Architectural Evaluations on TSV Redundancy for Reliability Enhancement

Yen-Hao Chen[*] yhchen@cs.nthu.edu.tw
Chien-Pang Chiu[*] s103062556@m103.nthu.edu.tw
Russell Barnes[†] russell@ece.ucsb.edu
TingTing Hwang[*] tingting@cs.nthu.edu.tw

[*]Department of Computer Science, National Tsing Hua University, Taiwan
[†]Department of Electrical and Computer Engineering, University of California at Santa Barbara, United States

Abstract

Three-dimensional Integrated Circuits (3D-ICs) is a next-generation technology that could be a solution to overcome the scaling problem. It stacks dies with Through-Silicon Vias (TSVs) so that signals can be transmitted through dies vertically. However, researchers have noticed that the aging effect due to the electormigration (EM) may result in faulty TSVs and affect the chip lifetime [1]. Several redundant TSV architectures have been proposed to address this issue. By replacing the faulty TSV with redundant TSVs which are added at design time, chips can achieve better reliability and longer lifetime. In this paper, we will study the tradeoff of various redundant TSV architectures in terms of effectiveness and cost. To allow the measurement of reliability more realistically, we propose a new standard, repair rate, to appraise the redundant TSV architectures. Moreover, to design a more flexible and efficient structure, we enhance the ring-based design [2] that can adjust the size of the TSV block and TSV redundancy.

I. Introduction

Three-dimensional integrated circuits (3D ICs) have been proposed to be an effective solution to the interconnect bottleneck of CMOS scaling. 3D ICs use through-silicon vias to connect dies such that signals can be transmitted through dies with TSVs. 3D ICs possess numerous advantages due to the shorter vertical interconnect paths provided by TSVs. These benefits include lower parasitic losses, reduced power consumption, higher I/O density and improved system performance. Although TSVs have the above benefits, the reliability of TSVs may become a serious problem due to electromigration (EM) [3]–[6], which causes TSVs to have voids when the chip is in use. These voids could increase the delay and resistance of TSVs and even result in chip failure.

There are many redundant TSV architectures [2], [7]–[9] that have been proposed to avoid chip failure caused by faulty TSVs. Previous work [7], [8] proposed row-based redundant TSV architectures. Hsieh et al. [7] proposed a signal-shifting design to form a TSV chain with one spare TSV as shown in Figure 1(a). Kang et al. [8] used a signal-switching design to improve the yield of its 3D DRAM as shown in Figure 1(b) where redundant TSVs are linked to two ends of each TSV row. With these two redundant TSV architectures, a row can tolerate one or two TSV failures. The advantage of those architectures is low area cost, because they just need some 2-to-1 MUXes and 3-to-1 MUXes for neighboring routing. However, one-dimension shifting of signals limits usage of redundant TSVs. Jiang et al. [9] proposed a router-based redundant TSV architecture. Figure 2 presents the illustration of router-based design. The router consists of three 3-to-1 MUXes, and at least one router is needed for one TSV. Their design can repair not only neighboring faulty TSVs but also distant faulty TSVs. It is superior for repairing TSV failures, but the disadvantage is that it requires too much area cost for routers. In addition, re-routing from a faulty TSV to another faraway redundant TSV may increase the path delay. Recently, Lo et al. [2] proposed a ring-based redundant TSV architecture. They noticed that most of the faulty TSVs caused by the manufacturing process are clustered. That is, if a faulty TSV is found in a TSV block, TSVs near it are also very likely to be faulty. Ring-based redundant TSV architecture considers this fault clustering problem and provides an area-cost efficient solution. Moreover, the shifting length of this architecture is always 1 so that it guarantees the minimum timing overhead of each signal. However, the ring-based design does not consider the aging effect.

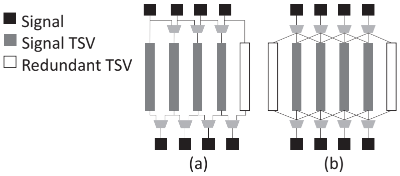

Fig. 1. (a) The signal-shifting design, (b) the signal-switching design.

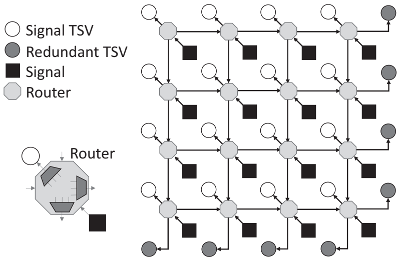

Fig. 2. The router-based redundant TSV architecture.

In this paper, we will focus on the reliability of redundant TSV architectures. Rather than the widely used mean-time-to-failure (MTTF) metric, we will also propose and use repair rate to measure the reliability of different architectures. Repair rate means the probability that a chip's time-to-failure is longer than the target lifetime. We will study what the best architecture is for reliability when the size of the TSV block is different by using the repair rate as a measurement. First, we will enhance the ring-based design because it is flexible and low cost among the above mentioned architectures. We will implement two area reduction methods to enhance the ring-based design and compare it to the original design using the metric of reliability. Next, we will evaluate the enhanced ring-based design and other redundant architectures for different sizes of the TSV blocks in terms of cost and repair rate for different sizes of the TSV blocks.

