A Compressive-sensing based Testing Vehicle for 3D TSV Pre-bond and Post-bond Testing Data

Hantao Huang, Hao Yu
School of Electrical and Electronic Engineering
Nanyang Technological University
50 Nanyang Avenue, Singapore
haoyu@ntu.edu.sg

Cheng Zhuo
Intel Corporation
Hillsboro, OR 97124 U.S.A
cheng.zhuo@intel.com

Fengbo Ren
School of Computing, Informatics, Decision Systems Engineering
Arizona State University
699 S Mill Ave, Tempe, Arizona, U.S.A
renfengbo@asu.edu

ABSTRACT

Online testing vehicle is required for 3D TSV pre-bond and post-bond testing due to high probability of TSV failures. It has become a challenge to deal with large sets of generated testing data with limited probing when transmitting the data out. In this paper, a lossless compressive-sensing based testing vehicle is developed for online testing of TSVs. By exploring sparsity of the testing data under constraint of failure bound of TSV, sparse-representation based encoding can be deployed by XOR and AND network on chip to deal with large volume of testing data. Experimental results (with benchmarks) have shown that 89.70% pre-bond data compression rate can be achieved under 0.5% probability of failures; and 88.18% post-bond data compression rate can be achieved with 5% probability of failures.

1. INTRODUCTION

With the introduction of vertical through silicon via (TSV), 3D-IC provides energy efficient interconnect for memory and logic integration [1]. However, 3D-IC has low yield with high probability of TSV manufacturing defects [2]. TSV may be shorted to substrate due to pin-hole or open because of micro-void or partial filling [3]. To increase the yield, online TSV testing vehicle is thereby required to provide just-in-time diagnosis to detect faulty TSVs and replace them with redundant TSVs [4, 5] to achieve self-healing (TSV repair).

Due to the stacking nature of 3D-IC, TSV testing is needed for every layer before stacking. However, the testing data can be only collected from specially designed probe pad as shown in Fig. 1a, which are limited in number due to area limitation. After bonding, the bottom die can communicate with the external tester. Therefore, as shown in Fig. 1b, the testing data has to be transferred from bottom die to upper die through the so-called TSV elevator [6]. As the 3D-stacking is mainly applied for high-volume I/O circuits (with TSV density of $10,000/mm^2$ or more) [7, 8], it poses a grand challenge for TSV pre-bond and post-bond testing because the bandwidth of TSV testing data becomes the primary bottleneck [5, 9] for online TSV testing vehicle. As such, one needs to develop an efficient yet low-loss method to compress TSV testing-data with preserved fault detection information.

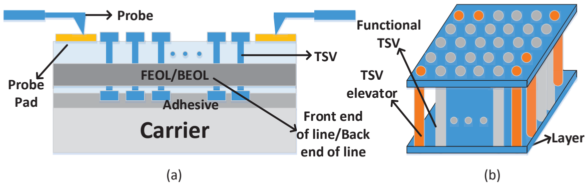


Figure 1: (a) Probing on extra DFT pads in pre-bond testing (b) TSV elevator in post-bond testing

Conventional testing data compression is widely used with according simple compaction circuits. It can be categorized as space compression and time compression. For space compression, XOR network is used to reduce many testing bits into single bit, which is however, limited by high aliasing rate because any even number of faulty responses could cancel error with a 'correct' response. For time compression such as multiple input signature register (MISR) method, it can achieve lower aliasing effect for conventional testing but cannot be directly deployed in the TSV testing because of the clustering effect [10, 11]. Furthermore, such a many-to-one compression method provides less online diagnosis information for TSV self-healing, which is a critical to improve the TSV yield.

In this paper, to perform online testing with improved bandwidth, we propose a lossless data compression method based on compressive-sensing for both pre-bond and post-bond TSV testing. It has no aliasing rate with on-chip data compression and off-chip lossless data recovery with according online hardware implementation for testing data compression. The problem of compressive-sensing based TSV testing data compression is formulated to find the maximum compression rate with lossless recovery under the constraint of TSV failure probability. For the pre-bond TSV testing, the failure probability refers to TSV failure probability due to variation of manufacturing [7]; whereas for the post-bond TSV testing, the failure probability is the faulty IC probability resulting in error output during functional testing

ISPD '16, April 3–6, 2016, Santa Rosa, California, USA.
© 2016 ACM. ISBN 978-1-4503-4039-7/16/04. . . $15.00
DOI: http://dx.doi.org/10.1145/2872334.2872351

[12]. Experiment results show that with the testing data compression, 89.70% data compression rate can be achieved under 0.5% failure probability and 88.18% post-bond data compression rate with 5% failure probability.

The rest of the paper is organized as follows. Section II provides the TSV online testing vehicle architecture with the according problem formulation. Section III discusses the compressive-sensing based testing data compression and recovery. Section IV shows the application in pre-bond and post-bond TSV testing. Section V presents the numerical results with conclusion drawn in Section VI.

2. COMPRESSIVE 3D-IC TESTING PROBLEM

In this section, we will present the 3D-IC testing architecture for pre-bond and post-bond online TSV testing with the probe pad and TSV elevator. Moreover, the problem of the testing data compression is formulated based on this testing architecture.

