JOURNAL OF SEMICONDUCTOR TECHNOLOGY AND SCIENCE, VOL.15, NO.3, JUNE, 2015
http://dx.doi.org/10.5573/JSTS.2015.15.3.423

ISSN(Print) 1598-1657
ISSN(Online) 2233-4866

Multi-operation-based Constrained Random Verification for On-Chip Memory

Hyeonuk Son, Jaewon Jang, Heetae Kim, and Sungho Kang*

***Abstract*—Current verification methods for on-chip memory have been implemented using coverpoints that are generated based on a single operation. These coverpoints cannot consider the influence of other memory banks in a busy state. In this paper, we propose a method in which the coverpoints account for all operations executed on different memory banks. In addition, a new constrained random vector generation method is proposed to reduce the required random vectors for the multi-operation-based coverpoints. The simulation results on NAND flash memory show 100% coverage with 496,541 constrained random vectors indicating a reduction of 96.4% compared with conventional random vectors.**

***Index Terms*—Constrained random verification (CRV), functional verification, coverpoint, NAND flash, constrained random vector**

I. INTRODUCTION

Many contemporary verifications for on-chip memory have adopted constrained random testing [1, 2]. Constrained random verifications (CRVs) have received significantly less attention, despite their contribution to verification. In [3], a word-level constraint-solving system was proposed to analyze constraint problems. Recently, parallel schemes that enable concurrent logic and memory operations have been used in many devices [4]. These parallel schemes, such as interleaving, improve memory throughput by executing many operations in each bank at the same time. Therefore, coverpoints for functional verification need to consider these parallel operations.

In this work, we present a CRV method for on-chip memory in which the coverpoints guarantee multi-operations across numerous memory banks. Since the multi-operation-based coverpoints require a greater test stimulus than the previous ones, an effective generation method for constrained random vectors is also proposed. The proposed constrained random vectors allow all coverpoints to be covered with fewer patterns.

II. PROPOSED CRV METHOD

In on-chip memory under a parallel system, unexpected faults can occur because of interference between operations. In this paper, we present a new coverpoint by considering the parallel operations. Fig. 1 describes the conventional single-operation-based coverpoint and the proposed multi-operation-based coverpoint. In Fig. 1(a), because the conventional coverpoints target an operation in a certain memory bank, the interference from operations in other memory banks is ignored. In contrast, as shown in Fig. 1(b), the proposed coverpoint is generated based on multi-operations, which include the interference between busy and active state operations. This is composed of busy state information (bank #*m*, operation k_1) and active state information (bank #*n*, operation k_2), represented by ($[m, k_1]$-$[n, k_2]$).

Since the proposed coverpoints consist of numerous components, a large number of random vectors are required to achieve 100% coverage. To reduce the

Manuscript received Mar. 21, 2015; accepted May. 31, 2015
Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea
E-mail : shkang@yonsei.ac.kr

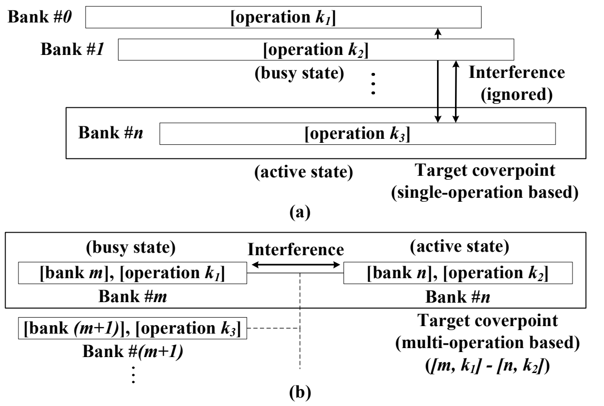

Fig. 1. (a) Single-operation-based coverpoint (conventional), (b) Multi-operation-based coverpoint (proposed).

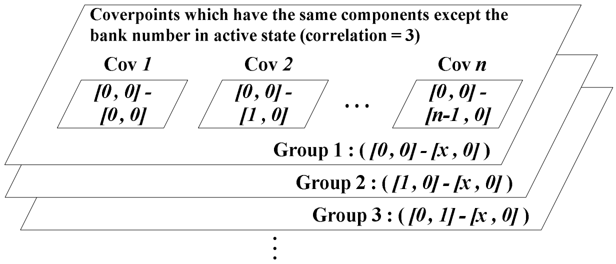

Fig. 2. Generation of coverpoint groups (for the bank number in the active state).