The remainder of this paper is organized as follows. Section II introduces the metric of reliability, our motivation and objective. Section III gives our evaluation methodology. Next, in Section IV, we will present two methods to enhance the ring-based design. Section V shows the evaluation results of five redundant TSV architectures including three designs mentioned above and two enhanced ring-based designs. Finally, the conclusions are given in Section VI.

II. Reliability

First of all, we introduce how to measure the reliability. Next, the motivation and the objective are presented.

978-3-9815370-8-6/17/$31.00 ©2017 IEEE

A. *Metric for reliability*

Mean-time-to-failure (MTTF) is a common measurement for chip lifetime. MTTF represents an average time of which a chip can reasonably be expected to perform in the field based on specific testing. There are many studies proposed concerning TSV reliability and methods for calculating MTTF. Frank et al. [10] concerned resistance increase due to electromigration. When the resistance of a TSV increases, the delay will increase. A TSV is failed if the delay is longer than the timing constraint. Equation (1) shows the increased resistance of a TSV due to electromigration.

$$R(t) = R_0 - A\ ln(\frac{t}{t_0}) \tag{1}$$

R_0 is the initial resistance of the TSV, A is the slope of the resistance degradation on a logarithmic scale, and t_0 is the time when the resistance begins to increase exponentially. Because the parameters are affected by multiple values, such as TSV barrier resistivity, TSV barrier thickness, copper thickness, radius and the portion of vacancy flow which generates the void under the TSV, Jiang et al. [1] used Gaussian distribution as the variation of the parameters to obtain the MTTF of TSVs. Ye et al. [11] also discussed delay problems of open defects. We will borrow these methods to calculate the TSV MTTF. For the resistance increase, we follow the previous work [1] to use a Gaussian distribution to obtain parameters, such as R_0 and A. We also apply the Gaussian distribution to t_0. Parameter values are chosen to simulate exceptionally harsh conditions for the hardware [1]. For the delay calculation in redundant TSV architectures, we follow the simulated results from the previous work [11]. We also use the delay simulations to calculate the resistance value of faulty TSVs. To summarize, given R_0, A and t_0, we calculate t for when the TSV is failed and then average the MTTF of the TSVs for different parameters.

Given the above assumptions, we compute the mean-time-to-failure (MTTF) of four redundant TSV architectures under harsh conditions as shown in Figure 3. By the figure, all of the redundant TSV architectures can survive after 3 years for a size of 8×8 to 20×20. However, MTTF does not present the yield rate under a target lifetime. Although MTTF is a good metric for the lifetime, it is hard to determine the reliability using simply MTTF. That is, when we have a chip, what we care about is not just how long the MTTF can be. Instead, we care about the probability that the chip can survive longer than a target lifetime. To realize the concept, we introduce a new measurement about reliability using probability. Given a target lifetime, t_life, a metric of reliability, $repair\_rate_{t\_life}$, is defined as the probability of a chip of which MTTF is longer than t_life.

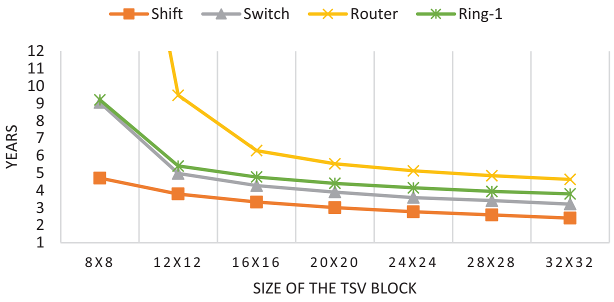

Fig. 3. The mean-time-to-failure (MTTF) comparisons of the redundant TSV architectures.

B. *Motivation and objective*

Because various sizes of the TSV blocks are used for 3D ICs with different complexities, we like to understand how the best redundant TSV architecture for different sizes of the TSV blocks differs. Figure 4 shows for a target lifetime of three years the $repair\_rate_{3-year}$ of five architectures with different sizes of TSV blocks. The X-axis represents sizes of the TSV blocks and the Y-axis represents $repair\_rate_{3-year}$. $None$ means no redundant TSV architecture, $Shift$ means the signal-shifting design, $Switch$ the signal-switching design, $Router$ the router-based design, and $Ring{-}1$ means our enhanced ring-based design which will be discussed in Section IV. When the target year is 3 years, both $Router$ and $Ring{-}1$ achieve above 99% repair rate for all sizes of the TSV blocks. When the repair rate is required to be 95%, $Shift$ architecture reaches the rate for a size of 4×4 to 8×8 while $Switch$ reaches the rate for a size of 4×4 to 16×16. However, in terms of hardware cost, $Shift$ and $Switch$ are much less than the $Router$ architecture. On the other hand, Figure 5 shows the $repair\_rate_{2-year}$ of five architectures for different sizes of the TSV blocks. When the target year becomes shorter, $Shift$ architecture can achieve above 99% for a size of 4×4 to 12×12 while $Switch$ reaches the rate for all sizes.