2.1 TSV Testing Vehicles

The online TSV testing vehicle architecture can support pre-bond die testing, post-bond stack testing and board level interconnect testing based on [6]. Pre-bond testing is mainly designed to detect TSV manufacturing faults; while post-bond testing is not only for TSV interconnect testing but also for scan-chain and functional testing. As shown in Fig. 2, the testing wrapper will provide testing access mechanism (TAM) and send/receive testing signal to/from I/Os of external stack. Testing data will be injected from probe pad for pre-bond testing or from bottom die to upper die through TSV elevator for post-bond testing. Decoder will decode the input testing data and function as Automatic Testing Pattern Generator (ATPG) to cover the sequential and combinational circuits. The testing data is collected from the scan chain and compressed by the testing data compression (TDC) block. Then, such compressed testing data will be transported from the upper die to the bottom and eventually collected by the tester or probe pad. For pre-bond TSV testing, the expected output is the same as the input testing data, whereas for post-bond TSV testing the expected output is collected from output bandwidth based on time-division multiplexing technique, in which the first slot is used for expected output and the remaining slots are for compressed testing data output [13]. Therefore, no additional bandwidth is required for expected output.

For the pre-bond TSV testing, testing each TSV is time consuming and impractical [5, 7], groups of TSVs are tested. The TSV groups are formed based on the pitch of probe head such that the probe can contact the whole TSV group at once. The faulty TSV interconnect can be detected by the difference between the received signal and expected signal. The target of pre-bond TSV testing data compression is to minimize the output bits to save testing time while be able to lossless recover original signal to provide diagnostic information for TSV repair. Probe pad is used to collect the pre-bond TSV testing data.

For the post-bond TSV testing, as there are only TSV elevators and probe pads to provide input and output data for non-bottom die, the testing time is constrained by the bandwidth, given a large volume of testing data. Moreover, increasing the number of TSV elevator will incur more die area and reduce the functional TSV densities. Therefore, data compression is needed to reduce the bandwidth requirement of TSV elevator and save testing time.

2.2 Problem Formulation

As previously discussed, lossless testing data compression is required to provide diagnostic information for TSV repair. The main proposed solution here is to fully utilize the compressive-sensing to compress the TSV testing data with high data compression rate under the given failure probability while being able to losslessly recover the original signal. Here, we assume that the TSV failure probability is known prior. As such, the TSV testing data compression problem can be formulated.

Problem: Find minimum number of output bits OB to locate faulty TSV/IC $E_{fault} \in \mathbb{R}^N$

$$\begin{aligned} & Min. < OB = MLog_2(Max(Y)) > \\ & S.T.(i)\ X_r = (I + E_{fault})X_e \\ & \quad (ii)\ Y = \Phi(X_r - X_e) \\ & \quad (iii)\ \|E_{fault}\|_0 \le K \end{aligned} \tag{1}$$

where $X_r \in \mathbb{R}^N$ and $X_e \in \mathbb{R}^N$ denote the received and expected testing data through TSV for pre-bond testing or scan chain for post-bond testing and $Y \in \mathbb{R}^M$ is the compressed output testing data. $\Phi \in \mathbb{R}^{M \times N}$ is the compressive-sensing matrix generated from TDC block using XOR-AND networks. M and N is the compressed testing data length and original testing data length respectively. $E_{fault} \in \mathbb{R}^N$ are the defective TSV location in the pre-bond testing, while for the post-bond testing, E_{fault} represents the error bits introduced by faulty ICs. Sparsity K represents the maximum number of non-zero values in E_{fault} , which can be estimated by the TSV failure probability or IC failure probability. The lossless compression thereby means that when given the compressed result Y and sensing matrix Φ, we can losslessly recover E_{fault} such that no TSV testing data information is lost. By making use of sparsity of E_{fault}, an unique sparse solution can be found for the undetermined linear system [14, 15]. Please note that only compression process performed as $Y = \Phi(X_r - X_e)$ is on-chip. The rest

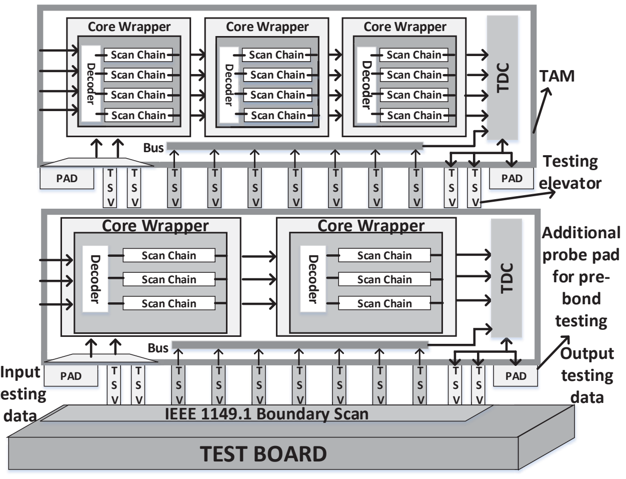


Figure 2: TSV testing vehicle with compressive-sensing based testing circuits

computation including testing data recovery is performed off-chip.

3. COMPRESSIVE-SENSING BASED COMPRESSION ALGORITHM

In this section, we discuss the sparsity of the testing data which is the foundation to perform compressive-sensing. Moreover, orthogonal matching pursuit is introduced here to solve undetermined equations. Please note that this section focuses on testing data analysis and recovery.

