Cheng YQ, Zhang L, Han YH *et al.* TSV minimization for circuit — Partitioned 3D SoC test wrapper design. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 28(1): 119–128 Jan. 2013. DOI 10.1007/s11390-013-1316-6

TSV Minimization for Circuit — Partitioned 3D SoC Test Wrapper Design

Yuan-Qing Cheng[1,2] (成元庆), *Student Member*, *CCF*, *IEEE*, Lei Zhang[1] (张 磊), *Member*, *CCF*, *IEEE*
Yin-He Han[1,*] (韩银和), *Member*, *CCF*, *ACM*, *IEEE*, and Xiao-Wei Li[1] (李晓维), *Senior Member*, *CCF*, *IEEE*

[1] *State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences Beijing 100190, China*

[2] *Graduate University of Chinese Academy of Sciences, Beijing 100049, China*

E-mail: yuanqing@ieee.org; {zlei, yinhes, lxw}@ict.ac.cn

Received July 3, 2012; revised October 3, 2012.

Abstract Semiconductor technology continues advancing, while global on-chip interconnects do not scale with the same pace as transistors, which has become the major bottleneck for performance and integration of future giga-scale ICs. Three-dimensional (3D) integration has been proposed to sustain Moore's law by incorporating through-silicon vias (TSVs) to integrate different circuit modules in the vertical direction, which is believed to be one of the most promising techniques to tackle the interconnect scaling problem. Due to its unique characteristics, there are many research opportunities, and in this paper we focus on the test wrapper optimization for the individual circuit-partitioned embedded cores within 3D System-on-Chips (SoCs). Firstly, we use existing 2D SoCs algorithms to minimize test time for individual embedded cores. In addition, vertical interconnects, i.e., TSVs that are used to construct the test wrapper should be taken into consideration as well. This is because TSVs typically employ bonding pads to tackle the misalignment problem, and they will occupy significant planar chip area, which may result in routing congestion. In this paper, we propose a series of heuristic algorithms to reduce the number of TSVs used in test wrapper chain construction without affecting test time negatively. It is composed of two steps, i.e., scan chain allocation and functional input/output insertion, both of which can reduce TSV count significantly. Through extensive experimental evaluations, it is shown that the test wrapper chain structure designed by our method can reduce the number of test TSVs dramatically, i.e., as much as 60.5% reductions in comparison with the random method and 26% in comparison with the intuitive method.

Keywords three-dimensional system-on-chip, test wrapper chain, through-silicon vias optimization

1 Introduction

As technology node shrinks endlessly, more and more devices are packed into a single chip but the package technology advances much slowly. These trends lead to interconnection delay dominating gate delay in the Ultra-Deep Sub-Micron realm[1]. Therefore, performance degradation caused by severe interconnect delay becomes unacceptable unless some novel design and fabrication techniques come to aid. On the other hand, due to the huge number of transistors integrated on the chip, the design complexity is becoming more and more intractable. Three-dimensional (3D) integration technology emerges as a promising solution to overcome the severe interconnect delay problem and System-on-Chip (SoC) design paradigm is proposed to alleviate design complexity. 3D SoC, which combines benefits of both, is believed to have the potential to continue Moore's law for few more decades.

As an emerging technology, 3D SoC technology benefits integrated circuits in the following aspects:

• *Integration of Heterogenous Technologies.* By stacking dies vertically, it can integrate ties fabricated with respective optimal technologies without degrading the yield.

• *High Integration Density.* Since chip grows vertically rather than horizontally, high integration density can be achieved without increasing the chip footprint.

• *Improved Interconnect Latency and Energy Efficiency.* The traditional long global interconnects can be replaced by through-silicon vias (TSVs), which are only tens of micrometers long. Therefore, the interconnect latency, power and energy efficiency can be improved

Regular Paper

This work was supported in part by the National Basic Research 973 Program of China under Grant No. 2011CB302503 and the National Natural Science Foundation of China under Grant Nos. 60806014, 61076037, 60906018, 61173006, 60921002, 60831160526.

*Corresponding Author

impressively.

Regarding to the partition granularity, 3D SoC can be classified as technology-level partition, architecture-level partition and circuit-level partition[2]. The former two introduce little modifications to the existing design paradigm whereas the third one can exploit the potentials of 3D technology to the extreme.

Although 3D SoC has many advantages, there are still many road-blocks needed to be cleared before it prevails. Among others, DFT (design for testability) is viewed as a key challenge[3-4]. Depending on the stage when test is applied, 3D SoC testing can be classified as pre-bond testing and post-bond testing. In this paper, we focus on the post-bond test wrapper design for an individual embedded core within a circuit-partitioned 3D SoC.