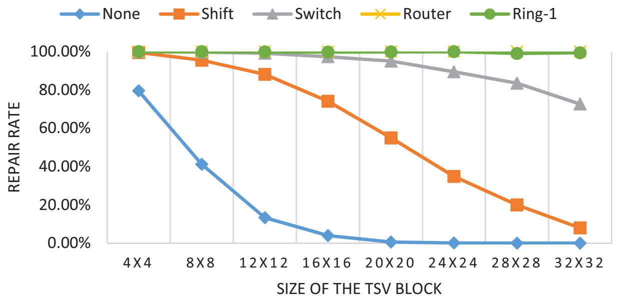

Fig. 4. The $repair\_rate_{3-year}$ comparisons of the redundant TSV architectures.

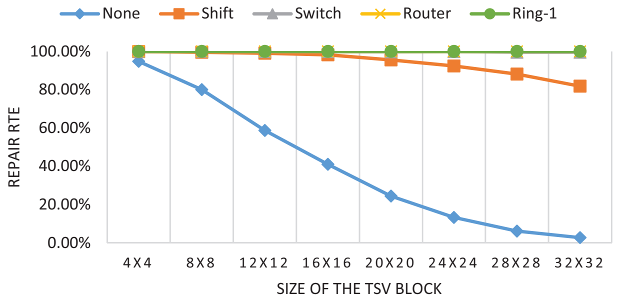

Fig. 5. The $repair\_rate_{2-year}$ comparisons of redundant TSV architectures.

For different specifications, the best architecture is different. For example, with a 95% repair rate specified, when the size is 4×4, all of the architectures can achieve the target repair rate, but $Shift$ is the best choice because it requires the lowest area cost. For another example, with a 99% repair rate specified, when the size is 32×32, only $Router$ and $Ring{-}1$ architectures can achieve the target repair rate. After comparing the area overhead, $Ring{-}1$ is the best choice in this condition. Hence, we can say that the best redundant structure is different in terms of cost and repair rate for different sizes of the TSV blocks.

Following the above observations, in this work, we will study the relations among the size of the TSV block, redundancy rate, routability, and $repair\ rate$. Redundancy rate is related to the number of redundant TSVs in the structure and routability to the total number of multiplexers (MUX). Based on different aspects, we will compare five different redundant TSV architectures in the following sections.

III. Evaluation Flow

Before discussing how to enhance the ring-based design and comparing all the redundant TSV architectures, we firstly present the evaluation flow. Figure 6(a) describes our evaluation flow. To evaluate an architecture, a target lifetime and TSV parameters are given as inputs. Then, we initialize $time$ to zero and start the steps. If $time$ exceeds the given target lifetime, the case passes. If not, $time$ continues and TSV fault patterns are generated from the TSV parameters with Gaussian distribution. In the step $Maximum\ flow\ \&\ path\ check$, we use the maximum flow algorithm to check if the faulty TSVs can be repaired [2]. If the repair fails before the given target lifetime, the case is reported fail. Otherwise, $time$ is advanced and the iteration repeats until a pass or fail is reported. In total, 1000 cases are tested to obtain the $repair\ rate$.

In the step, $Maximum\ flow\ \&\ check\ path$, we check if the faulty TSVs can be repaired by redundant TSVs or not as shown in Figure 6(b). First, we perform the step $graph\ construction$ to construct the flow graph [2]. Let the flow framework be denoted as $G(V, E)$, where node set $V = \{Source\} \cup \{Signal\} \cup \{TSV\} \cup \{Sink\}$ and edge set $E = \{IE\} \cup \{AE\} \cup \{OE\}$. Nodes $Source$ and $Sink$ represent the source and the sink node of the flow network, respectively. Nodes in $Signal$ represent a set of signals, which feed into a set of signal TSVs and a set of redundant TSVs. IE, AE, and OE represent sets of incoming edges, assignment edges, outgoing edges. Incoming edges are constructed from node $Source$ to $Signal$, and outgoing edges are constructed from the TSVs to node $Sink$. An assignment edge $e_{ij} \in AE$ is constructed if a signal $v_i \in Signal$ can be safely transmitted by a TSV $v_j \in TSV$ without any timing violation. Figure 7 shows an illustration of a flow network. The capacity of incoming edges, assignment edges and outgoing edges are all set to 1. After the graph is constructed, the max flow can be computed in the next step. We apply a maximum flow algorithm on the graph to generate a flow value. If the flow value is less than the number of signal TSVs, there are some signals that cannot be transmitted by any TSV, and thus, the repairing fails. If the flow value equals the number of signal TSVs, then we check whether all the paths meet timing constraints. If there is any negative-slack path, we update the graph by removing the path and repeat the maximum flow algorithm. The iteration repeats until pass or fail is reported.