3.1 Sparsity of Testing data

Traditional lossless testing data compression methods such as length run (LR) coding and Golomb coding (GLC) are limited by the hardware implementation complexity for high volume of testing data. Compressive-sensing is one recently developed compression technique with sparse-representation of data by finding the most sparse solution of undetermined linear systems [14, 16]. Such a compression method results in simple encoding with comparable hardware implementation complexity as MISR since only XOR and AND network is required on-chip with data reconstruction off-chip.

To perform compressive-sensing, sparsity of testing data is estimated based on its yield. For TSV testing, signal is sparse in nature as only $1 - yield$ portion of data is nonzero if we compare the received result to expected outcome using XOR operation. For a common TSV yield such as 99% under pre-bond testing, there are 99% interconnects functioning properly. The defects are sparse in this sense indicating that the difference between the received and expected outputs are sparse with only 1% non-zero value at locations of faulty TSV interconnects. Similar to the pre-bond testing, the post-bond functional testing data can also be sparse by taking the difference between the received and expected data [12]. In order to know the sparsity, we need to estimate the TSV yield Y_{pre} and IC fault-free probability Y_{post}. Fig. 3 shows the testing data by XOR the expected and received expected output for 1024 TSV under different yields. Signal difference $'1'$ indicates an defective TSV. Thus, we can conclude that the higher yield, the more sparse the testing data will be.

Similar to [11, 17], we assume a uniform failure probability p for TSV under testing following binomial distribution. The overall probability of having x defective TSVs is

$$Y_{pre} = C_{N_{TSV}}^{x} p^{x} (1-p)^{N_{TSV}-x} \tag{2}$$

where N_{TSV} is the number of TSVs. The yield of pre-bond TSV can be calculated as $x = 0$. Similarly for post-bond TSV testing, the fault-free IC probability can be estimated using Poisson distribution [18].

$$Y_{post} = \frac{(N_{com} * P_{com})^{N_{fault}}}{N_{fault}!} e^{-N_{com} * P_{com}} \tag{3}$$

where N_{faulty} is the number of faulty components. N_{com} and P_{com} are the number of components and the probability of being faulty components respectively. The IC failure probability is $P_{post} = 1 - Y_{post}(N_{fault} = 0)$.

For post-bond, due to the application of ATPG and different functional testing algorithms, the testing data may not directly reflect the faulty component location. Therefore, the clustering effect is not considered. As the affected output bits due to faulty IC components depends on the testing algorithm, we assume the affected output signal (error bit probability) is proportional to faulty component probability. For pre-bond TSV testing, TSV testing data is directly related to faulty TSV location. The clustering effect is considered and there exists a spatial correlation between defective TSVs. This indicates that presence of a defective TSV increases the probability of defective TSV nearby. Based on [11, 17], the probability of defect for i-th TSV P_i can be modeled as below.

$$P_i = P(1 + \sum_{j=1}^{N_c} (1/d_{ic})^{\alpha}) \tag{4}$$

where P is the single TSV failure rate, d_{ic} is the distance between the TSV_i and cluster center, and α is the clustering coefficients indicating cluster extent. A large α indicates higher clustering effect. In our simulations, we assume the cluster center is injected randomly and only forms a proportion of total defective TSV number. The rest of failure TSV is generated with the combination of failure probability and clustering effect as mentioned in (4). The testing data compression rate N_C and output bandwidth improvement for both pre-bond and post-bond circuit can be defined as follows.

$$Bw_{Imp} = \frac{N}{M * log_2(Max(Y))}, \quad N_C = 1 - \frac{1}{Bw_{Imp}} \tag{5}$$

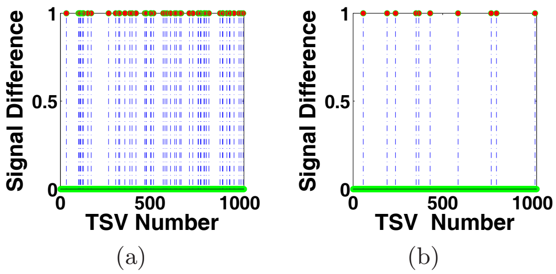
where $Max(Y)$ is maximum digital value of the compressed testing data, and N is the original data size. Please note that for a given bandwidth, data compression is equivalent to improve bandwidth, which has the same effect on throughput. We select the maximum code bit size $log_2(Max(Y))$ to ease the decoding after receiving them from probe pad or tester.

3.2 Lossless Compression and Recovery in L_1 Norm

The lossless compression is performed on-chip; whereas recovery is done off-chip. The recovery of compressed testing data can be formulated as L_0 norm minimization problem given below:

$$\begin{array}{ll} \text{argmin} & \|E_{fault}\|_0 \\ \text{subject to} & Y = \Phi E_{fault} \end{array} \tag{6}$$

where E_{fault} is N dimensional sparse testing data ($E_{fault} \in \mathbb{R}^N$), Φ is the sensing matrix ($\Phi \in \mathbb{R}^{M \times N}$) generated by TDC from XOR-AND network, and Y is the compressed testing data in low dimension ($Y \in \mathbb{R}^M$ and $M \ll N$). Note the solution of L_0 norm is equivalent to L_1 norm with overwhelming probability [14].