For traditional 2D SoC testing research, there are already many papers on test wrapper design, such as [5-7]. Test time is treated as the most important metric for optimization. 3D SoC testing has many similarities with 2D testing except that it introduces extra TSV overhead for constructing the testing infrastructure. Therefore, 3D SoC test time optimization can be transformed to a 2D problem by mapping the scan chains and functional inputs/outputs (functional inputs/outputs are latches for I/O purpose) onto a hypothetical plane. However, for 3D SoCs, particularly for those with circuit-partitioned design style, the testing infrastructure — test wrapper chain (a structure connecting functional inputs/outputs and scan chains as a chain for testing purpose) requires TSVs to connect scan chains and functional inputs/outputs on different ties within it. As TSV fabrication is a new technique introduced for 3D ICs, which incurs low fabrication yield and relative large area estate[8], TSV minimization under the optimal test time constraint is an important issue.

Many existing testing researches focus on system-level optimizations, including TAM (Test Access Mechanism) allocations among different IP cores[9], test scheduling[10], etc. However, in our work, we map different layers of a 3D SoC to a hypothetical plane and adopt existing 2D DFT solutions to construct test wrapper chains. The unique problem here is how to connect scan chains on a test wrapper chain within an IP core of a 3D SoC, because a wrapper chain may cross several ties and different connection styles of scan chains would consume different amount of TSV resources. Until now, there are few papers focusing on test wrapper optimization for the individual IP core within a 3D SoC. Among them, Noia *et al.* explored the TSV and test time co-optimization problem for a single IP core of a circuit-partitioned 3D MPSoC in [11]. However, they set TSV count as a constraint and optimized test time under it. As a result, test time cannot be guaranteed to be globally optimal. In our work, we explore the TSV optimization method under the optimal test time constraint since it affects test cost and test temperature directly.

At first, we formulate the problem and divide it into two parts, i.e., scan chain allocation and functional input/output insertion. The first one can be solved by finding the shortest Hamiltonian path in a complete digraph and the second one can be solved by taking the minimum value of a piece-wise function. Subsequently, we propose two heuristic algorithms to attack them respectively. The combination of the two algorithms forms our 3D test wrapper design framework called 3D-TWCD (3D Test Wrapper Chain Design). In order to evaluate the effectiveness of our method, we apply random method and intuitive method as baselines for comparisons (See Sections 3 and 4 for the details). Extensive experimental results show that we can reduce TSV count by 60.5% compared with the random method and by 26% compared with the intuitive method.

The rest of the paper is organized as follows. In Section 2, we present the preliminaries for test wrapper design of circuit-partitioned 3D SoC and motivate our research work through an illustrative example. The TSV optimization problem is formulated in Section 3. The heuristics are also proposed in this section. Experimental results and analysis are presented in Section 4. Related work is presented in Section 5 and Section 6 concludes this paper.

2 Preliminaries & Motivation

2.1 Preliminaries

To facilitate the SoC test, each embedded core is designed with a test wrapper as shown in Fig.1. In the figure, the SoC consists of a 3D circuit-partitioned core and two planar cores. Generally, the test wrapper contains several test wrapper chains according to the TAM bus width. A test wrapper chain consists of function inputs, scan chains and functional outputs[12] as shown in Fig.1. The scan chain can cross several layers due to the circuit-partitioned design style. During the test procedure, test patterns are fed through TAM buses to test wrapper of the CUT (Core Under Test). They act as stimuli to activate the circuit under test. Then, test responses are captured into scan registers. Finally, TAM buses send the responses back to ATE (Automatic Test Equipment) through test pins. The circuit associated with each test wrapper chain can be tested in parallel.

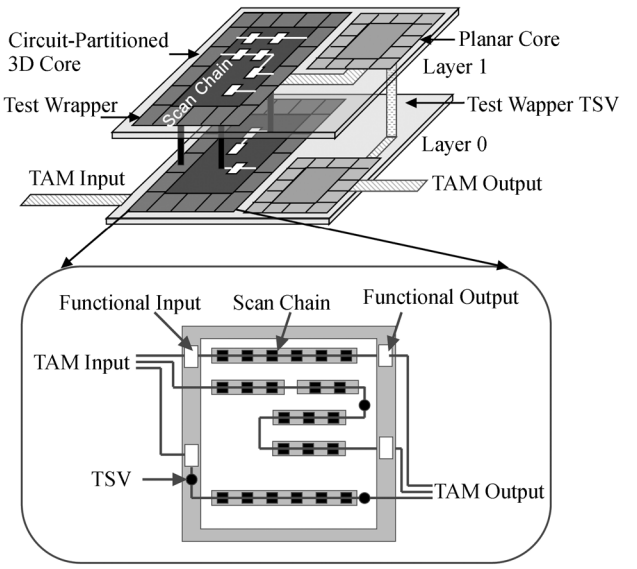


Fig.1. Test wrapper structure of a circuit-partitioned 3D SoC.