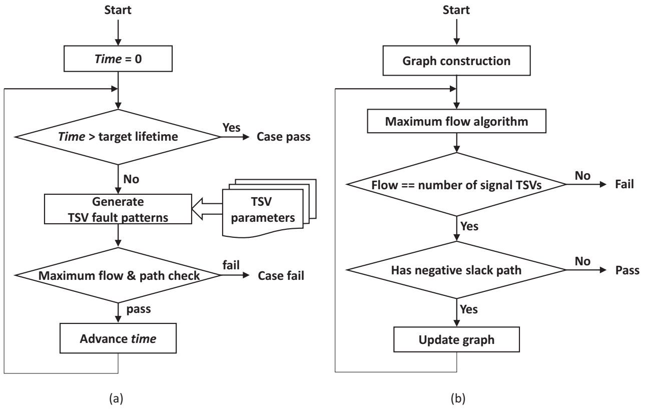

Fig. 6. (a) The evaluation flow chart, (b) the $Maximum\ flow\ \&\ path\ check$ steps.

IV. Enhancement of Ring-based Design

In this section, we enhance the ring-based design. The reason why ring-based design is chosen is because it is more flexible and cost-efficient than the other architectures. We expect that after we enhance the ring-based design, the new ring-based design can be more efficient in terms of the area cost. In this section, we firstly present the relationship between redundancy rate and repair rate, and then we propose two methods to reduce the MUX-area cost.

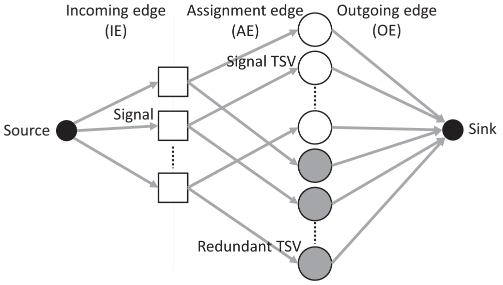

Fig. 7. An example of a flow network.

A. *The number of Redundant TSVs and repair rate*

Before proceeding with the enhancement of ring-based design, a review of original ring-based design is necessary. The overall ring-based redundant TSV architecture is illustrated in Figure 8. We use an 8×8 TSV block as an example. There are four rings in the design and four redundant TSVs which are located at the corners. The signal shifting directions of each ring are different. First, among rings, the signals of TSVs in the first ring which is also called outermost ring can be shifted inside to their neighboring TSVs in the second ring. The signals of TSVs in the inner rings can be shifted outside to their outer neighbor. The shifting direction within the outermost ring is dual, while the direction of the second ring is clockwise and the third ring counterclockwise. When the number of rings grows larger, the shifting directions of inner rings are clockwise and counterclockwise alternatively.

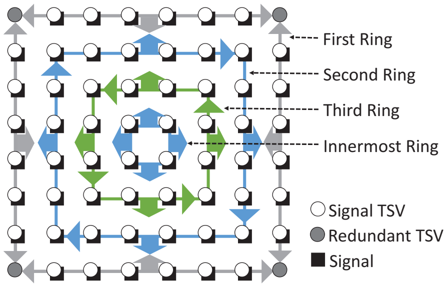

Fig. 8. The original ring-based redundant TSV architecture.

Our first design decision for modifying the ring-based design is to determine the number of redundant TSVs. When more redundant TSVs are used, a higher repair rate can be achieved. The higher number of redundant TSVs also means more area overhead. Choosing as small of a number of redundant TSVs as possible while maintain a high repair rate is very important. We would like to answer this first question. For different sizes of the TSV blocks and different numbers of redundant TSVs, we will compute the $repair\ rate$. Figure 9 shows how to calculate the number of redundant TSVs for different sizes of TSV blocks. Four corners of the outermost ring are designated as redundant TSVs because four corners have one less neighbor than other nodes, and their routability is less flexible than other nodes. Then, let $RTSV\_margin$ be the number of signal TSVs between two redundant TSVs. When $RTSV\_margin$ equals 0, the TSVs on four sides are all redundant TSVs. When $RTSV\_margin$ equals to 1, there is one signal TSV between two redundant TSVs on each side. Figure 9 shows an example of $RTSV\_margin$ = 2. A lower $RTSV\_margin$ means

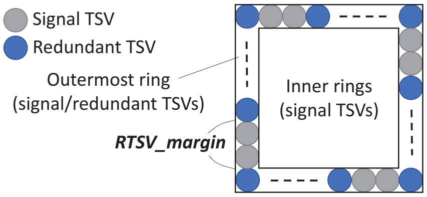

Fig. 9. Illustration of the enhanced ring-based design.

that there are more redundant TSVs in the structure and hence more TSV area cost.

Figure 10 shows the results of different $RTSV\_margin$ settings where $margin$-5 means that $RTSV\_margin$ is equal to 5, and so on. X-axis represents the size changing from 4×4 to 32×32. Y-axis represents repair rates. In Figure 10, we can obviously observe that when margin is equal to 6, repair rate decreases rapidly for sizes larger than 28×28. For $margin$-5, the repair rate can still be maintained above 99% with a 32×32 TSV block. Hence, $margin$-5 is chosen in the next subsection.