(a)

(b)

Figure 3: (a) Sparse data for yield (Y_{pre}) of 95% (b) sparse data for yield (Y_{pre}) of 99%

To ensure a successful recovery without loss, the sensing (or sampling) matrix must satisfy the restricted isometry property (RIP) [14]. In this paper, we use random Bernoulli matrix that can be implemented from pseudo number generator in hardware with well recognized RIP property. For the online TSV testing vehicles in Section 2, we can implement a counter to record non-zero values for lossless data compression and reconstruction. The minimum dimension M of compressed testing data [14] is

$$M = O(Klog(N/K)) \tag{7}$$

where K is the sparsity of data that can be estimated as (8) and N is the total length of original testing data. N has relationship with number of TSV N_{TSV} by $N = N_{TSV} * N_{data}$, where N_{data} is the number signal sent from each TSV. From (7), we can sample the signal based on its sparsity. The sparsity K can be estimated as below.

$$K = Ceil(N - NY_{pre}) \tag{8}$$

where Y_{pre} is the yield of pre-bond TSV testing. If we replace Y_{pre} by Y_{post}, the sparsity K for the post-bond TSV testing can also be calculated.

3.3 OMP based Compressed Data Recovery

As discussed in Section 3.1, L_0 norm solution can be applied to solve (6). In this paper, we deploy the Orthogonal Matching Pursuit (OMP) solver for the L_0 norm solution, which is a heuristic solver based on greedy algorithm to find the most sparse solution [19]. More details on compressed testing data recovery from OMP are provided in Algorithm 1. The residual is initialized as compressed testing data Y. Index set Λ_0 and chosen matrix Φ_0 are empty. The largest correlated column is found from Step 4, while index and chosen matrix will be updated as Step 5. The new estimated testing data is reconstructed in Step 6 via $L2$ minimization. The residual is updated from the estimated testing data and compressed testing data. The iteration will stop after K iterations. In summary, OMP performs two functions as follows. Firstly, it finds the most correlated column from the sensing matrix Φ by comparing simple dot multiplication. Secondly, the largest correlated column is added to the selected column and by solving a $L2$ norm minimization, the most fitted new estimate testing data is generated. This procedure will repeat K times to find the recovered testing data. Note that K is the sparsity of the testing data E_{faulty}, which can be estimated from the Y_{pre} and Y_{post}.

4. COMPRESSIVE-SENSING BASED HARDWARE IMPLEMENTATION

In this section, we will discuss the hardware implementation of compressive-sensing with XOR and AND network followed by the application of compressive-sensing to pre-bond and post-bond TSV testing.

4.1 Compressive-sensing based Testing Circuit

To perform the data compression using the proposed algorithm, outputs from the scan chain and expected output are provided as inputs for the testing data compression (TDC), as shown in Fig.2. Note that X-states (unknown states) will be masked to '0' based on the information from testing data controlled by mask controller. The TDC testing can be implemented using adders and XOR gates as shown in Fig. 4.

Algorithm 1 Orthogonal Matching Pursuit Algorithm

```
Require: An MxN TDC generated sensing matrix $\Phi$ =[ $\varphi_1$, $\varphi_2$, ...,$\varphi_N$], an M-dimensional compressed testing data $Y$ and yield $Y_{yield}$
Ensure: An accurate testing data $E_{faulty}$
1: Initialize the residual $r_0 = Y$, the index set $\Lambda_0 = \emptyset$ , $\Phi_{\Lambda_0} = \emptyset$ and iteration counter $t = 1$.
2: Calculate sparsity $K = Ceil(N - NY_{yield})$
3: While $(t \leq K)$
4:     Find column index $\lambda_t$ of $\Phi$ correlates $Y$ most as below
       $\lambda_t$= $argmax_{j=1,...N} | < r_{t-1}, \varphi_j > |$,
5:     Update column index set and matrix of chosen columns
       $\Lambda_t = \Lambda_{t-1} \cup \lambda_t$
       $\Phi_{\Lambda_t}$ =[$\Phi_{\Lambda_{t-1}}$ $\varphi_{\lambda_t}$]
6:     Solve a least squares problem to obtain new signal
       $x_t = argmin||$ $Y$-$\Phi_{\Lambda_t}$ $x$ $||_2$
7:     Calculate the new approximation and residual
       $a_t$ =$\Phi_{\Lambda_t} x_t$, $r_t = Y - a_t$, $t = t + 1$
8: End While
9: $E_{faulty}$ =$x_t$
```

The scan chain output and the expected output from the probe pad are XORed to obtain the testing data, E_{faulty} , which is normally sparse with $'1'$ to denote the failure. The Bernoulli function is realized using a linear feedback shift register (LFSR) with M measurements collected after performing M shifts. Here, M is the dimension of compressed testing data Y. Furthermore, the Bernoulli matrix is multiplied with the testing data using AND gates, where bits are added using the adder and fed to the probe pad. This implementation reduces the number of outputs from N dimensions to M dimensions. Note that the above-mentioned TDC implementation is suitable for both pre-bond and post-bond TSV testing on-chip.