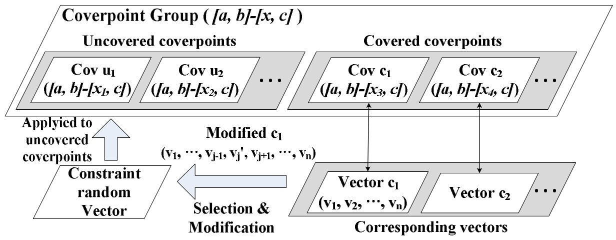

Fig. 3. Generation of constrained random vectors.

number of random vectors required under the multi-operation-based coverpoints, we propose a new generation method of constrained random vectors. The proposed random vectors examine correlations between the generated coverpoints to create groups of highly correlated coverpoints. The correlation between two coverpoints is defined as the number of identical components. For example, the correlation between coverpoint *A([a, b]-[c, d])* and *B([a, e]-[c, d])* is determined to be 3.

To create the coverpoint groups easily, we need a criterion for classifying each group. Fig. 2 illustrates the generation of the coverpoint groups when the criterion is set to the bank number in the active state. This is expressed as x in Fig. 2. The coverpoints in the same group have the same components except for the criterion. In this case, the correlation is equal to 3. It is possible that the criterion has two or more components, which means more coverpoints in a group and fewer groups.

After the generation of coverpoint groups, the conventional random vectors are applied. In this procedure, random vectors covering the coverpoints are stored. The proposed constrained random vectors are applied when all coverpoint groups contain at least one covered coverpoint. Fig. 3 shows the generation of constrained random vectors. The proposed constrained random patterns are generated through the stored vectors. Because there is a high correlation between coverpoints in the same group, the corresponding vectors also have similarities. Therefore, the modified vectors from the corresponding vectors have a high probability of covering the coverpoints. In the corresponding vector set, one vector is selected at random, and its component (v_j) are modified to generate a constrained random vector. This vector is applied to all uncovered coverpoints in the group. This process is repeated until all coverpoints are covered.

The more coverpoints there are in a group, the more times the constrained random vectors can be applied.

However, if the group size is too large, coverpoints having no relevance are more likely to be in the same group. To estimate the most effective group size g, we define several variables and calculate the required number of patterns according to the group size ($P(g)$). $P(g)$ can be determined by (1), where n is the total number of coverpoints, h is the ratio of hard-to-cover coverpoints (HCC), and d is the average number of random vectors needed to cover HCC. The first term in (1) denotes the estimated number of patterns such that all coverpoint groups have at least one covered coverpoint. The second term represents the number of patterns required to cover all coverpoints in the groups. From (1), the approximate group size needed to minimize the required vectors can be calculated.

$$p(g) \cong (nh/g)\cdot d + (n/g)\cdot(g-1)^2 \qquad (1)$$

III. SIMULATION RESULTS

To confirm the efficiency of the proposed CRV for on-chip memory, simulations were conducted using C++. The NAND flash memory used for these verifications was provided by Hynix, and Table 1 shows the NAND flash memory operations used in the simulations. Because sub-operations are frequently executed during the main process on the request of memory controllers, we set the sub and main operations to work as a pair to constitute operations in the active state. For four memory banks, we generated 43,320 coverpoints using the proposed multi-operation-based method.

Fig. 4 shows the reduction in the number of required patterns compared with previous methods. In [3], the constraint problems were solved using a high-level netlist model. This is applied to the NAND flash to prevent the generation of patterns for banks in the busy state. For 43,320 coverpoints, 13,650,479 conventional random vectors and 2,536,129 constrained random vectors by [3] were required to achieve 100% coverage. When applying the proposed constrained random vectors, 496,541 patterns were required for a group size of 6. The results indicate a decrease in the number of required vectors of 96.4% and 80.4% compared with the conventional random vector and the proposed method [3], respectively.