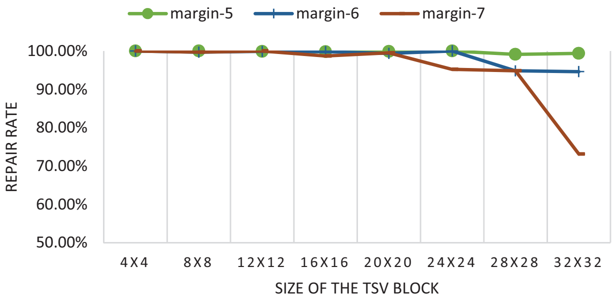

Fig. 10. The $repair\_rate_{3-year}$ comparisons of different $RTSV\_margin$.

B. *Two methods for MUX-area cost reduction*

In addition to cutting down the number of redundant TSVs, area overhead reduction by reducing the MUX-area cost is also important. Taking into consideration repair rate, we propose two methods to reduce the MUX-area cost. The first method is to consider reducing the outermost ring area cost. In order to allow the TSVs in all rings to be able to shift to another ring, TSVs in the outermost ring were designed to shift to its inner neighboring ring. We desired to understand if the design choice is effective, so we performed an experiment on the outermost ring with and without connections to the inner ring. The result presented in Figure 11 shows that the repair rates with and without connections to the inner neighbor are the same. Therefore, the first modification is the removal of connections from outermost ring to its inner neighbor.

The second method is to consider the MUX-area cost of inner rings. In the original ring-based design, every TSV can re-route to its outer ring neighbor TSV using its MUX. Allowing all TSVs to connect to their outer ring requires too much MUX-area cost. A new parameter, Num_share, is used to adjust the inner ring's MUX-area cost. Num_share means the number of signal TSVs which share one MUX hardware to a shift signal to the outer ring. When $Num\_share = 1$, it means there is no inner ring reduction, i.e., the original design. When $Num\_share = 2$, it means that one out of two signal TSVs is able to shift its signal to the outer ring, and so on.

Num_share is changed from 1 to 3 for the experiment. When $Num\_share = 3$ or more, the repair rate decreases rapidly. The $share{-}2$ case can remain at a high repair rate when the size is less than 20×20. Therefore, we will choose $Num\_share = 2$ as the parameter to implement inner rings reduction.

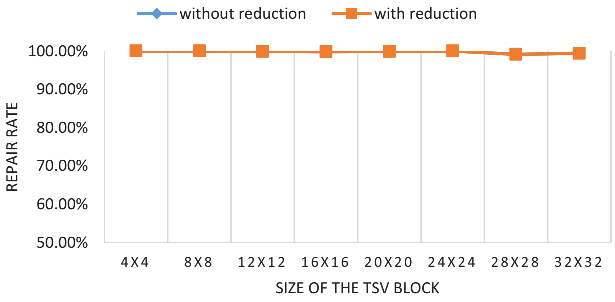

Fig. 11. The $repair\_rate_{3-year}$ comparisons between without and with an outermost ring reduction.

V. Evaluation of Redundant TSV Architectures

In this section, we compare redundant TSV architectures using $repair\ rate$. Notice that signal-shifting and signal-switching designs are one-dimension. We duplicate TSVs in rows to create a TSV block.

Figure 12 shows the repair rate of each redundant TSV architecture with 3-year target lifetime. The X-axis represents size and the Y-axis represents repair rate. $Ring{-}1$ is a ring-based design with only the outermost ring reduction. $Ring{-}2$ is a ring-based design with both the outermost and inner rings reduction of which Num_share equals to 2. We can see that the repair rate of $Switch$ and $Ring{-}2$ decrease when the size is larger than 20×20. $Ring{-}1$ and $Router$ achieve 99% repair rates with every size.

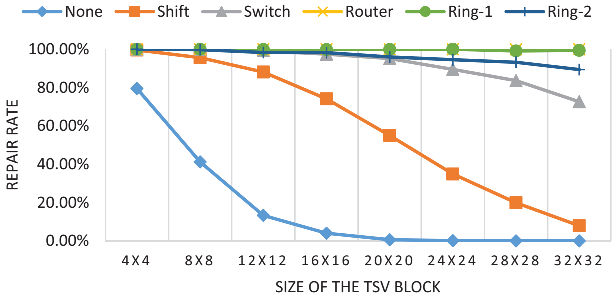

Fig. 12. The repair rate comparisons of the redundant TSV architectures with a 3-year target lifetime.

Next, we consider the area costs to implement different redundant TSV architectures. Figure 13 presents the MUX area cost of each redundant TSV architecture. The area of the MUX is based on the NanGate 45nm Open Cell Library [12]. The area of a 2-to-1 MUX is $1.862\mu m^2$ in this library. The 3-to-1 MUX consists of two 2-to-1 MUXes and the 4-to-1 MUX three 2-to-1 MUXes. Therefore, the area of a 3-to-1 MUX is $3.724\mu m^2$ and the area of a 4-to-1 MUX is $5.586\mu m^2$. The X-axis represents size and the Y-axis represents MUX area cost per signal TSV in μm^2. We can obviously see that $Router$ has higher MUX area cost than others because each signal TSV has a router which consists of three 3-to-1 MUXes. The MUX area cost of $Switch$ and $Ring{-}1$ are almost the same. $Ring{-}2$ is an obvious reduction compared to $Ring{-}1$. $Shift$ has the lowest MUX area cost because it uses fewer 2-to-1 MUXes.