2.2 Motivation

In order to reduce the test time of the 3D SoC, we can neglect TSVs at first and map the 3D test wrapper to a plane. Consequently, 3D SoC test time optimization problem can be transformed to a 2D problem. Then, we can take advantage of existing research efforts for 2D SoCs. From [13], we know that test time of the specific embedded core within a SoC can be expressed as follows:

$$T = (1 + \max(S_i, S_o)) \times p + \min(S_i, S_o), \qquad (1)$$

where S_i, S_o denote the scan-in length which is the number of functional inputs plus the total length of scan chains on the test wrapper chain and the scan-out length which is the number of functional outputs plus the total length of scan chains on the test wrapper chain respectively and p is the number of test patterns. (1) indicates that test time is minimized when the maximum scan-in and scan-out lengths are balanced.

Using the LRT (Largest Processing Time) and FFD (First Fit Decreasing) algorithms proposed in [13], we can minimize test time and determine the constitution of each test wrapper chain. Unfortunately, the connection order of components within the test wrapper chain cannot be derived by the algorithms. It can impact the number of TSVs significantly (this point will be illustrated later). Since TSV process is still under development, the yield is much lower than the mature planar CMOS process[8]. In addition, the typical size of TSV is about $5\mu\text{m} \times 5\mu\text{m}$[14]. For the alignment purpose, a metal pad, whose size is larger than the TSV, is attached to the TSV. Therefore, the area occupied by the TSV is much larger than a transistor. In order to improve yield and alleviate routing congestion, the number of TSVs used by the test wrapper should be as small as possible.

To illustrate the optimization potential of test wrapper TSVs, we use an example shown in Fig.2. We assume that the embedded core is implemented with two ties and a single TAM bus is designated to test it. The components (including scan chains and inputs/outputs) within the test wrapper chain are connected in different orders denoted by scheme A and B respectively. Although the test time of scheme A is the same as that of B, B consumes two additional TSVs compared with A (i.e., 1 for A and 3 for B) in the 3D scenario.

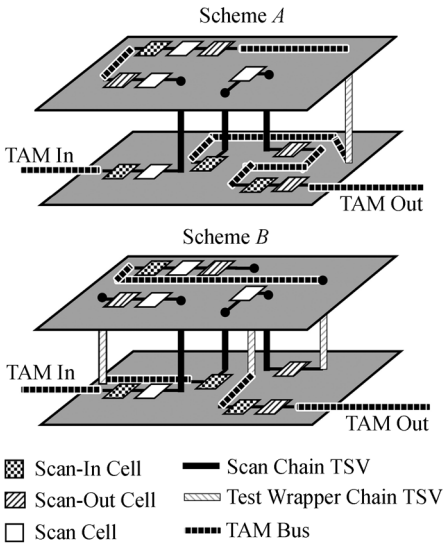


Fig.2. Different test wrapper chain designs consuming different TSVs.

To further evaluate the potential of TSV optimization, we choose core 13 of p93791 from International Test Conference (ITC) 2002 SOC Test Benchmarks① and design it with two layers. The test time is optimized using the algorithm proposed by [13]. The constitution of test wrapper chain is also determined during optimization. Then, we enumerate all possible connection orders of components within the test wrapper chain. TSV consumptions corresponding to different connection orders are plotted in Fig.3. From the figure, it is observed that different test wrapper designs can introduce as large as 3.46 times difference on TSV consumptions although their test time is the same. Therefore, the connection order of components has large potential for optimization.

①http://itc02socbenchm.pratt.duke.edu, Aug. 2012.

