A New TSV Set Architecture for High Reliability

Jaeseok Park [1], Sungho Kang [2]
[1 2] Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea
[1] bbajaepjs@soc.yonsei.ac.kr, [2] shkang@yonsei.ac.kr

Abstract

Recently, 3D IC design is a very attracting issue, and the importance of system reliability increases. This paper proposes a new reliable and repairable TSV set architecture. The proposed architecture supports the previous TSV repair scheme using TSV redundancies and provides a defect/error detection function reutilizing residual TSV redundancies for high reliability of 3D ICs. This can be applied to both on-line test and soft error detection/analysis. The results show that the proposed TSV set architecture guarantees high TSV redundancy efficiency and reliability. And, the results show that the proposed TSV architecture achieves defect/error coverages which are steady and predictable by a simple formula.

Keywords

3D IC, TSV, redundancy, reliability

1. Introduction

In recent IC industries, 3D IC design is becoming a main issue of circuit design as need for higher speed and complexity of ICs increases. The one of main components of 3D IC architecture is a TSV(Through-Silicon Via) which is a vertical electrical connection passing completely through the silicon substrate of a wafer or a die.[1] To verify quality of TSVs, all TSVs in 3D ICs should pass two test steps, pre-bond and post-bond tests [2][3][4][5]. The pre-bond test is accomplished when each wafer is produced. Then, the post-bond test is required when the wafers passing pre-bond test are stacked. Test results are analyzed and faulty TSVs are checked.

After each TSV test step, faulty TSVs should be repaired for raising yield, actually, replaced by TSV redundancies which are extra TSVs built for the repair process. Many repair techniques have been proposed lately. [6][7] Usually, a TSV set which is composed of the fixed numbered TSVs and TSV redundancies shares connections using switches or fuses, and the connections are selected according to the test results. When repair processes have done, there can be residual TSV redundancies which are not utilized in the repair processes. They are just metal lines with no use.

As the density of ICs increases like 3D IC, abnormal phenomena from overheat and signal interference, such as malfunctions and system failures, occur more frequently in use.[8] So, the importance of system reliability grows remarkably, especially about TSVs between stacked dies. Even if all TSVs pass tests, there may be problems such as soft errors and defects from aging. Error detection and analysis schemes have been researched for solving this problem in 2D IC. However, in 3D IC, other paths are required between dies for error detection logic of TSVs, and these require highly additional cost.

This paper proposes a new TSV set architecture having repair ability and high reliability. It supports TSV repair using TSV redundancies first, then it can detect defects and errors reutilizing residual TSV redundancies. These residual TSVs can be used as passages of error detection logic between dies. Section 2 reviews TSV repair system and Section 3 describes the proposed TSV set architecture. Then, section 4 analyzes the performance of the proposed architecture and shows the proposed architecture achieves steady and predictable defect/error coverages. At the end, Section 5 concludes the paper.

2. Previous TSV set architecture

For the TSV repair step, the circuit designers build TSV redundancies and a routing logic. There are two representative TSV repair concepts: signal-switching [9] and signal-shifting [10]. Only faulty TSVs are replaced by the fixed redundancies in the former concept. So, there is no priority among the redundancies for repair. On the other hand, connections of TSVs are shifted from faulty TSVs to the redundancies toward one by one in the latter concept. The latter should change much more connections, but can reduce a delay gap problem. The both repair systems can produce residual TSV redundancies when any TSV redundancies are not used for repair. Then, these residual TSV redundancies descend to unnecessary metal line in the IC.

