Application of Convolutional Neural Networks to Regenerate Deterministic Test Patterns for BIST

Tsutomu Inamoto and Yoshinobu Higami
Graduate School of Science and Engineering, Ehime University
inamoto@ehime-u.ac.jp

Abstract

This study displays preliminary results on a simple technique to improve fault coverages in the BIST technology. The technique assumes that a circuit which implements an ANN can be used with the target circuit, and effective test patterns are given beforehand. Such ANN circuit is utilized as a degraded memory which approximately regenerates given test patterns. In computational illustrations, fault coverages of test patterns by the technique are calculated on c7552 of the ISCAS'85 benchmark.

Keywords: combinational circuit, fault detection, logic BIST, test pattern, artificial neural network

1. Introduction

Nowadays people use highly condensed Large-Scale Integrated circuits (LSIs) in their electric appliances such as smart phones in their everyday life. Corrupted LSIs can easily break down the world, so it is necessary to make efforts to keep LSIs reliable. Companies which produce LSIs have tested and distinguished resultant LSIs into 2 categories; faulty or not. This production test is obviously important. However, its relative importance will lessen since more and more LSIs are going to be used in outdoors. The production test cannot detect future faults of LSIs.

One solution to keep LSIs in outdoors reliable is the built-in self test (BIST)[1]. In the BIST technology, an LSI is equipped with a test pattern generator (TPG). Such LSI in operation can test itself by using test patterns generated by its TPG. The heart of the BIST is to generate test patterns which are effective in the sense that they can detect many faults. Another solution is to use special circuits for generating pseudo random test patterns. The Linear Feed-back Shift Register (LFSR)[1] is a typical of such circuit. One of its strong points is that a proper LFSR circuit of m registers can generate all of $2^m - 1$ bit patterns except for zero. However, the LFSR has such a weak point that it is unavoidable to generate useless bit patterns which cannot detect any fault. The ideal technique which yields most effective test patterns is to store test patterns which are known effective in a memory. However, this technique is impractical because the number of such test patterns is numerous. Here, the output and input of a circuit of such memory are the test pattern and its address, respectively. It is expected that such input-output relation can be represented by the artificial neural network (ANN), which is widely applied nowadays due to the development of the deep learning technology.

In this study, we focus on a simple technique to improve fault coverages of LSIs by using the ANN. This technique presumes that circuits which implement ANNs are available and reliable. In the technique, each test pattern is associated with a binary vector which represents the address of that pattern, and the relation between such binary vector and the test pattern is learned in an ANN circuit. Then, by using such ANN circuit and a counter circuit which covers all addresses of the test pattern, each effective test pattern will be regenerated. This expectation is investigated preliminarily by calculating fault coverages of c7552 of the ISCAS'85 benchmark.

2. LSI test problem

2.1 Fault model of this study

On the logic circuit, there are many fault models such as the bridging fault, the open fault, the delay fault, and the stuck-at fault [1]. In this study, the single stuck-at fault model is considered because it is most basic. In this fault model, a faulty line in a circuit has logic value "0" (stuck-at-0, sa0) or "1" (stuck-at-1, sa1) no matter what input signals are applied to the circuit. The logic circuit is classified into sequential or combinational. This study considers only the latter type because the sequential circuit can be tested as same as the combinational one by the scan design [1].

2.2 Fault detection problem

The LSI test is classified into the fault detection and the fault diagnosis. The objective of the former is to judge whether a subject circuit (Circuit under Test, CUT) contains faults or not. The objective of the latter is to distinguish which faults happened in the subject circuit. This study focuses on the fault detection.

In the fault detection, we can say that a CUT is faulty by using a test pattern if its output vector

yielded by the CUT with no fault differs from an output vector yielded by the CUT with a fault. Let us denote $\mathbb{t} \in \{0,1\}^{N^{\mathrm{I}}}$ a test pattern which represents the bit vector for primary inputs (PIs) of the CUT, where N^{I} denotes the number of PIs of the CUT. Let us denote $G_k(\mathbb{t}) \in \{0,1\}^{N^{\mathrm{O}}}$ the output vector of the CUT when it has fault $k \in \hat{\mathcal{F}} \coloneqq \{-N^{\mathrm{L}}, -N^{\mathrm{L}}+1, \ldots, -1, 0, 1, \ldots, N^{\mathrm{L}}-1, N^{\mathrm{L}}\}$ and is applied test pattern $\mathbb{t}$, where N^{O} and N^{L} denote the number of primary outputs (POs) and the number of lines of the CUT, respectively. Here, it should be noted that $k = 0$ means there is no fault and negative (positive) k means that an sa0 (sa1) fault happens at line k. By using these symbols, the objective of the automatic test pattern generation is said to find such $\mathbb{t}$ for each fault $k \in \mathcal{F} \coloneqq \hat{\mathcal{F}} \setminus \{0\}$ that satisfies $G_0(\mathbb{t}) \neq G_k(\mathbb{t})$. The fault detection is usually conducted by using a test set $\mathcal{T}$ which is comprised of test patterns. The fault coverage $f(\mathcal{T})$ of test set $\mathcal{T}$ is calculated as follows:

$$f(\mathcal{T}) \coloneqq \frac{|\{k \in \mathcal{F} \mid \sum_{\mathbb{t} \in \mathcal{T}} d_k(\mathbb{t}) \geq 1\}|}{|\mathcal{F}|} . \quad (1)$$

Here, $d_k(\mathbb{t})$ denotes whether test pattern $\mathbb{t}$ can detect fault k or not and is defined as follows:

$$d_k(\mathbb{t}) \coloneqq \begin{cases} 1 & \text{if } G_0(\mathbb{t}) \neq G_k(\mathbb{t}) , \\ 0 & \text{otherwise} . \end{cases} \quad (2)$$

3. Regenerating deterministic test patterns by artificial neural networks

The technique focused in this study presumes that an effective test set is given. The technique is comprised of the following 3 stages: (i) to associate binary vectors with test patterns in the given test set, (ii) to create an ANN which represents the association between binary vectors and test patterns, and (iii) to implement the ANN as a reliable circuit. The stages (i) and (ii) are considered in this study. It is one of most important future works to clarify the allowable resources in implementing ANNs as circuits, then to improve the accuracy of the ANN in the limitation of the allowable resources.

4. Computational illustrations

The focused technique is evaluated by calculating fault coverages of c7552 of the ISCAS'85 benchmark. That circuit has 207 PIs, and 108 test patterns comprise the test set which can detect all of detectable stuck-at faults in that circuit. As stage (i) in Sect. 3, [0, 107] integers are represented as 7-bit vectors then associated with 108 test patterns. As stage (ii) in Sect. 3, the ANNs are designed to contain convolutional layers as displayed at Fig. 1. Three ANNs were created, learned, then evaluated with different pseudo random numbers. Their losses on binary cross-entropy and accuracies of regenerated test patterns are displayed in Table 1. This table contains fault coverages of 108 test patterns regenerated by ANNs and those by an internal-XOR LFSR of 207 bits. We can see in Table 1 that all trials of this study completely regenerated test patterns.

The number of weights of the ANNs is roughly $7 \times 7 + 7 \times 7 \times 14 + 98 \times 207 = 21{,}021$. If each weight is implemented as an op-amp of 3 transistors, the number of transistors becomes 63,063. On the other hand, if a clocked D-latch is implemented as 6 transistors, the LFSR requires about $6 \times 207 = 1{,}242$ transistors. The fault coverage of $108 \times \frac{63{,}063}{1{,}242} \simeq 5{,}484$ test patterns by the LFSR was 96.58 [%] on average of 3 trials.

Table 1: Fault coverages of 108 test patterns and statistics related to ANNs [%]

Trial number		1	2	3
LFSR	Coverage	92.26	91.45	92.13
ANN	Loss	0.3199	0.3138	0.1740
	Accuracy	100.0	100.0	100.0
	Coverage	100.0	100.0	100.0

5. Conclusion

In this study, we display a simple technique to regenerate deterministic test patterns by using ANNs. That technique succeeded in achieving higher fault coverages than those of test patterns by an LFSR. There are many future works which include to examine resources in implementing ANNs as circuits and to evaluate the technique on larger circuits.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 19K11877.

References

[1] N. K. Jha and S. Gupta, "Testing of Digital Systems," Cambridge University Press, June 2012.

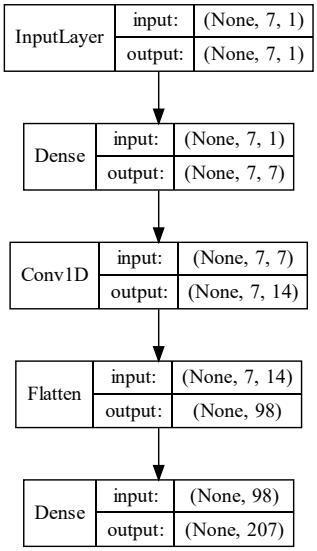

Figure 1: The design of created ANNs