An Efficient Construction of Online Testable Circuits using Reversible Logic Gates

K. Ganesh Kumar[1] R. Arvind[2] S. Gowtham[3] Y. Azain Abdul Kadhar[4] P. Dass[5]
[1,2,3,4,5]Department of Electronics and Communication Engineering
[1,2,3,4,5]Saveetha School of engineering, Saveetha University, Chennai, India

Abstract— The vital for many safety critical applications is the testable fault tolerant system. Due to its less heat dissipating characteristics, the reversible logic gaining interest in the recent times. Any Boolean logic function can be implemented using reversible gates. The credential part of the paper proposes a technique to convert any reversible logic gate to a testable gate that is also reversible. The resultant reversible testable gate can detect online any single bit errors that include Single Stuck Faults and Single Event Upsets S. Karp et.al. The proposed technique is illustrated using an example that converts a reversible decoder circuit to an online testable reversible decoder circuit.

Key words: Reversible gate, single stuck fault, testable gate.

I. INTRODUCTION

Reversible Logic has gained importance in the recent past. The rapid decrease in the size of the chips has lead to the exponential increase in the transist account per unit area. As a result, the energy dissipation is becoming a major barrier in the evolving nano-computing era. Reversible logic ensures low energy dissipation. An operation is said to be physically reversible if there is no energy to heat conversion and no change in entropy. In reversible logic, the state of the computational device just prior to an operation is uniquely determined by its state just after the operation. In other words, no information about the computational state can ever be lost and hence the reversible logic can be viewed as a deterministic state machine. Computations performed by the current computers are commonly irreversible, even though the physical devices that execute them are fundamentally reversible. At the basic level, however, matter is governed by classical mechanics and quantum mechanics, which are reversible. With computational device technology rapidly approaching the elementary particle level, it has been argued many times that this effect gains in significance to the extent that efficient operation of future computers requires them to be reversible. Hence, reversible logic is gaining grounds. A reversible gate is a logical cell that has the same number of inputs and outputs. Also, the input and output vectors have a one-to-one mapping. Direct fan-outs from the reversible gate are not permitted. Feedbacks from gate outputs to inputs are not allowed. A reversible gate with n-inputs and n-outputs is called a n x n reversible gate. A previous research has been done on testable reversible circuits. Conditions for a complete test set construction were discussed and the problem of finding a minimum test set was formulated as an integer linear program with binary

The technique proposed in this paper can be employed to convert any reversible circuit with arbitrary number of gates to an online testable reversible one and is independent of the type of reversible gate used. The constructed circuit can detect any single bit errors that include single bit stuck-at-fault and single event upset S.Karp et.al . An important advantage of the technique is that the logic design of a reversible circuit remains the same and the reversible circuit need not be redesigned for adding the testability feature to it. Another advantage is that the technique ensures that the garbage generated during the process of conversion to testable reversible circuit is minimized. The proposed technique is illustrated using an example that converts a decoder circuit that is designed by reversible gates to an online testable reversible decoder circuit.

II. DESIGN

A. Construction of online testable circuit

This section describes an algorithmic approach to convert any reversible circuit to an online testable reversible circuit. Given a reversible circuit consisting of reversible gates, the following algorithm converts it into an online testable reversible circuit.

Algorithm

Input: Reversible Circuit C

Output: An online testable reversible circuit C^T

Construct C' by replacing every reversible gate R in C by TRC(R). The parity input bits of TRC(R) are set such that $P_{ia} = P_{ib}$ in the construction of TRC(R). By Lemma 2, C' is reversible. Fig 1

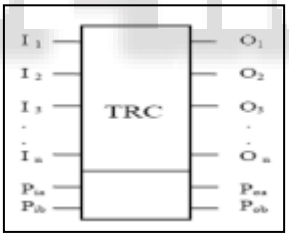

Fig 1: Block diagram of TRC

Let n be the number of reversible gates in C. Construct a (2n+1) x (2n+1) Test Cell (TC)

First 2n inputs are the output parity bits from each of the n testable reversible cell TRC of C' gate.

The last bit of the input, called e, is either set to logic 0 or logic 1.

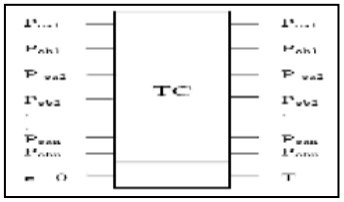

Fig 2 Block diagram of TC

First 2n inputs are transferred to the output without any change.