Figure 14 shows the ratio of the number of redundant TSVs to the number of signal TSVs in the structure. The X-axis represents size, and

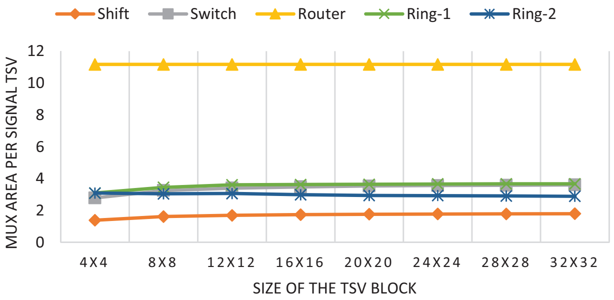

Fig. 13. The MUX area cost (μm^2) comparisons normalized to the number of signal TSVs.

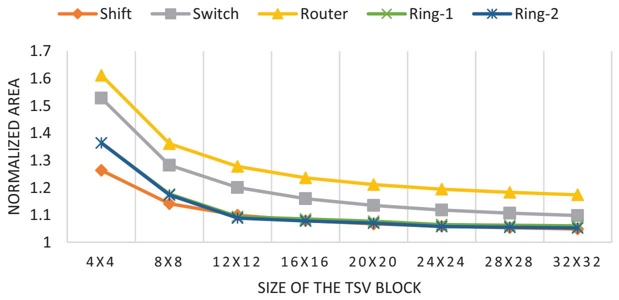

Fig. 15. The area overhead comparisons normalized to the area of design without redundant TSVs.

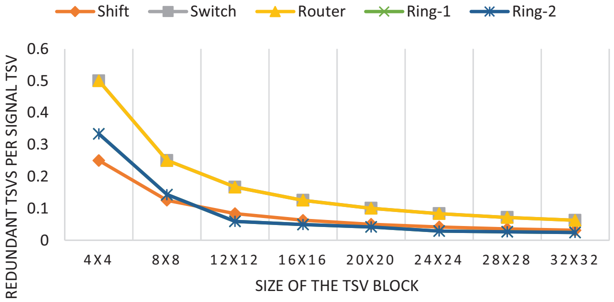

Fig. 14. The comparisons on the ratio of redundant TSVs and signal TSVs.

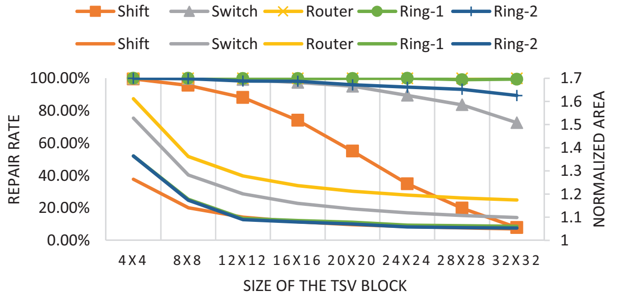

Fig. 16. The repair rate and area overhead comparisons with a 3-year target lifetime.

the Y-axis represents the ratio. Notice that the ratios of $Switch$ and $Router$ are the same. The reason is that we duplicate row-based TSV blocks to form a grid-based TSV block. Hence, the number of signal TSVs is the same in both $Switch$ and $Router$, and so is the number of redundant TSVs. Also, the ratios of $Ring-1$ and $Ring-2$ are the same because they use the same number of redundant TSVs. The ratio of $Switch$ and $Router$ is the highest because they use more redundant TSVs than the others. The ratio of ring-based designs is lower than the ratio of $Shift$ when size is larger than 12×12.

Now, we define the overall area overhead cost function to determine which design is better as the following:

$$Area\_cost = MUX\_area + \alpha \times TSV\_area. \qquad (2)$$

MUX_area means the total area of all of the MUXes. TSV_area is the total area of TSVs in the structure. The setting of TSV parameters follows the previous work [13] which also use 45nm technology library. TSV diameter is set to $5\mu m$ with a $2.5\mu m$ keep-out-zone. α is a user-defined parameter, and we set it 1 in our experiment. Figure 15 is the final area overhead calculated by the cost function defined in Equation (2). The X-axis represents size. The Y-axis represents area overhead normalized to the area of the design without any TSV redundancy. We can see that $Router$ has the highest area overhead and is followed by $Switch$, $Ring-1$ and $Ring-2$. $Shift$ has the lowest area overhead.

With repair rate, area overhead cost function and size of the TSV block, one likes to know the lowest area overhead cost of all of the architectures that achieves the target repair rate. We combine the charts of $repair\ rate$ and area overhead in Figure 16, which shows the relations between cost and repair rate for a 3-year target lifetime.