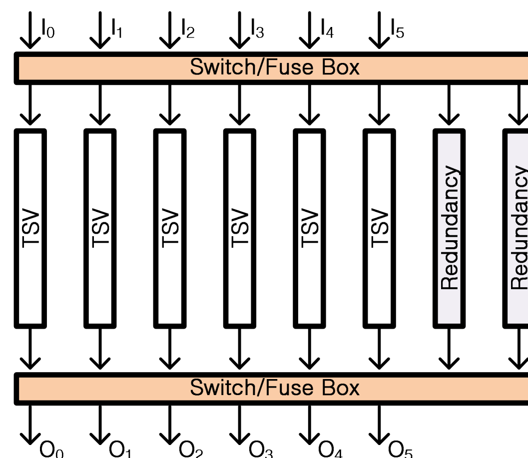


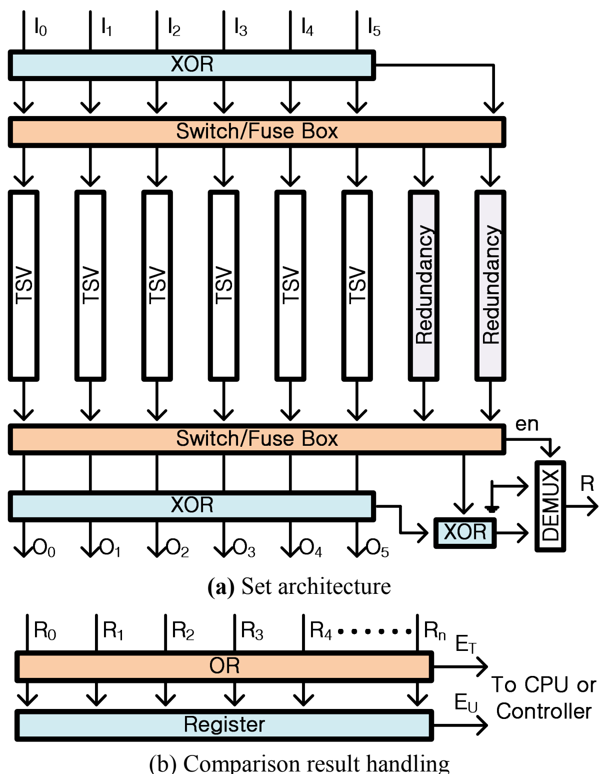
Figure 1: Signal-shifting repair

In this paper, a reliability-improved TSV set architecture based on a signal-shifting TSV set as shown in Figure 1 is proposed. The proposed architecture can use residual redundancies of the signal-shifting repair scheme more efficiently than those of the signal-switching repair scheme because the redundancies of signal-shifting have different priorities in use of repair. The connection of simple logics

for reliability is assigned first to the TSV redundancy which has the lowest priority. This calls the reduction of wasteful routing. TSV connections can be rearranged using switches or fuse boxes if the number of faulty TSVs is no more than the number of redundancies

3. Proposed TSV set architecture

Figure 2(a) describes a simple example of the proposed TSV set architecture, which is composed of 6 TSVs and 2 TSV redundancies. In the proposed architecture, the switch/fuse box needs to be modified for routing a few additional signals such as 1-bit input-XOR-to-redundancy, 1-bit redundancy-to-XOR, and *en* which is an enable signal of an XOR result. A signal *R* is the XOR result between XORed inputs and outputs. If there is no residual TSV redundancy from the repair process, a signal *en* is set as *0* to hold *R* as *0*. On the other hand, the inputs of TSVs are compared to the outputs of them using the XOR function if there are any residual TSV redundancies. The XOR function result of inputs is delivered through a TSV redundancy which has the lowest priority of replacement for repair. This simple example can detect the odd-numbered defects or errors because just 1-bit comparison is accomplished.



(a) Set architecture

(b) Comparison result handling

Figure 2: Proposed TSV set architecture

The comparison results of *n* TSV sets in a 3D IC, *R*s, are handled as shown in Figure 2(b). Error/defect occurrence is recognized by observing E_T which is an OR function of all *R* values from TSV sets. So, E_T is triggered on when any TSV sets have defects or errors. Then, the location of TSV sets where defects or errors exist can be analyzed by shifting E_U which is serial data of *R* values, if the analysis is necessary. If *m* test input patterns are used, only *m* test clocks are required in case of all-good TSVs. On the other hand, *n* test clocks are additionally required for one test pattern which detects defects or errors to diagnose the location of faulty TSV sets when E_T is triggered on.

