing result, (*T*) where,

$$T = (P_{oa} \oplus P_{ob})$$

For *N*-TRCs, the total number of testing outputs is 2*N* and the error (*e*) will get the following mapping:

$$e = (\overline{P_{oa1} \oplus P_{ob1}}) + (\overline{P_{oa2} \oplus P_{ob2}}) + \cdots + (\overline{P_{oaN} \oplus P_{obN}})$$

By setting (2*N*+1)th input of (2*N*+1)×(2*N*+1) dimensional MTC to HIGH, mapping can be shown as :

$$\begin{aligned} I_v &= (P_{oa1}, P_{ob1}, P_{oa2}, P_{ob2}, \cdots, P_{oaN}, P_{obN}, 1) \\ O_v &= (P_{oa1}, T_1, P_{oa2}, T_2, \cdots, P_{oaN}, T_N, e) \end{aligned}$$

where,

$$\begin{aligned} T_i &= (P_{oai} \oplus P_{obi}) \;\; [\text{Here}, i = 1, 2, 3, \cdots, N] \\ e &= (\overline{T_1}) + (\overline{T_2}) + \cdots + (\overline{T_N}) \end{aligned}$$

So, (2*N*+1) dimensional MTC can detect error (*e*) produced by *N*-TRC(s).

Algorithm 1: Cell_Error_Detection

```
Input: Testing bit(s) ($P_{oai}$, $P_{obi}$) from i$^{th}$-TRC(s) where $i$= 1, 2, 3, ..., $N$.
Output: Error status ($e$).
Begin
Step 1:
        Set both testing inputs of all $N$-TRC(s) as 0, 1
Step 2:
        Connect N-TRC(s) testing outputs ($P_{oa1}$, $P_{ob1}$, $P_{oa2}$, $P_{ob2}$, ..., $P_{oaN}$, $P_{obN}$) as first 2$N$ inputs of MTC but set the last input HIGH and corresponding error output (e) will be as follows:

$e = (\overline{P_{oa1} \oplus P_{ob1}}) + (\overline{P_{oa2} \oplus P_{ob2}}) + \cdots + (\overline{P_{oaN} \oplus P_{obN}})$
Step 3:
        If e =1 then
                Print ”Error”.
        Else
                Print ”No Error”.
        End If
End
```

Fig. 8 shows the mapping of (2*N*+1)x(2*N*+1) MTC where (*N*+1)th output can be expressed as follows when corresponding input is HIGH.

$$e = (P_{oa1} \oplus P_{ob1})(P_{oa2} \oplus P_{ob2}) \cdots (P_{oaN} \oplus P_{obN}) \oplus 1$$

B. Multiple Faulty Cell (MFC) Detection

MTC enhances the usability of embedded circuit where only faulty cell(s) will be replaced instead of the whole system. Proposed design of MTC is able to specify the error Cell(s) along with minimum number of garbage outputs.

```
Algorithm 2: Multiple_Faulty_Cell_Detection
Input: Testing bit(s) $P_{oai}$, $P_{obi}$ from i$^{th}$-TRC(s) where i = 1, 2, 3, ..., N.
Output: Testing bits, $T_1, T_2, T_3, \ldots, T_N$
Begin
Step 1:
        Set both testing inputs of N-TRC(s) as 0, 1
Step 2:
        Connect N-TRC(s) testing outputs ($P_{oa1}$, $P_{ob1}$, $P_{oa2}$, $P_{ob2}$, ..., $P_{oaN}$, $P_{obN}$) as first $2N$ inputs of MTC but set the last input HIGH and corresponding error output is,
$e = (\overline{P_{oa1} \oplus P_{ob1}}) + (\overline{P_{oa2} \oplus P_{ob2}}) + \cdots + (\overline{P_{oaN} \oplus P_{obN}})$
    And the corresponding cells test bit are:
        $T_1 = (P_{oa1} \oplus P_{ob1})$
        $T_2 = (P_{oa2} \oplus P_{ob2})$
        $\vdots$
        $T_N = (P_{oaN} \oplus P_{obN})$
Step 3:
        If = 1 then
            For $i$=1 to N do
                If $T_i$= LOW then
                    Print " i$^{th}$ TRC is Faulty".
                End If
            End Loop
        Else
            Print "No Error".
        End If
End
```

For Instance, Ripple Carry Adder (RCA) can be realized by using single MTC where the dimension of MTC depends on the number of TRCs. Any ($N \times N$) Reversible can be Testable Reversible gate by adding two inputs and two outputs testing bits [4]. The value of testing inputs can be identical or complement.