B. Constructible reversible circuit

1) Theorem:

The cell TC constructed in Algorithm has the following properties:

- It is reversible.
- If there is a single bit error in any TRC in C^T, then, T = 1 provided e =0.
- Function T is implemented with minimum possible garbage.

Illustrated in Fig 2

2) Proof:

We can easily see that [P_{oa1}, P_{ob1}, . . P_{oan}, P_{obn}, 0] maps to [P_{oa1}, P_{ob1}, P_{oan}, P_{obn}, T] and [P_{oa1}, P_{ob1}, . . P_{oan}, P_{obn}, 1] maps to [P_{oa1}, P_{ob1}, . . P_{oan}, P_{obn}, ?T], where T = [((P_{oa1}? P_{ob1}) + (P_{oa2}? P_{ob2}). . . + (P_{oan}? P_{obn})) ?e] Hence, TC is reversible.

P_{ia} = P_{ib} in TRC, from the step 1 of Algorithm 1. If there is a single bit error in any TRC then by Lemma 3, $P_{oa}$is complementary to P_{ob}.Therefore, P_{oa}? P_{ob} = 1. Hence T = 1.

For an n-input k-output function f, the minimum number of garbage bits required to make it reversible is ceil(log M), where M is the maximum number of times an output pattern is repeated D. Maslov (2004). For the function T, $M = 2^{2n} - 2^n$, where n is the number of TRCs in C^T. Therefore, log M = log ($2^{2n} - 2^n$) > log (2^{2n}/2) =(2n - 1). Therefore, ceil (log M) = 2n.

The garbage bits that are generated for any circuit that is implemented using the cascaded block of R1 and R2 in D.P.Vasudevan et.al (2004), will be certainly greater than or equal to the circuit implemented using TRC(R). For example a two input AND gate implemented using R1 gate generates extra 2 garbage bits compared to TRC (Toffoli gate). The two pair rail checker used in D.P.Vasudevan et.al (2004) to detect errors is constructed from 6 R3 gates and produces garbage of 8 bits. For a reversible circuit with 2n testable reversible gates the technique proposed in D. P. Vasudevan et.al (2004) generates garbage of 8n bits where as the Test Cell, TC, generates garbage of 2n bits. This paper proposes a hierarchical construction for online testable reversible circuits. Consider a reversible circuit C that is the integration of the individual reversible modules C_i ∀ $_i$ = 1, 2, k. C_i is made online testable by applying Algorithm.

A Multi Modular Testable Cell MMTC is used to detect errors in the integrated circuit C.

MMTC is defined as follows:

Inputs: T_1, T_2 . . . Tn and e, where e can be either 0 or 1.

Outputs: T_1, T_2 . . . Tn and MMT = (T_1 + T_2 + . . + T_n) ⊕ e.

3) Theorem:

The cell MMTC has the following properties:

- It is reversible.
- MMTC detects any single bit error in C_1, C_2, . . . C_k, where C_i is a online testable reversible module ∀ $_i$.
- Function MMT is implemented with minimum possible garbage.

4) Proof:

- We can easily see that [T_1, T_2, . .T_k, 0] maps to [T_1, T_2, . . . T_k, MMT] and [T_1, T_2, . . . T_k, 1] maps to [T_1, T_2, . . T_k, ~MMT] and MMT = (T_1 + T_2 + T_3 . . . + T_k) ⊕ e Hence, MMTC is reversible.
- T_i bit is the test bit from the TC of module C_i. The multi modular test bit MMT is the logical OR of T_i bits for i = 1, 2, . . k. Hence, if there is any error in any of the modules C_i, then MMT will be logical 1, provided e = 0. Hence, the error is detected.
- The minimum number of garbage bits required to make function MMT reversible is ceil(log M), where M is the maximum number of times an output pattern is repeated in the truth table of MMT, D. Maslov et.al (2004). For the function MMT, $M = 2^{2k} - 1$, where k is the number of online testable reversible modules. Therefore, ceil(log M) = log ($2^{2k} - 1$) = 2k Hence, the.

III. ILLUSTRATION OF THE PROPOSED TECHNIQUE

To illustrate the proposed technique, a reversible decoder circuit is converted to an online testable reversible decoder. A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2^n unique output lines. Let us consider a 2-to-4 decoder with the enable bit. The construction of the reversible decoder circuit uses three Fredkin gates. A and B are the one-bit inputs to the decoder, E is the one-bit enable and O_1,O_2, O_3 and O_4 are the output bits of the decoder. Algorithm is applied to convert the decoder circuit into an online testable one. We illustrate this technique in a detailed step by step manner.