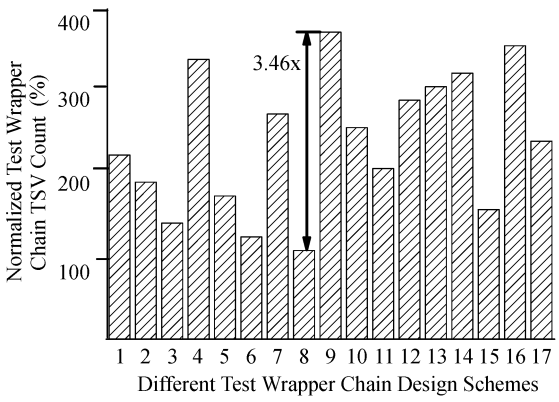


Fig.3. TSV consumptions among different test wrapper designs for core 13 of p93791.

3 TSV Optimizations for Test Wrappers of Circuit-Partitioned 3D SoCs

As stated above, the test wrapper chain consists of functional inputs/outputs and scan chains. According to [13], scan chains are usually considered at first during test time optimization. Then, functional inputs/outputs are considered. Therefore, the TSV optimization procedure is divided into two parts: scan chain allocation and functional input/output insertion.

3.1 Scan Chain Allocation

3.1.1 Problem Formulation

The scan chain allocation problem can be formulated as follows:

Given: the TAM bus width M designated to the target embedded core of the 3D SoC; location information of scan chains (i.e., start layer S_i, end layer E_i and scan chain length L_i);

Constraint:

$$t \leqslant T, \tag{2}$$

where t represents the test time of the embedded core, T represents the optimal test time derived by the algorithm mentioned in Section 2.

The objective is to determine the connection order of scan chains such that TSV count is minimized, i.e.,

$$\text{minimize}: \quad \sum_{i=1}^{M} TSV_{\text{wrapper chain}}(i), \tag{3}$$

where $TSV_{\text{wrapper chain}}(i)$ denotes the number of TSVs used in the i-th test wrapper chain. Further, $TSV_{\text{wrapper chain}}$ can be calculated by the following expression:

$$TSV_{\text{wrapper chain}}(i) = \sum_{k=1}^{u-1} |SI_{i,k+1} - SO_{i,k}|, \tag{4}$$

where SI and SO represent scan-in and scan-out layer of the scan chain. u denotes the number of scan chains within the test wrapper chain. k represents the scan chain location. For example, $TSV_{\text{wrapper chain}}(1) = 1$ according to the scenario shown in Fig.4.

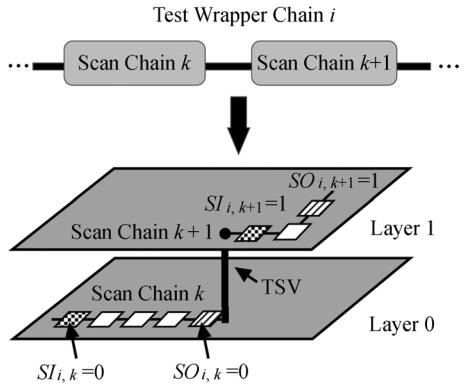


Fig.4. Example for explanation of (4).

3.1.2 Scan Chain Allocation Algorithm

We use an illustrative example to describe our scan chain allocation algorithm.

Illustrative Example. As shown in Fig.5, assume that the core is designed as four ties and contains five scan chains, three functional inputs and one functional output. In the figure, layer 0 denotes the bottom layer where test pins locate. The layers of functional inputs are 0, 1, and 2 respectively. The functional output lies on layer 3. The information of each scan chain is represented by a triple (x, y, z), where x represents the scan-in layer, y represents the scan-out layer and z denotes its length. For the example, they are as follows: $A(0, 1, 32)$, $B(1, 3, 30)$, $C(1, 3, 19)$, $D(3, 2, 20)$ and $E(0, 1, 10)$. Two TAM buses are assigned to test the core (i.e., the number of test wrapper chains is 2). The composition of each test wrapper chain derived by test

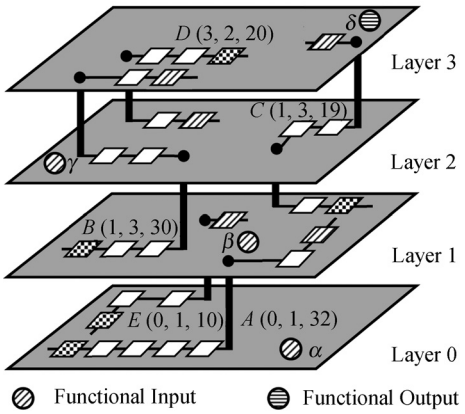


Fig.5. Test wrapper structure as an example.

time optimization is shown in Fig.6. Please note that only scan chains are considered in this step, and the insertion of functional inputs/outputs will be discussed later. In the figure, test wrapper chain #1 consists of scan chains A and C while test wrapper chain #2 consists of scan chains B, D and E.