Table 1. Operations of NAND flash memory

Sub (5)	Set parameter, Get parameter, Set feature, Get feature, Read status
Main (19)	Reset, LUN reset, Read ID, Single/Multi read, Single/Multi copy-back read, Single/Multi program, Single/Multi read cache (random), read cache (sequential), Single/Multi block erase, Single/Multi program with random data in, Single/Multi cache program with random data in, Read parameter page

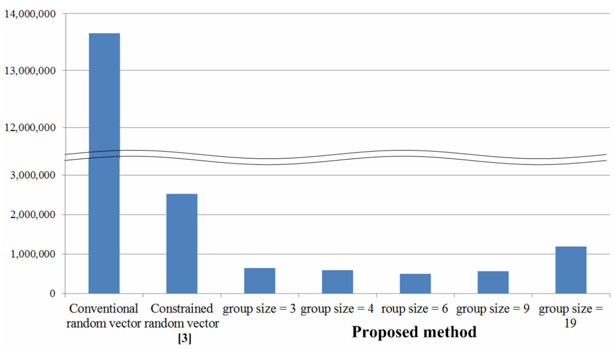

Fig. 4. Number of random vectors required to achieve 100% coverage.

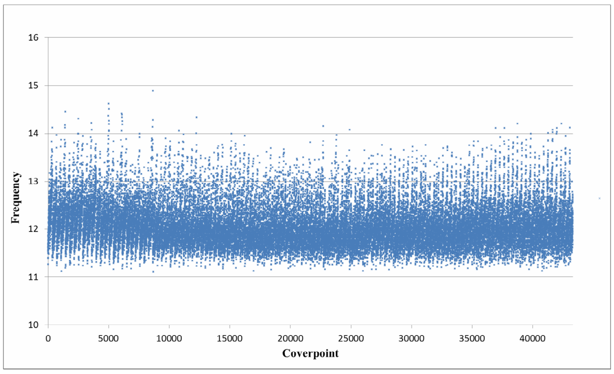

Fig. 5. Distribution of proposed random vectors.

To check that the proposed random vectors were uniformly generated, their distribution on each coverpoint was measured. For 100 simulations, the average number of random vectors satisfying each coverpoint is described in Fig. 5. Fig. 5 shows that the proposed constrained random vectors are uniformly distributed without depending on the particular constraints.

IV. CONCLUSIONS

In this work, a multi-operation-based CRV method was proposed. As the proposed coverpoints require a large number of random vectors, the constrained random vectors based on the coverpoint groups were created by considering the correlation between coverpoints. The simulation results show that the required number of random vectors could be decreased by up to 80.4% compared with the previous constraint solver system [3].

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2015R1A2A1A13001751).

REFERENCES

[1] N. Kichen and A. Kuehlmann, “Stimulus Generation for Constrained Random Simulation,” *International Conference on Computer Aided Design (ICCAD)*, pp. 258-265, Nov. 2007.

[2] M. D, Moffitt and G. E. Gunther, “Scalable

Scheduling for Hardware Accelerated Functional Verification," *International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 162-169, Jun. 2011.

[3] M. A. Iyer, "RACE: A Word-Level ATPG-Based constraints Solver System for Smart Random Simulation," *International Test Conference (ITC)*, pp. 299-308, Sep. 2003.

[4] M. Kudlugi, C. Sevidge and R. Tessier, "Static Scheduling of Multi-Domain Memories For Functional Verification," *International Conference on Computer Aided Design (ICCAD)*, pp. 2-9, Nov. 2001.

Hyeonuk Son received his BS in electrical and electronic engineering from Yonsei University, Seoul, Rep. of Korea, in 2007. Currently, he is working toward a combined MS/PhD in electrical and electronic engineering at Yonsei University. His current research interests include ADC/DAC testing, high-speed I/O testing, and 3D-IC testing.

Jaewon Jang received his BS and MS in computer science from Sogang University, Seoul, Rep. of Korea, in 2005 and 2007, respectively. Currently, he is working toward a PhD in electrical and electronic engineering at Yonsei University. His current research interests include ADC/DAC, BIST, high-speed I/O testing, and TSV testing.

Heetae Kim received his BS in electrical and electronic engineering from Yonsei University, Seoul, Rep. of Korea, in 2014. Currently, he is working toward a combined MS/PhD in electrical and electronic engineering at Yonsei University. His current research interests include logic BIST, high-speed I/O testing, and functional testing.

Sungho Kang received his BS in control and instrumentation engineering from Seoul National University, Seoul, Rep. of Korea, and his MS and PhD in electrical and computer engineering from the University of Texas at Austin, Austin, TX, USA, in 1992. He was a research scientist with the Schlumberger Laboratory for Computer Science, Schlumberger Inc., and a senior staff engineer with Semiconductor Systems Design Technology, Motorola Inc. Since 1994, he has been a professor in the Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Rep. of Korea. His main research interests include VLSI/SOC design and testing, design for testability, and design for manufacturability.