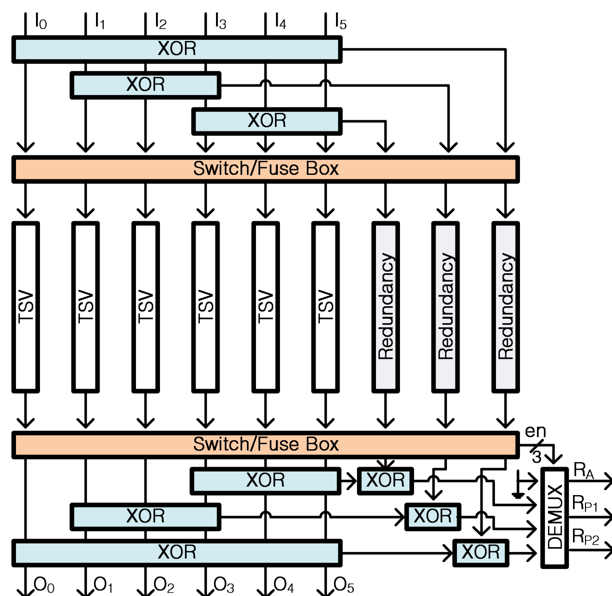


Figure 3: An example of extended Set architecture

The proposed TSV set architecture can be extended as an example composed of 6 TSVs and 3 TSV redundancies shown in Figure 3 for a higher defect/error detection coverage. Additional XOR trees which cover different input/output groups are added in the extended version. The XOR trees as many as residual TSV redundancies can be available and additional XOR trees increase the defect/error detection coverage. The half part of the even-numbered defects/errors case is covered with an additional XOR tree (second XOR tree) of the extended one while the TSV set including single full XOR tree can detect only odd-numbered defects/errors. And, a third XOR tree can cover more a half of the rest undetectable cases. This is proved in section 4.

This proposed TSV set architecture can be applied to various ways for high reliability, representatively, on-line test and soft error detection.

First of all, this proposed architecture can be applied to a simple way of on-line test. There are 3 steps for on-line test with the proposed TSV set as shown in Figure 4. On Step 1, the existence of faulty TSV sets is checked by observing E_T during test pattern insertion. So, on-line test with *m* test patterns request *m* test clock for only fault-existence check. To diagnosis of the location of faulty TSV sets, all *R* values should be analyzed in step 2. $n \cdot s$ test clocks are needed additionally for shifting *R* values through E_U. (*n* is the number of TSV sets in the 3D IC, and *s* is the number of XOR stages in the TSV sets, respectively.) Therefore, Step 2 can be started only when E_T is triggered on in Step 1. Optionally, a TSV diagnosis may be done by using design-for-test circuits or reutilizing flip-flop chains built for the pre-bond test.

Even if there is no faulty component in an IC, an intermittent malfunction can be happened. This is caused by

soft errors in the IC which occur from environmental reasons. These soft errors can be blocked by the normal test, so signal should be observed all the time. This proposed TSV set can catch soft errors during system operations. E_T can alert soft error occurrence with simple observation. In case of recognition of soft errors, all tasks or processes at that time restart for avoiding malfunction. To enhance efficiency of this process with the proposed TSV set, R values from TSV sets can be analyzed through E_U if the restart of all tasks at soft error occurrence is too big burden. The locations of TSV sets where soft errors are generated is diagnosed by analyzing R values, then the involved parts or IPs related with those TSV sets can be figured out. From this analysis, fewer tasks should restart, so unnecessary task executions can be avoided.