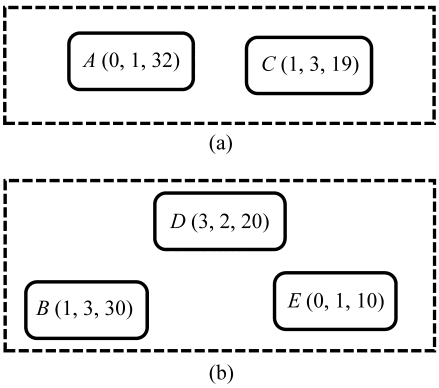


Fig.6. Compositions of test wrapper chains corresponding to the optimal test time. (a) Test wrapper chain #1. (b) Test wrapper chain #2.

Intuitive TSV Count Optimization. Referring to (4), it is intuitive to connect scan chains in such a fashion that the scan-in layer of the latter scan chain is as near as possible to the scan-out layer of the former one. The intuitive method is similar to the No-Look Ahead method proposed in [11]. We use it as the baseline for comparisons. As for the example shown in Fig.5, the connection order of scan chains within test wrapper chain #2 should be E, B, D by the intuitive method.

In some cases, however, the solution derived by the intuitive method may not be optimal. A counter example is shown in Fig.7. In the figure, there are four scan chains and one functional output distributing across three layers. They are stitched together to form a test wrapper chain. The connection order derived using the intuitive method is depicted as scheme A, where scan chain A connects B firstly since the scan-in layer of B is the same as the scan-out layer of A. But if we use the scheme B shown in Fig.7, TSV count can be reduced further (from 6 to 2 in the example). It is due to the fact that the intuitive method only finds the local optimal solution in each step without considering the problem globally. Subsequently, we propose a heuristic algorithm to optimize TSV count globally.

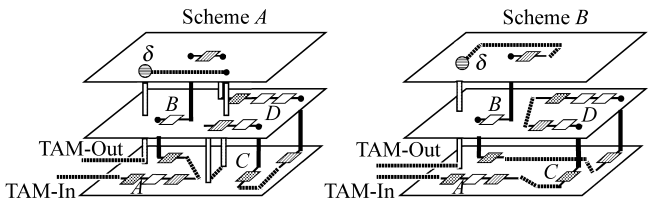


Fig.7. Counter example of the intuitive method[11].

Our Proposed Heuristic Algorithm for Scan Chain Allocation. We can represent scan chains within a test wrapper chain by a complete digraph. Each node in the graph represents a scan chain and each edge denotes TSV consumption when the source node (the node pointed by an arrow tail) connects to the sink node (the node pointed by an arrow head). For example, assume the composition of a test wrapper chain is as shown in Fig.6, the complete digraph corresponding to test wrapper chain #2 is depicted in Fig.8.

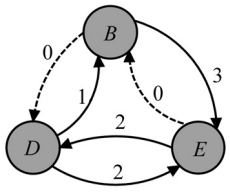


Fig.8. Complete digraph corresponding to test wrapper chain #2.

When the test wrapper chain composition has been derived by test time optimization, we propose the following algorithm to minimize TSV count. Firstly, a complete digraph is constructed corresponding to the test wrapper chain composition as shown in Fig.8 (taking test wrapper chain #2 for example). Secondly, edges are sorted in the descendant order regarding to their weights. Then, the shortest edge is selected and deleted from edge list (edge BD). Two vertices of this edge are marked (B and D). Assume the vertex corresponding to the arrow tail is $V_1(B)$ and the one corresponding to the arrow head is $V_2(D)$. The other edge connecting V_1 and V_2 is deleted from the edge list (edge DB) accordingly since only one edge between the two vertices can be selected. At the same time, edges starting from V_1 and ending to V_2 are also deleted from the edge list (edge BE and ED) since the scan-in/scan-out cell of a scan chain can only connect to one scan chain. Subsequently, the shortest edge in the updated edge list is considered (edge EB). The procedure iterates until a directed path is found, which visits all vertices only once, i.e., the shortest Hamiltonian path in the graph. It is drawn with dashed line in Fig.8. Fig.9 depicts the final connection order of scan chains within test wrapper chain #2. See Fig.10 for the detail of the algorithm. Fig.11 is the pseudo-code of *select* function in Fig.10.

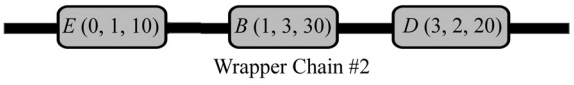


Fig.9. Final connection order of scan chains within test wrapper chain #2.