4.2 Pre-bond TSV Testing

For pre-bond TSV testing, short or open defects will lead to receive incorrect testing data, which can be used to detect faulty TSVs after comparing to the expected data. Fig. 5a shows an example of TSV failure. In general, the size of probe heads is large compared to the pitch of TSVs. Hence, the probe head will contact a group of TSVs instead of each individual TSV. As shown in Fig. 5b, among the group of TSVs, only a few can have the I/O driving ability, which can be used to output the testing data and the rest can only

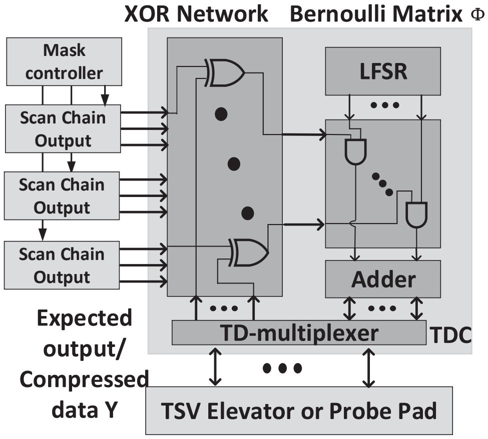


Figure 4: Compressive-sensing based testing circuit diagram for testing data compression

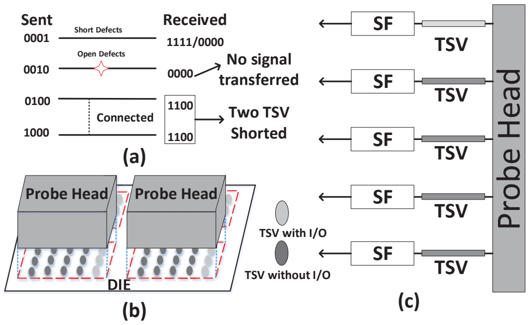


Figure 5: Conceptual diagram for pre-bond TSV testing

receive data from probe. As multiple TSVs are connected to one probe head, it is important to differentiate each TSV in the receiving end. Similar to [20], a scan flip-flop (SF) controlled by digital enable circuit is utilized to differentiate the input from the TSVs as described in Fig. 5c. This received data from top SF is provided as the input to the XOR network of TDC (in Fig. 4) and then the compressed testing data Y is sent through top TSV with I/O. The original testing data E_{fault} can be recovered from compressed testing data Y based on the proposed Algorithm 1.

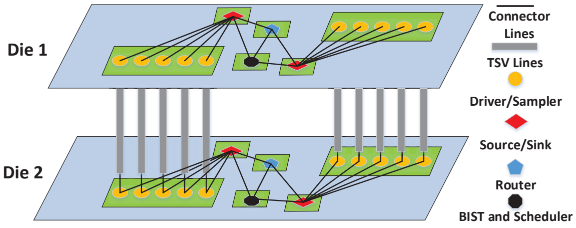


Figure 6: Conceptual diagram for post-bond TSV testing

4.3 Post-bond TSV Testing

For post-bond functional testing, our proposed compression technique can work independently or co-work with conventional MISR techniques to further compress the testing data depending on the lossless/lossy testing requirement. Our proposed compression algorithm provides general solutions to digital signal lossless compression. Fig. 6 shows a conceptual post-bond TSV testing diagram where built-in-self-test (BIST) circuit is shared between different TSV groups. The signal generated from BIST is scheduled and sent to router. Router will send the signal to TSV group driver for testing purposes by a sequence of digital bits similar as in the pre-bond test. Testing data compression (TDC) will receive the expected testing data from the router and compress them. Similarly ATPG can also be performed where the expected output is known. As shown in Fig. 4, the compressed testing data Y is collected and original testing data E_{fault} can be recovered from Y as shown in the proposed Algorithm 1.

4.4 TSV Testing Flow

Fig. 7 depicts the overall testing flow for the proposed TSV online testing vehicle. The input patterns and the expected output will be sent to device under test (DUT) and XOR-AND network respectively through input and output testing pads or elevators as shown in Fig. 2. Note that the expected output will be sent through output testing pads based on time-division multiplexer (TD-multiplexer). Therefore, no additional bandwidth is required. The testing data compression (TDC) will compress the testing data through XOR-AND network described in Fig. 4. The compressed testing data will be sent to off-chip computational resource for recovery. Note that as for the existing MISR-based compression method, faults are detected from one-in-many testing based on single signature. Insufficient diagnostic information is available to designers. Moreover, detection failure rate of MISR increases due to the error-cancelling effect under the clustering effect and the non-uniform TSV failure probability. In contrast, the proposed method performs the lossless compression without the error-cancelling effect. Moreover, it results in pin-level output of testing data for TSV self-repair or the other debugging capability.

5. SIMULATION RESULTS

In this section, we discuss the experiments set-up, the testing data recovery and compression for pre-bond and post-bond TSV testing. The compressive-sensing based TSV testing simulation platform is implemented in Matlab 2014a on a computer with 3.2 GHz core and 8.0G memory.

5.1 Experiment Set-up

Firstly, we present the compressed testing data of TSVs under different yields and the corresponding reconstructed data. The experiment is performed for 1024 number of TSVs with input as '1' for each TSV to verify whether '1' is received. Faulty TSV is inserted based on (2).

Secondly, for pre-bond TSV testing, 4096, 16384 and 65536 TSVs are tested with a scan signal and a 3-bit testing output [20]. To model defective TSV distribution, 10% defective TSVs are inserted as center in a TSV map; and the rest is generated based on clustering effect presented in (4).