Input: Reversible Decoder Circuit C

Step 1: Replace Fredkin gate F_k with its Testable Reversible TRC (F_k) for k = 1, 2, 3. Let the input vector of F_k be [a, b, c] and the output vector be shown in figure Fig 3.

E	A	B	O_1	O_2	O_3	O_4
0	X	X	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

Fig 3: Truth table for 2:4 decoders.

Fig . Decoder Truth Table

Step 1: Replace Fredkin gate F_k with its Testable Reversible TRC (F_k) for k = 1, 2, 3. Let the input vector of F_k be [a, b, c] and the output vector be [O_1, O_2, O_3]. The deduced Fredkin gate, DR_a can be obtained as with the following as the inputs and outputs:

Inputs: a, b, c and P_{ia}

Outputs: O_1 = a; O_2 = a ⊕ ab ⊕ ac

O_3 = b ⊕ ab ⊕ ac; P_{oa} = O_1 ⊕ O_2 ⊕ O_3 ⊕ P_{ia}

P_{oa} = (a) ⊕ (c ⊕ ab ⊕ ac) ⊕ (b ⊕ ab ⊕ ac) ⊕ (P_{ia}) = a ⊕ b ⊕ c ⊕ (P_{ia})

The truth table of the deduced Fredkin gate DR_a is shown. To construct DR_b, take X to be a 3 x 3 gate that has inputs as [I_1, I_2, I_3] and outputs as [U_1, U_2, U_3], where U_i and I_i are related as U_i = I_i for i = 1, 2, 3.

The deduced gate DR_b is as shown in Figure 11, with the following as the inputs and outputs:

- Inputs: I_1, I_2, I_3 and P_{ib}
- Outputs: U_i = I_i where i = 1, 2, 3.

$P_{ob} = P_{ib} \oplus U_1 \oplus U_2 \oplus U_3 = P_{ib} \oplus I_1 \oplus I_2 \oplus I_3$

Truth table of DR_b is as shown in Table III. We cascade the above two deduced gates, DR_a and DR_b to get a Testable Reversible Fredkin Cell (TRC). $P_{ia} = P_{ib}$, let us set them to logic 0. This completes the construction of TRC(R) for the Fredkin gate. Now we replace each Fredkin gate F_k for k = 1, 2 and 3 in the input decoder circuit with TRC(R) constructed above.

Step 2: Add a Test Cell (TC) with 2n + 1 input line. As n = 3 for the given decoder circuit, TC has 2n+1 = 7 input lines. Connect its first six input lines to the parity bits P_{oak} and P_{obk} of the Fredkin gates F_k for k = 1, 2 and 3 as shown in Figure 12. First six input lines are passed to the output lines without any change. Output bit T is the error detecting bit. The value of T will determine if there is an error in the circuit.

Output: Circuit thus obtained is shown in Figure 12 and is the required online testable reversible decoder circuit C^T. Let us consider the case when the input vectors [A, B, E] = [1, 0, 1]. From the Truth Table I, if the circuit is error free, output vector O should be [0010]. In this case, the parity vector [P_{oa1}, P_{ob1}, P_{oa2}, P_{ob2}, P_{oa3}, and P_{ob3}] is equal to [0, 0, 1, 1, 0 and 0] and T = 0 which shows that the circuit is error free.

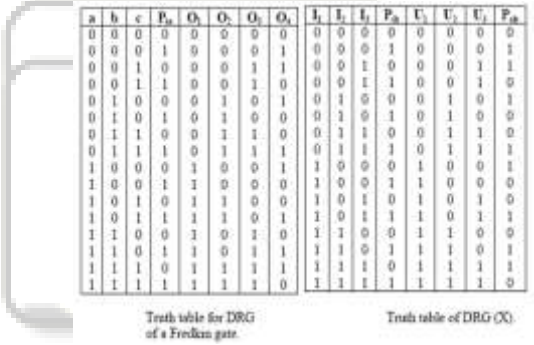

Fig 4 Truth table for DRG

In this case, the parity vector [P_{oa1}, P_{ob1}, P_{oa2}, P_{ob2}, P_{oa3}, and P_{ob3}] is equal to [0, 0, 1, 1, 0 and 0] and T = 0 which shows that the circuit is error free. Suppose there is some error in the circuit, say in F_1. For the given input vector [A, B, E] = [1, 0, 1], output of F_1 is [1, 1, 1] instead of [1, 0, 1]. In this case, parity vector [P_{ia1}, P_{ib1}, P_{ia2}, P_{ib2}, P_{ia3}, P_{ib3}] It is straight forward to infer that if there is any single bit error in the circuit, it will be indicated by the error bit T.