Algorithm 1

Given the number of test wrapper chains $\mathcal{M}$, scan chain set $\mathcal{SC}$
Sort scan chains in descending order according to their lengths
while ! *Empty*($\mathcal{SC}$) **do**
  Get current shortest wrapper chain *wrapper*[i]
  $\mathcal{SC}_i$ = *Select*($\mathcal{SC}$, *wrapper*[i])
  Insert $\mathcal{SC}_i$ into *wrapper*[i]
  Delete $\mathcal{SC}_i$ from $\mathcal{SC}$
//test wrapper chain reorder procedure
for $i = 0$; $i < \mathcal{M}$; $i++$ **do**
  Construct complete digraph $\mathcal{G}$ for *wrapper*[i]
  $(\mathcal{P}, l)$ = *FindShortestHamiltonianPath*($\mathcal{G}$) //$\mathcal{P}$ is the optimized connection order of test wrapper chain components and l denotes the number of TSVs consumed
  Reorder(*wrapper*[i], $\mathcal{P}$)

Fig.10. Scan chain allocation algorithm.

Algorithm 2

Given scan chain set $\mathcal{SC}$, the i-th wrapper chain
if $\mathcal{SC}_i[0]$ has unique length **then**
  Return $\mathcal{SC}_i[0]$
else //assume n scan chains have the same length
  for $i = 0$; $i < n$; $i++$ **do**
    Construct complete digraph $\mathcal{G}$ for ($\mathcal{SC}_i[i]$ + *wrapper*[i])
    $(\mathcal{P}, l)$ = *FindShortestHamiltonianPath*($\mathcal{G}$)
  Choose $\mathcal{SC}_i$ according to the smallest l
  Return $\mathcal{SC}_i$

Fig.11. *Select* function.

For 2D test wrapper optimization algorithms, they only concern about the scan chain length information. Therefore, if two scan chains have the same length, they are not differentiated. However, for 3D DFT solutions, it will affect TSVs utilized as their scan-in/scan-out layers may be different. Therefore, we must consider scan chains with the same length. In our algorithm, each of them is allocated to the appropriate test wrapper chain for a trial and a complete diagraph is constructed accordingly. The one corresponds to the shortest directed path is selected, which is very similar to the procedure described above except that it is performed in the test time optimization. After scan chain allocation, we consider the functional input/output insertion strategy, which will be described in the next subsection.

3.2 Functional Input/Output Insertion

3.2.1 Problem Formulation

After the optimization for scan chain allocation is performed, functional inputs/outputs are considered. Each functional input/output has length 1. Taking functional input insertion as an example, the problem can be formulated as follows.

Given: the TAM width M, the number of functional inputs P.

Objective: functional input insertion scheme using minimal test wrapper chain TSV count, i.e.,

$$\text{minimize}: \sum_{i=1}^{M}(\alpha_i + \mid \alpha_i - \gamma_i \mid), \tag{5}$$

where

$$\alpha_i = \max_{j \in S_i} Layer(j), \tag{6}$$

γ_i is the scan-in layer of the first scan chain on the i-th test wrapper chain. S_i is the number of functional inputs to be inserted to the i-th test wrapper chain according to the test time optimization algorithm. Thus, α_i is the maximal layer of functional inputs assigned to the i-th test wrapper chain. In addition, the following condition should be met:

$$\sum_{i=1}^{M} S_i = \mathcal{P}. \tag{7}$$

Observe that expression $\alpha_i + \mid \alpha_i - \gamma_i \mid$ in (5) can be rewritten as follows:

$$\alpha_i + \mid \alpha_i - \gamma_i \mid = \begin{cases} 2\alpha_i - \gamma_i, & \text{if } \alpha_i > \gamma_i, \\ \gamma_i, & \text{otherwise.} \end{cases} \tag{8}$$

We plot (8) in Fig.12. As shown in the figure, when $\alpha_i < \gamma_i$, TSV count is always minimal, i.e., γ_i; when $\alpha_i \geqslant \gamma_i$, the layer of the functional input to be inserted should be as near to α_i as possible in order to minimize TSV count.