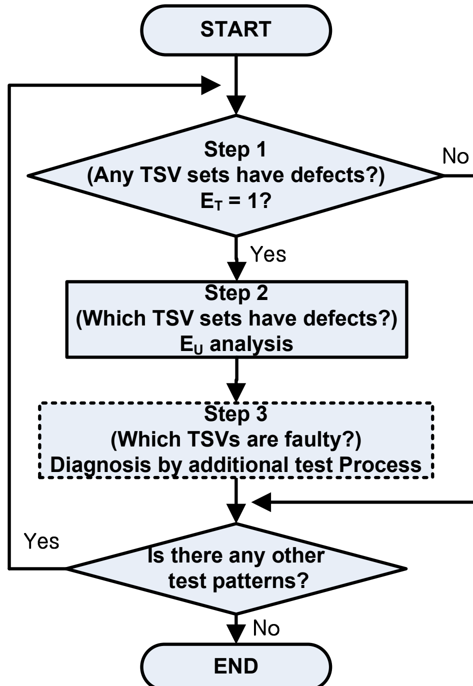


Figure 4: Flow chart of online test with the proposed TSV set

4. Results

Table 1. Comparison among three TSV units

TSV Set	Basic	Repairable	Proposed
Redundancy efficiency	None	Low	High
Repair support	No	Yes	Yes
TSV on-line test support	No	No	Yes
TSV soft error detection	No	No	Yes
Hardware overhead	Low	Medium	High

Table 1 shows the performance comparison among the previous works and the proposed TSV set. The proposed TSV set includes repair architecture for satisfactory yield, and reutilizes residual TSV redundancies after the repair process to build logics for high reliability. This additional hardware for reliability supports on-line test and soft error detection, and results are observed by two kinds of signal, E_T and E_U, which have different priorities to minimize time cost. On the other hand, the additional hardware cost is required for XOR trees and modification of switch/fuse boxes.

An extended architecture is proposed to raise defect/error coverage and reutilization rate of residual TSV redundancies. Table 2 shows the result in the extended architecture example including 6 TSVs and 3 TSV redundancies as shown in Figure 3. The defect/error coverage increases as more residual TSV redundancies are available. 32 kinds of defect/error occurrences are detected using only full XOR trees because only the odd-numbered defect/error can be recognized. And, 16 and 8 defect/error occurrence cases are covered additionally when one and two results of partial XORs are observed. So, the proposed architecture guarantees almost 90% defect/error coverage when 3 residual TSV redundancies are available in the example. If t XOR trees are used for a TSV set which includes m TSVs, a total defect/error coverage is defined as (1).

$$C_T = \frac{(2t-1)2^{m-1}}{t(2^m-1)} \tag{1}$$

Table 2: Three cases of the extended set in Figure 3

# of residual redundancies	XOR tree usage	Defect/error coverage
1	Only full trees	32/63 (50.8%)
2	Full trees and 1 pair of partial trees	48/63 (76.2%)
3	Full tree and 2 pairs of partial trees	56/63 (88.9%)

Table 3. Comparison according to the number of 2nd XOR tree inputs in case of 6 TSV set

# of inputs of 2nd XOR tree	# of defects/errors	Defect/error coverage
2	2	8/15 (53.3%)
	4	8/15 (53.3%)
3	2	9/15 (60%)
	4	6/15 (40%)
5	2	5/15 (33.3%)
	4	10/15 (66.7%)

The defect/error coverages according to the number of defects or errors vary in different characteristic of the partial XOR tree when 2 residual TSV redundancies are available. Table 3 shows the coverage difference when the number of inputs of partial XOR trees is 2, 3 and 5 in case of a TSV set which include 6 TSVs. The result of a 4-input-XOR tree is the same as that of a 2-input-XOR tree. And, the XOR trees which have the odd numbered inputs can catch the 6-defect/error case. Therefore, the overall defect/error coverage is fixed as 76.2% in Table 2. The XOR tree with 3 inputs will be the most useful if the probability that fewer defects or errors occur is higher. However, the XOR tree with 5 inputs can be the best one if defects or errors are generated usually in an assembled group.