Finally, for post-bond TSV testing, the proposed testing data compression method is applied for ISCAS-85 benchmarks [21] in Verilog based on stuck-at fault model. The testing pattern is generated using Mintest [22], which provides 100% fault coverage. An 8-bit output signal after scan testing is assumed and error bit probability is modeled based on (3) for stuck-at fault model. We select length-run (LR) coding and Golomb coding (GLC) coding for performance comparison as they both perform lossless compression and close to the entropy of information [23]. The area overhead of proposed TDC is synthesized using DC Synopsys

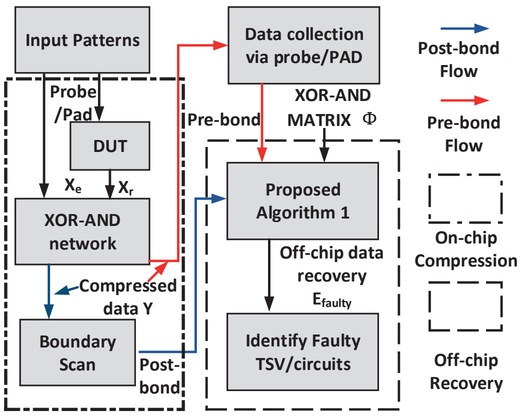


Figure 7: Testing flow for proposed testing vehicle

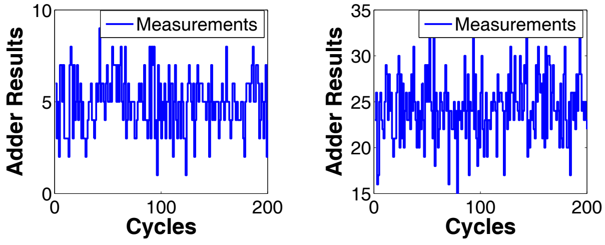
Table 1: Testing data Compression Rate and Bandwidth Improvement in Pre-bond Testing with 0.5 % (upper row) and 1% (lower row) Failure Rate

TSV Number	**Cluster**	**Proposed**		**Length Run Coding**		**Golomb Coding GS= 8**		**Golomb Coding GS= 16**		**Golomb Coding GS =32**	
		Comp. Rate	Bw Imp.	Comp. Rate	Bw Imp.	Comp. Rate	Bw Imp.	Comp. Rate	Bw Imp.	Comp. Rate	Bw Imp.
4096	$\alpha = 0$	89.45%	9.48 x	81.80%	5.49 x	76.83%	4.32 x	79.98%	5.00 x	81.23%	5.33 x
	$\alpha = 1$	89.70%	9.71 x	81.14%	5.30 x	77.29%	4.40 x	80.18%	5.05 x	81.30%	5.35 x
	$\alpha = 2$	89.32%	9.36 x	81.52%	5.41 x	76.68%	4.29 x	79.76%	4.94 x	80.98%	5.26 x
	$\alpha = 0$	65.29%	2.88 x	50.99%	2.04 x	35.57%	1.55 x	44.38%	1.80 x	48.02%	1.92 x
	$\alpha = 1$	65.03%	2.86 x	50.73%	2.03 x	35.57%	1.55 x	43.80%	1.78 x	47.29%	1.90 x
	$\alpha = 2$	66.48%	2.98 x	51.52%	2.06 x	37.16%	1.59 x	45.85%	1.85 x	49.51%	1.98 x
16384	$\alpha = 0$	89.16%	9.22 x	80.37%	5.09 x	75.95%	4.16 x	79.11%	4.79 x	80.38%	5.10 x
	$\alpha = 1$	89.23%	9.28 x	80.15%	5.04 x	73.99%	3.84 x	77.28%	4.40 x	78.63%	4.68 x
	$\alpha = 2$	89.35%	9.39 x	80.47%	5.12 x	75.59%	4.10 x	78.89%	4.74 x	80.27%	5.07 x
	$\alpha = 0$	64.80%	2.84 x	49.42%	1.98 x	34.08%	1.52 x	43.39%	1.77 x	47.35%	1.90 x
	$\alpha = 1$	64.29%	2.80 x	46.93%	1.88 x	29.49%	1.42 x	38.37%	1.62 x	42.03%	1.73 x
	$\alpha = 2$	64.86%	2.85 x	50.62%	2.03 x	34.51%	1.53 x	43.84%	1.78 x	47.73%	1.91 x
65536	$\alpha = 0$	89.17%	9.24 x	79.77%	4.94 x	73.48%	3.77 x	76.86%	4.32 x	78.26%	4.60 x
	$\alpha = 1$	89.21%	9.27 x	79.73%	4.93 x	73.09%	3.72 x	76.63%	4.28 x	78.12%	4.57 x
	$\alpha = 2$	89.24%	9.30 x	79.38%	4.85 x	73.49%	3.77 x	76.86%	4.32 x	78.26%	4.60 x
	$\alpha = 0$	65.32%	2.88 x	50.03%	2.00 x	34.27%	1.52 x	43.35%	1.77 x	47.17%	1.89 x
	$\alpha = 1$	64.59%	2.82 x	48.57%	1.94 x	34.19%	1.52 x	43.41%	1.77 x	47.28%	1.90 x
	$\alpha = 2$	64.88%	2.85 x	47.08%	1.89 x	26.68%	1.36 x	40.21%	1.67 x	45.34%	1.83 x

(D-2010.03-SP2) and estimated $4522um^2$ using $65nm$ process technology from Globalfoundries.