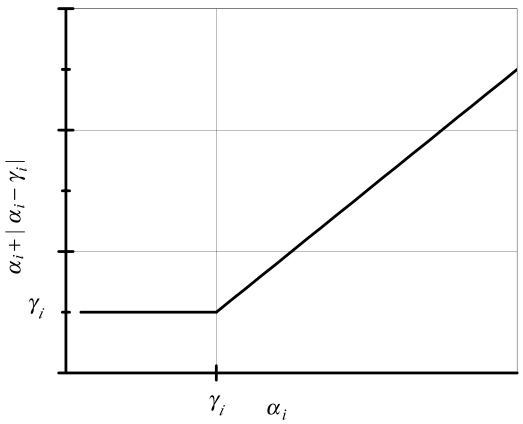


Fig.12. Plot of function $\alpha_i + \mid \alpha_i - \gamma_i \mid$.

3.2.2 Functional Input/Output Insertion Algorithm

The proposed functional input insertion algorithm is described as follows. Firstly, the number of functional inputs assigned to each test wrapper chain is determined by the test time optimization. Then, all

functional inputs are sorted in the ascendant order according to their layer information. Test wrapper chains are also sorted in the ascendant order according to their scan-in layer information (i.e., the scan-in layer of the first scan chain within the test wrapper chain). Insert functional inputs to the first wrapper chain until the number of function inputs equals what should be inserted in the test wrapper chain. Then, the remaining functional inputs are inserted to the next test wrapper chain in the sorting list. The procedure continues until all functional inputs have been inserted. The functional output insertion scheme is similar. The algorithm for functional input/output insertion is described in Fig.13.

Algorithm 3.

1: Given functional input set IN, functional output set OUT and test wrapper chain list $wrapper[1..N]$
2: $[a_i, b_i] = Test\text{-}TimeOptimize(IN, OUT, wrapper[1..N])$ //determine the number of functional inputs/outputs to be inserted to each test wrapper chain. a_i inputs and b_i outputs should be inserted to $wrapper[i]$
3: Sort functional inputs in ascending order according to their layer information //the lower functional input has the higher priority for insertion
4: Sort functional outputs in ascending order according to their layer information
5: Sort $wrapper[1..N]$ in ascending order according to their first scan-in port layer information //the scan-in port with the lower layer should be considered earlier
6: **for** $i = 0; i < N; i++$ **do**
7: Allocate first a_i functional inputs from IN to $wrapper[i]$
8: Delete these inputs from IN
9: Allocate first b_i functional outputs from OUT to $wrapper[i]$
10: Delete these outputs from OUT

Fig.13. Functional input/output insertion algorithm.

Scan chain allocation scheme and functional input/output insertion scheme are combined to form our so-called 3D-TWCD (3D Test Wrapper Chain Design) algorithm. The final structure of test wrapper chain #2 derived by 3D-TWCD is illustrated in Fig.14.

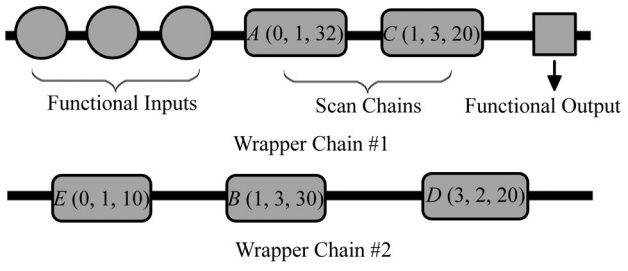


Fig.14. Final test wrapper chain #2 structure of the illustrative example.

4 Experimental Results

4.1 Experimental Setup

To verify the effectiveness of our algorithm, several ITC 2002 SOC Test Benchmarks are chosen for our experiments, including core7 of d281, core13 of p93791, and core 26 of p22810. Since ITC 2002 SOC Test Benchmarks only contain 2D structure specifications without layer information, every wrapper chain component is added with layer information randomly in our experiments as in [11]. The detailed information of every benchmark is listed in Table 1. The last column is the layer count implementing the corresponding benchmark. The algorithm is written in C++ and run on the Linux server configured with Pentium Dual-Core E5300 2.6 GHz CPU and 2 GB memory.

Table 1. Specifications of ITC 2002 SOC Test Benchmarks Used in Our Experiments

Benchmark	Number of Functional Inputs/Outputs	Number of Scan Chains	Layer Count
Core 7 of d281	700/790	20	4
Core 13 of p93791	111/31	46	4
Core 26 of p22810	66/33	31	6

4.2 Evaluation of Test Wrapper Chain TSV Optimization

In the experiments, the random algorithm (Random), which connects test wrapper chain components randomly, and the intuitive algorithm (Intuitive), which mentioned in the former section, are used to compare with our proposed algorithm to validate the effectiveness of our algorithm. Since the optimization procedure of 3D-TWCD is divided into two steps, i.e., scan chain allocation and functional input/output insertion, we evaluate the effects of the two steps separately.