Table 4 describes defect/error coverage results according to different numbered TSVs included in a TSV set when two residual TSVs redundancies are available. The XOR trees which catch fewer defect/error cases well are selected for

this comparison. The overall coverage of all cases is similar, and it is about 50%. Also, the defect/error coverages according to the number of defects/errors are steady near 50%.

Table 4. Comparison according to the number of TSVs in case of 2 residual TSV redundancies

# of TSVs	# of defects/errors	Defect/error coverage	# of inputs of 2nd XOR tree
4	2	4/6 (66.7%)	2
5	2	6/10 (60%)	
	4	2/5 (40%)	
6	2	9/15 (60%)	3
	4	6/15 (40%)	
7	2	12/21 (57.1%)	
	4	16/35 (45.7%)	
	6	4/7 (57.1%)	
8	2	16/28 (57.1%)	4
	4	32/70 (45.7%)	
	6	16/28 (57.1%)	

5. Conclusion

A TSV repair process after pre-bond and post-bond tests exists commonly in the 3D IC manufacturing flow for increasing yield. When all TSV repair steps have done, residual TSV redundancies without replacement with faulty TSVs are useless in previous TSV set architecture. This paper proposes a new TSV set architecture which can reutilize these residual TSV redundancies for increasing reliability of 3D ICs. This TSV set supports not only repair, but also detection of defects for on-line test and soft errors during system operations. The extended set architecture can be adopted when higher defect/error detection coverages are required. This proposed TSV set architecture is more promising as the importance of system reliability increases.

Acknowledgment

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 2012R1A2A1A03006255).

References

[1] E. J. Marinissen, “Challenges and emerging solutions in testing TSV-Based 2.5D- and 3D-stacked ICs”, *Proc. of Design, Automation & Test in Europe Conference & Exhibition*, pp. 1277-1282, 2012.

[2] P. Chen, C. Wu, and D. Kwai, “On-chip testing of blind and open-sleeve TSVs for 3D IC before bonding”, *Proc. of VLSI Test Symposium,* pp.263-268, 2010.

[3] B. Noia and K. Chakrabarty, “Pre-bond probing of TSVs in 3D stacked ICs”, *Proc. of International Test Conference,* pp.1-10, 2011.

[4] Y. Huang, J. Li, J. Chen, D. Kwai, Y. Chou, and C. Wu, “A built-in self-test scheme for the post-bond test of TSVs in 3D ICs”, *Proc. of VLSI Test Symposium,* pp.20-25, 2011.

[5] B. Noia, K. Chakrabarty, S. K. Goel, E. J. Marinissen, and J. Verbree, “Test-architecture optimization and test scheduling for TSV-based 3-D stacked ICs”, *Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 11, Nov. 2011.

[6] Q. Xu, L. Jiang, H. Li, and B. Eklow, “Yield enhancement for 3D-stacked ICs: recent advances and challenges”, *Proc. of Design Automation Conference*, pp. 731-737, 2012.

[7] L. Jiang, Q. Xu, B. Eklow, “On effective TSV repair for 3D-stacked ICs”, *Proc. of Design, Automation & Test in Europe Conference & Exhibition,* pp. 793-798, 2012.

[8] J. Zhang, L. Yu, H. Yang, Y. L. Xie, F. B. Zhou, and W. Wang, “Self-Test method and recovery mechanism for high frequency TSV array”, *Proc. of International Conference on VLSI and System-on-Chip*, pp. 260-265, 2011.

[9] U. Kang, et al., “8 Gb 3-D DDR3 DRAM using through-silicon-via technology”, *Trans. on Journal of Solid-State Circuits*, 45(1):111–119, Jan. 2010.

[10] A. Hsieh, and T. Hwang, “TSV redundancy: architecture and design issues in 3-D IC”, *Trans. on Very Large Scale Integration Systems*, vol. 20, no. 4, April 2012.