5.2 Compressive-sensing in TSV Testing

As shown in Fig. 4, the XOR of received and expected data is multiplied by a Bernoulli based sensing matrix Φ, which is generated from a binary pseudo number generator. The TSV with defects will not receive '1' and therefore the XOR result will be '1', indicating the failure of this TSV. Furthermore, one encoded output is generated from the adder based on the row of sensing matrix Φ multiplied by the XOR result. As an example, we collected 200 output measurements and plotted the adder output under different yields in Fig. 8. From Fig. 8(a), one can observe that the adder results are smaller for TSV with 99% yield, compared to adder results shown in Fig. 8(b) for TSV with yield 95%. It indicates that high yield testing data will required less bits for encoding, which results in higher compression rate as per Equation (5).

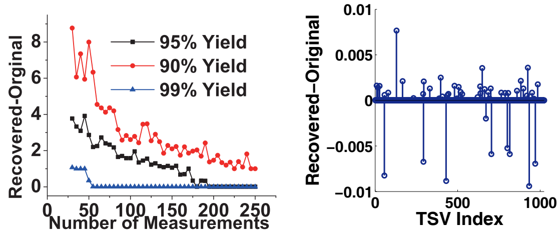


(a)

(b)

Figure 8: (a) Compressed output for yield of 99% (b) compressed output for yield of 95%

Furthermore, the minimum number of required measurements (dimension of compressed testing data) under different yields to achieve a lossless recovery is presented in Fig. 9. As illustrated in Fig. 9(a), the difference between the recovered and original data decreases dramatically as the measurements (M) increases, as expected from the compressive-sensing theory [14] , indicating a least number of measurements are required for lossless compression. As such, the higher the yield is, the less number of measurements (M) is required. For example, when the yield is as high as 99%, there are nearly 60 measurements good enough to fully recover the testing data; whereas, there are nearly 190 measurements needed to fully recover the testing data for yield 95%. Since the sparse solution is unique [14], we can confirm the correctness of reconstruction by performing OMP solver twice. For lossless reconstruction, the maximum difference between the recovered and original data should be sufficiently small. As Fig. 9(b) shows, for 99% yield, the maximum difference between the recovered and original data is as small as E^{-2} for 60 measurements to represent 1024 TSVs, corresponding to a 82.42% data compression rate $(1 - 60 \times 3/1024)$.



(a)

(b)

Figure 9: (a) Maximum difference vs. number of output (b) signal reconstruction with 60 measurements for yield of 99%

5.3 Compressive-sensing in Pre-bond Testing

In Fig. 10, the red square is the defective TSV cluster center and black circle is the defective TSV generated from failure probability and clustering effect based on (4). X-axis and Y-axis represent the location of TSV. The average failure probability is 20 % and due to the clustering effect, the failure probability can be as high as 81.52% for the TSVs close to the center as Fig. 10c. Table 1 shows the output testing data compression rate for different clustering factor α and the number of TSVs. A compression of nearly 89% is achieved for 4096 TSVs with failure rate of 0.5%, but is reduced to nearly 66% with failure rate of 1%. This indicates that more number of measurements is required for the lossless compression when the failure probability increases. It also shows that despite the existence of clustering effect,the compression rate will be maintained almost the same.

In addition, as shown in Table 1, we compare our compression algorithm with length-run (LR) coding and Golomb coding (GLC) based compression algorithms [23]. Note that Golomb coding is greatly affected by the tunable group size GS. For example, if we consider the case of 16384 TSVs with

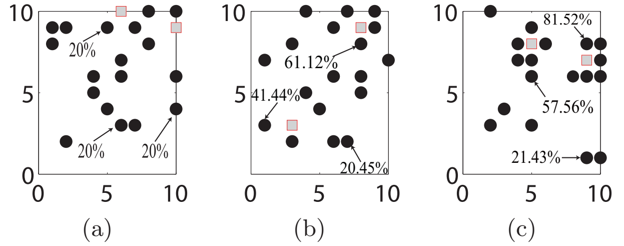


Figure 10: (a) No clustering effect (b) clustering effect with $\alpha = 1$ (c) clustering effect with $\alpha = 2$

failure probability 1% and clustering factor α as 1, our proposed algorithm can successfully compress 64.29%; whereas LR can only compress 46.93%; GLC with $GS = 8, 16, 32$ can only compress 29.49%, 38.37% and 42.03%, respectively. The bandwidth improvement can also be derived from (5) as shown in Table 1. Note that with increasing GS, GLC compression rate of testing-data will converge to the information entropy. However, hardware complexity and decoding time also increase dramatically, which limits us to compare until $GS = 32$.

Table 2: Testing Data Compression in Post-bond Testing

Failure Prob.	Benchmark	Output (bits)	Proposed	LR Coding	GLC GS =8	GLC GS =16
5%	c499	1696	88.15%	78.22%	73.01%	76.42%
	c432	196	88.18%	77.96%	76.38%	78.57%
	c1908	2475	86.89%	73.69%	72.83%	76.34%
	c2670	6300	82.56%	73.81%	69.77%	73.74%
	c3540	1870	87.76%	80.55%	75.75%	78.72%
	c5315	4674	81.80%	75.31%	71.13%	74.97%
	c6288	416	85.65%	81.15%	82.21%	84.13%
	c7552	7992	80.30%	74.27%	68.26%	72.49%
10%	c499	1696	73.82%	59.53%	50.17%	57.17%
	c432	196	82.19%	68.67%	63.06%	66.94%
	c1908	2475	70.57%	56.00%	45.86%	53.27%
	c2670	6300	61.42%	55.17%	44.25%	51.80%
	c3540	1870	70.82%	56.56%	46.24%	53.76%
	c5315	4674	63.71%	52.38%	40.88%	49.14%
	c6288	416	76.06%	58.51%	49.83%	56.49%
	c7552	7992	60.99%	54.45%	43.95%	51.55%