For a given repair rate, a normalized area-overhead can be determined. Let 99% be the expected repair rate. Figure 17 shows the results of Figure 16. With $Shift$, only the 4×4 TSV block achieves a 99% repair rate. With $Ring-2$, both 4×4 and 8×8 TSV blocks reach 99% repair rate. The $Switch$ architecture has reliable 4×4, 8×8, and 12×12 TSV blocks. Both $Router$ and $Ring-1$ can provide reliable TSV blocks for all sizes. When the size is small, e.g. 4×4, $Shift$ is the best choice in terms of the area overhead. When the size is 8×8, the $Ring-2$ architecture will be the most cost-efficient choice. With sizes of 12×12 and 32×32, $Ring-1$ reaches the repair rate and requires less area.

For another example, let the repair rate be 95%. Figure 18 shows the results of Figure 16. With the $Shift$ architecture, 4×4 and 8×8 TSV blocks reach 95% repair rate. The $Switch$ and $Ring-2$ architectures with sizes from 4×4 to 20×20 achieve the target repair rate. Both $Router$ and $Ring-1$ give reliable TSV blocks for all sizes. When the size is small, e.g. 4×4 and 8×8, $Shift$ is the best choice in terms of the area overhead. With a size of 12×12 to 20×20, $Ring-2$ architecture is the most cost-efficient choice. With a size of 24×24 to 32×32, $Ring-1$ is the best choice.

We also performed experiments for 5-year target lifetime. Figure 19 shows the results of $repair\_rate_{5-year}$ and area overheads of different architectures. We can obviously see that when the given target time is longer, using large sizes of TSV blocks results in a quickly-dropping repair rate. We observe that for sizes lower than 16×16, $Router$ still achieves a high repair rate, 99%. $Ring-1$, $Ring-2$, and $Switch$ can achieve a 99% repair rate only when the size is 4×4.

Finally, chip-level evaluation was performed. In this experiment, with a fixed total number of signals in TSV blocks, we would like to know how to select the size of the TSV block. The $Ring-2$ architecture is used and we assume that TSV blocks of the same size are used in a chip. Let the total number of signal TSVs be $total$ and the number of signal TSVs of a given size of TSV block be $signals$. The number of TSV blocks is computed by $\left\lceil \frac{total}{signals} \right\rceil$. Note that the smaller the TSV block size is, the more blocks there are.

Figure 20 shows the resultant repair rate. The X and Y axes represent

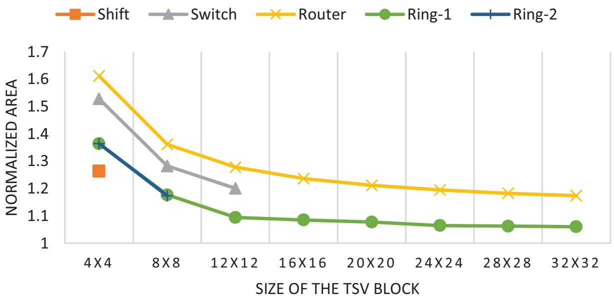

Fig. 17. The area overhead comparisons with a 3-year target lifetime and a 99% target repair rate.

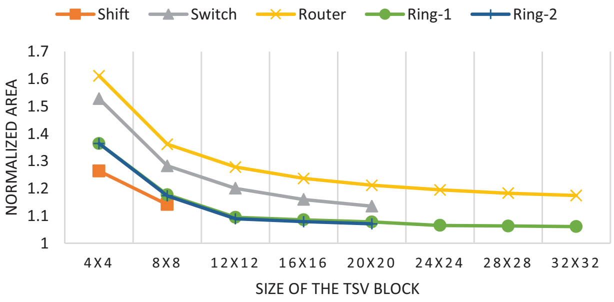

Fig. 18. The area overhead comparisons with a 3-year target lifetime and a 95% target repair rate.

the total number of signal TSVs and repair rate, respectively. From the figure, we can see that smaller sizes of TSV blocks are more reliable. Figure 20 also shows the results of area overhead. In this figure, we can see that in terms of normalized area overhead, a smaller TSV block is more efficient. Besides repair rate and area overhead, floorplanning and routing will also be affected by the size of the TSV block. In general, floorplanning and routing favors smaller TSV blocks.

VI. Conclusions

In this paper, we introduced a new measurement, *repair rate*, to compare the reliability of redundant TSV architectures. We also enhance the ring-based design to maintain a high repair rate with lower area overhead. In the evaluations, we compare several redundant TSV architectures with our own design. In terms of repair rate and an area overhead cost function, we can choose the most suitable architecture when given a size of the TSV block. Among the redundant TSV architectures compared, our ring-based design provides better reliability and area cost for reasonably large TSV block sizes.

References

[1] L. Jiang, F. Ye, Q. Xu, K. Chakrabarty, and B. Eklow, "On effective and efficient in-field TSV repair for stacked 3d ics," in *The 50th Annual Design Automation Conference 2013, DAC '13, Austin, TX, USA, May 29 - June 07, 2013*, pp. 74:1–74:6, ACM, 2013.

[2] W. Lo, K. Chi, and T. Hwang, "Architecture of ring-based redundant TSV for clustered faults," in *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition, DATE 2015, Grenoble, France, March 9-13, 2015* (W. Nebel and D. Atienza, eds.), pp. 848–853, ACM, 2015.