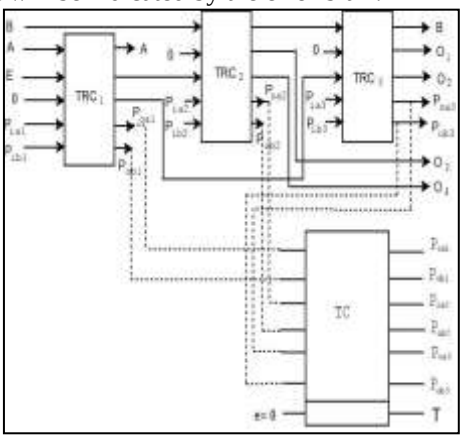

Fig 5: Block diagram of online testing 2:4 decoder

If the reversible circuit is made of more than one modules, then using the hierarchical structure these can be integrated by using MMTC making it online testable. Figure 5 shows the Block diagram of online testing 2:4 decoder.

IV. Simulation Result

A. *Test cell output*

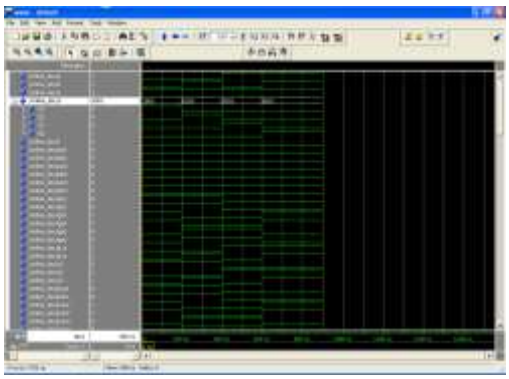

Fig. 6: Decoder with online testing with TC

B. *Multimodular test cell output*

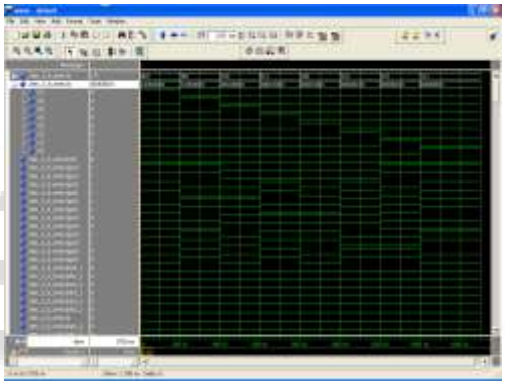

Fig. 7: Decoder with online testing with MMTC

V. Conclusion

In this paper, we proposed a methodology that converts any reversible gate into a testable reversible gate. Using the same, any reversible circuit made of reversible gates can be converted to an online testable one with minimum garbage. The resultant testable circuit can detect online any single bit errors that include Single Stuck Faults and Single Event Upsets. An important advantage of the technique is that the design of a reversible circuit need not be changed for the purpose of adding testability feature to it. This paper proposes the construction of multi-modular online testable reversible circuits hierarchically.

References

[1] R. W. Keyes and R. Landauer, "Minimal energy dissipation in logic," IBM J. Research and Development, pp. 152–157, March 1970.
[2] C. H. Bennett, "Notes on the history of reversible computation," IBM J. Research and Development, vol. 32, pp. 16–23, January 1988.
[3] R. Landauer, "Irreversibility and heat generation in the computing process," IBM J. Research and Development, vol. 3, pp. 183–191, July 1961.

[4] C. H. Bennett, "Logical reversibility of computation," IBM J. Research and Development, pp. 525–532, November 1973.
[5] R. Feynman, "Quantum mechanical computers," Optics News, vol. 11, pp. 11–20, 1985.
[6] T. Toffoli, "Reversible computing," Automata, Languages and Programming, pp. 632–644, 1980.
[7] E. Fredkin and T. Toffoli, "Conservative logic," Int'l Journal of Theoretical Physics, vol. 21, pp. 219–253, 1982.
[8] P. Kerntopf, "Synthesis of multipurpose reversible logic gates," Euromicro Symposium on Digital System Design (DSD'02), pp. 259–267, 2002.
[9] N. Margolus, "Physics and computation," Ph. D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, 1988.
[10] S. G. Younis and T. F. Knight, "Practical implementation of charge recovering asymptotically zero power cmos," Proceeding of the 1993 symposium on Research on integrated systems, MIT press, pp. 234–250, 1993.