5.4 Compressive-sensing in Post-bond Testing

We finally discuss the post-bond testing data compression. We assume 5% and 10% probabilities of failure IC for an 8-bit output signal (signature), which mean 0.639% and 1.308% failure probabilities for each bit. Similar to pre-bond testing, we compare our proposed compression algorithm with LR and GLC coding based compression algorithms, and is presented in Table 2. It also shows that the data compression rate outperforms length-run coding and Golomb coding. The testing data compression for 5% failure probability varies from 80.03% to 88.18%, whereas 74.27% to 68.26% for LR coding; and 76.42% to 72.49% for GLC with $GS = 16$. For circuit c7552, our proposed algorithm has a 6.03% and 7.81% improvement compared to LR and GLC, respectively. However, as failure probability increases, our proposed testing data compression rate outperforms further by 6.55% and 9.45% compared to LR and GLC with $GS = 16$ respectively. Moreover, since the proposed algorithm is lossless compression, it can also co-work with MISR or other conventional compression techniques to further compressed the testing data with shared circuit implementations.

6. CONCLUSION

In this paper, the testing data compression is discussed for pre-bond and post-bond TSV testing via compressive-sensing based method. By exploring the sparsity of the testing data, one can achieve on-chip data compression and lossless off-chip data recovery. The encoding for compression can be easily implemented on-chip using XOR and AND networks with significantly improved bandwidth for the output of the testing data. As such, it can result in an efficient implementation of online TSV testing vehicle to improve TSV yield with TSV self-repair capability. Experiment results with benchmarks have shown that 89.70% pre-bond data compression rate can be achieved under 0.5% failure probability; and 88.18% post-bond data compression rate can be achieved with 5% failure probability.

7. REFERENCES

[1] J. R. Cubillo et al., "Interconnect design and analysis for through silicon interposers (TSIs)," in *IEEE 3DIC*, 2012.

[2] L. Jiang, Q. Xu, and B. Eklow, "On effective TSV repair for 3D-stacked ICs," in *DATE*. ACM/IEEE, 2012, pp. 793–798.

[3] C. Zhang and et.al., "Novel crack sensor for TSV-based 3D integrated circuits: design and deployment perspectives," in *IEEE ICCAD*, 2013.

[4] F.-W. Chen, H.-L. Ting, and T. Hwang, "Fault-tolerant tsv by using scan-chain test tsv," in *IEEE ASP-DAC*, 2014.

[5] B. Zhang and V. D. Agrawal, "An optimal probing method of pre-bond TSV fault identification in 3D stacked ICs," in *IEEE S3S*, 2014.

[6] E. J. Marinissen, "Challenges and emerging solutions in testing TSV-based 1/2D-and 3D-stacked ICs," in *IEEE DATE*, 2012.

[7] B. Noia and K. Chakrabarty, "Pre-bond probing of TSVs in 3D stacked ICs," in *IEEE ITC*, 2011.

[8] E. J. Marinissen and Y. Zorian, "Testing 3D chips containing through-silicon vias," in *IEEE ITC*, 2009.

[9] J. Xie, Y. Wang, and Y. Xie, "Yield-aware time-efficient testing and self-fixing design for TSV-based 3D ICs," in *IEEE ASPDAC*, 2012.

[10] P. H. Bardell, W. H. McAnney, and J. Savir, *Built-in test for VLSI: pseudorandom techniques*. Wiley-Interscience, 1987.

[11] M. Tahoori, "Defects, yield, and design in sublithographic nano-electronics," in *IEEE Defect and Fault Tolerance in VLSI Systems*, 2005.

[12] W. Maly, "Realistic fault modeling for VLSI testing," in *IEEE DAC*, 1987.

[13] B. Sklar, *Digital communications*. Prentice Hall NJ, 2001, vol. 2.

[14] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via L1 minimization," *Proceedings of the National Academy of Sciences*, 2003.

[15] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Transactions on Information Theory,*, vol. 52, no. 12, pp. 5406–5425, 2006.

[16] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal L1-norm solution is also the sparsest solution," *Communications on pure and applied mathematics*, vol. 59, no. 6, pp. 797–829, 2006.

[17] Y. Zhao and et.al., "Cost-effective TSV grouping for yield improvement of 3D-ICs," in *IEEE ATS*, 2011.

[18] G. S. May and C. J. Spanos, *Fundamentals of semiconductor manufacturing and process control*. John Wiley & Sons, 2006.

[19] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory,*, vol. 53, no. 12, pp. 4655–4666, 2007.

[20] B. Noia, S. Panth, K. Chakrabarty, and S. K. Lim, "Scan test of die logic in 3D ICs using TSV probing," in *IEEE ITC*, 2012.

[21] M. C. Hansen, H. Yalcin, and J. P. Hayes, "Unveiling the ISCAS-85 benchmarks: A case study in reverse engineering," *IEEE Design and Test of Computers*, vol. 16, no. 3, pp. 72–80, 1999.

[22] I. Hamzaoglu and J. H. Patel, "Testset compaction algorithms for combinational circuits," in *IEEE ICCAD*, 1998.

[23] A. Chandra and K. Chakrabarty, "Test data compression for system-on-a-chip using golomb codes," in *IEEE VTS*, 2000.