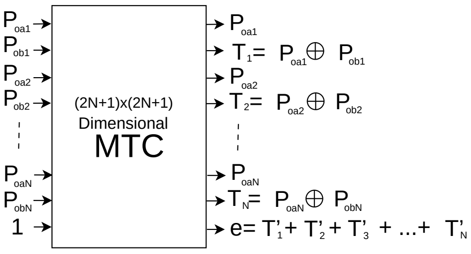
P_{oa1}
P_{ob1}
P_{oa2}
P_{ob2}
P_{oaN}
P_{obN}
1
(2N+1)x(2N+1)
Dimensional
MTC
P_{oa1}
$T_1 = P_{oa1} \oplus P_{ob1}$
P_{oa2}
$T_2 = P_{oa2} \oplus P_{ob2}$
P_{oaN}
$T_N = P_{oaN} \oplus P_{obN}$
$e = T'_1 + T'_2 + T'_3 + \ldots + T'_N$

Fig. 8. (2N+1)×(2N+1) Dimensional MTC

Reversible RCA can be design as online testable by adapting all gates into TRC(s) and finally, we can use proposed MTC to detect all types of subset of errors (shown in Fig. 9).

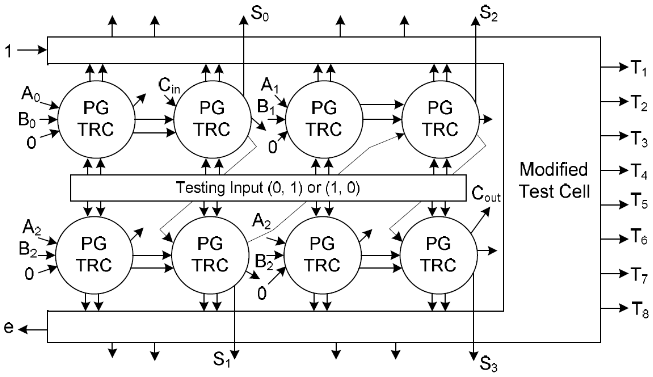

Fig. 9. Online Testable Ripple Carry Adder

Table I shows how faulty cells of RCA can be detected using Modified Test Cell (MTC).

TABLE I
MULTIPLE FAULTY CELLS DETECTION BY USING PROPOSED MTC

Test bits								Faulty TRC(s)
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	
1	1	1	1	1	1	1	1	No Error
0	1	0	1	1	1	1	0	1^{st}, 3^{rd}, 8^{th}
1	1	1	1	0	0	1	0	5^{th}, 6^{th}, 8^{th}

Table II shows the caparison between proposed and existing of all check circuits respect to the design of Online Testable RCA.

TABLE II
COMPARISON BETWEEN PROPOSED AND EXISTING [4], [10], [12]

Online Testable 4-bit RCA	Total Gates	Garbage
Proposed Design	1	8
Existing [4]†	1	16
Existing [10]†	42	56
Existing [12]	8	16

† Unable to detect multiple cell errors

V. CONCLUSION

Online Testability gains popularity day by day for detecting error along with minimum cost to provide fault free and noiseless Reversible circuit. We have explained how we can achieve the goal of Online Testing in most of the prominent field like Nanotechnology by increasing the reusability of circuit. Proposed design of Modified Test cell (MTC) has minimum cost enhances the testing methodology and increases the reusability of circuit. Our future plan is to provide better throughput by attaching Online Correction module with Online Testable circuit.

ISBN: 978-1-6123-3002-0

References

[1] R. Landauer,"*Irreversibility and Heat Generation in the Computational Process*," IBM Journal of Research and Development, Vol 5, pp 183-191, 1961.

[2] C. H. Bennet,"*Logical Reversibility of Computation*," IBM Journal of Research and Development, pp 525-532, 1973.

[3] D. P. Vasudevan, P. K. Lara and J. Parkerson,"*A Novel Approach for Online Testable Reversible Logic Circuit Design*," Proc of the 13th Asian Test Symposium, pp 325-330, 2004.

[4] Sk. N. Mahammad and K. Veezhinathan, "*Constructing Online Testable Circuits Using Reversible Logic*," IEEE Trans. Instrum. Meas, vol- 59, No. 1, 2010.

[5] R. Feynman, "*Quantum mechanical computers*," Opt. News, Vol- 11, pp 11-20, 1985.

[6] B. Parhami, "*Fault Tolerant Reversible Circuits*," In Proc. of 40th Asimolar Conf. Signals, Systems, and Computers, Pacific Grove, CA, pp 1726-1729, 2006.

[7] T. Toffoli, "*Reversible Computing*," Tech memo MIT/LCS/TM-151. MIT Lab for Comp. Sci, 1980.

[8] A. Peres, "*Reversible logic and quantum computers*," Physical Review A 32, pp 3266-3276, 1985.

[9] E. Fredkin and T. Toffoli, "*Conservative logic*," Int'l J. Theoretical Physics, Vol- 21, pp 219-253, 1982.

[10] D. P. Vasudevan, P. K. Lara and J. Parkerson, "*Reversible Logic Design with Online Reliability*," IEEE Transaction on Instrumentation and Measurement, Vol-55, No- 2, pp 406-414, 2006.

[11] D. P. Vasudevan, P. K. Lara and J. Parkerson,"*Online Testable Reversible Circuit Design using NAND blocks*," Proc of Symposium on Defect and Fault Tolerance, Oct, pp 324-331, 2004.

[12] M. Hasan,A. T. Islam and A. R. Chowdhury, "*Design and analysis of online testability of reversible sequential circuits*," International Conference on Computer and Information Technology. Independent University Bangladesh, 21th-23th December, pp 180-185, 2009.