[3] S.-K. Ryu, K.-H. Lu, X. Zhang, J.-H. Im, P. S. Ho, and R. Huang, "Impact of near-surface thermal stresses on interfacial reliability of through-silicon vias for 3-d interconnects," *Device and Materials Reliability, IEEE Transactions on*, vol. 11, no. 1, pp. 35–43, 2011.

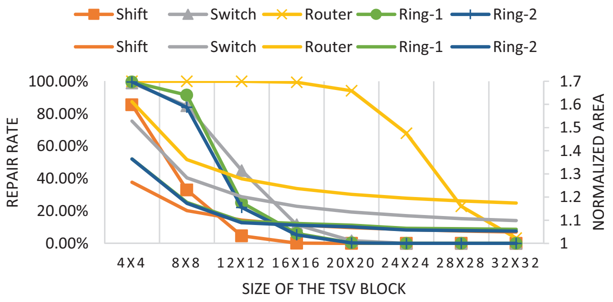

Fig. 19. The repair rate and area overhead comparisons with a 5-year target lifetime.

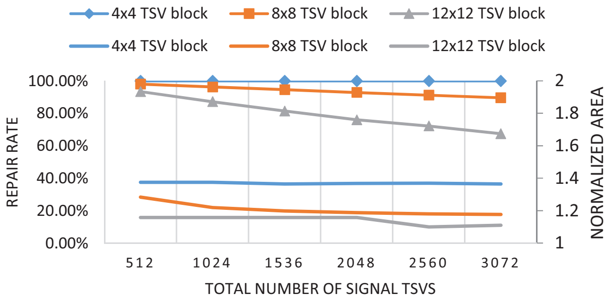

Fig. 20. The repair rate and area overhead comparisons of different sizes of the TSV blocks with the $Ring\text{-}2$ architecture.

[4] Y. C. Tan, C. M. Tan, X. Zhang, T. C. Chai, and D. Q. Yu, "Electromigration performance of through silicon via (TSV) - A modeling approach," *Microelectronics Reliability*, vol. 50, no. 9-11, pp. 1336–1340, 2010.

[5] M. Pathak, J. Pak, D. Z. Pan, and S. K. Lim, "Electromigration modeling and full-chip reliability analysis for BEOL interconnect in tsv-based 3d ics," in *2011 IEEE/ACM International Conference on Computer-Aided Design, ICCAD 2011, San Jose, California, USA, November 7-10, 2011* (J. R. Phillips, A. J. Hu, and H. Graeb, eds.), pp. 555–562, IEEE Computer Society, 2011.

[6] T. Frank, S. Moreau, C. Chappaz, L. Arnaud, P. Leduc, A. Thuaire, and L. Anghel, "Electromigration behavior of 3d-ic tsv interconnects," in *Electronic Components and Technology Conference (ECTC), 2012 IEEE 62nd*, pp. 326–330, IEEE, 2012.

[7] A. Hsieh and T. Hwang, "TSV redundancy: Architecture and design issues in 3-d IC," *IEEE Trans. VLSI Syst.*, vol. 20, no. 4, pp. 711–722, 2012.

[8] U. Kang, H. Chung, S. Heo, D. Park, H. Lee, J. H. Kim, S. Ahn, S. Cha, J. Ahn, D. Kwon, J. Lee, H. Joo, W. Kim, D. H. Jang, N. Kim, J. Choi, T. Chung, J. Yoo, J. Choi, C. Kim, and Y. Jun, "8 gb 3-d DDR3 DRAM using through-silicon-via technology," *J. Solid-State Circuits*, vol. 45, no. 1, pp. 111–119, 2010.

[9] L. Jiang, Q. Xu, and B. Eklow, "On effective TSV repair for 3d-stacked ics," in *2012 Design, Automation & Test in Europe Conference & Exhibition, DATE 2012, Dresden, Germany, March 12-16, 2012* (W. Rosenstiel and L. Thiele, eds.), pp. 793–798, IEEE, 2012.

[10] T. Frank, C. Chappaz, P. Leduc, L. Arnaud, S. Moreau, A. Thuaire, R. El Farhane, F. Lorut, and L. Anghel, "Resistance increase due to electromigration induced depletion under tsv," in *IEEE International Reliability Physics Symposium (IRPS'11), Monterey, CA, USA, April 10-14*, pp. 3F–4, IEEE Computer Society, 2011.

[11] F. Ye and K. Chakrabarty, "TSV open defects in 3d integrated circuits: characterization, test, and optimal spare allocation," in *The 49th Annual Design Automation Conference 2012, DAC '12, San Francisco, CA, USA, June 3-7, 2012* (P. Groeneveld, D. Sciuto, and S. Hassoun, eds.), pp. 1024–1030, ACM, 2012.

[12] Nangate, ""the nategate 45nm open cell library"." http://www.nangate.com/.

[13] F. Chen, H. Ting, and T. Hwang, "Fault-tolerant TSV by using scan-chain test TSV," in *19th Asia and South Pacific Design Automation Conference, ASP-DAC 2014, Singapore, January 20-23, 2014*, pp. 658–663, IEEE, 2014.