To evaluate the scan chain allocation scheme, we only consider the scan chain reordering and neglect the functional input/output insertion procedure. The comparisons of the three strategies are depicted in Fig.15. In the figure, TSV counts of the intuitive algorithm and 3D-TWCD algorithm are normalized to that of the random method. We also perform experiments under different TAM widths to evaluate the scalability of our algorithm.

From Fig.15, we can see that the intuitive method performs better than the random method in most cases. However, due to its local optimal essence, it cannot grantee the globally optimal solution. In contrast, 3D-TWCD finds the globally optimal solution by the graph theory and thus can reduce TSV overhead significantly. In the figure, the intuitive method reduces 38.8% TSV

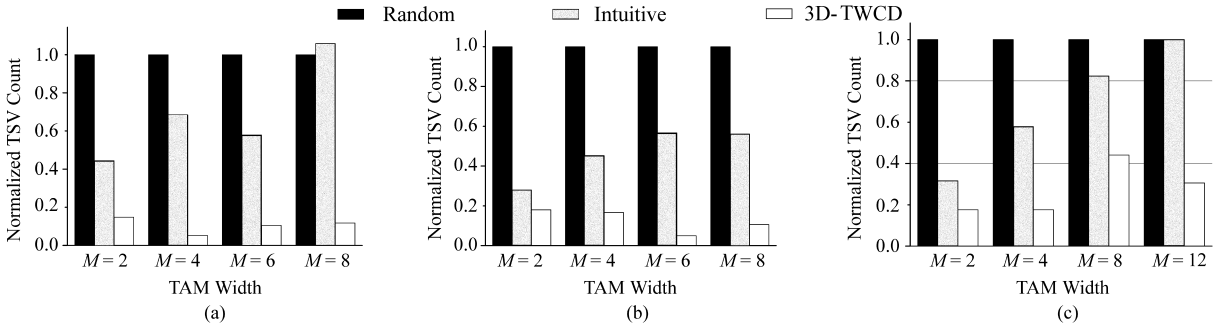


Fig.15. Test wrapper chain TSV count comparisons among three algorithms for (a) core 7 of d281, (b) core 13 of p93791 and (c) core 26 of p22810 when only scan chain optimization is considered.

overhead on average compared with the random method while 3D-TWCD reduces 83.2% TSV overhead compared with the same baseline. Through this experiment, the effectiveness of the scan chain allocation scheme is verified.

Then, we evaluate the effectiveness of functional input/output insertion scheme taking core 13 of p93791 as an example. To compare the three algorithms fairly, we choose the optimal scan chain allocation result obtained by the above experiment as the starting point. Then, different functional input/output insertion strategies are performed based on it. The TSV count comparisons are plotted in Fig.16. In the figure, 3D-TWCD performs best and can save 16% TSVs compared with other two algorithms. The other two benchmarks also show similar trends. Fig.16 indicates that the optimization space of functional input/output insertion is not so large as that of scan chain allocation. The explanations are as the following. Firstly, test patterns and responses must be delivered through test pins located at the bottom layer. Test pins connect to functional inputs/outputs directly, which requires some fixed number of TSVs. Secondly, functional inputs must locate at the beginning of the test wrapper chain while outputs must locate at the end of the test wrapper chain. Thirdly, it is necessary to connect functional inputs/outputs using TSVs if they are on different layers. However, scan chains can cross different layers themselves and may be connected to each other without extra TSVs. The above reasons make the TSV reduction very limited for functional input/output insertion.

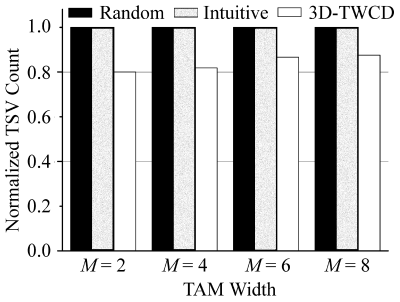


Fig.16. TSV count comparisons for core 13 of p93791 when using the same scan chain allocation scheme.

Next, we combine the two optimization subprocedures (i.e., scan chain allocation and functional input/output insertion) together and evaluate the effectiveness of 3D-TWCD method as a whole. The comparisons are shown in Fig.17. On average, both intuitive method and 3D-TWCD perform much better than the random one while 3D-TWCD performs the best. Taking d281 as an example, TSV count can be reduced by 60.5% for 3D-TWCD compared with the random method while it can only be reduced by 34.5% for the intuitive method. The other two benchmarks also show similar trends.

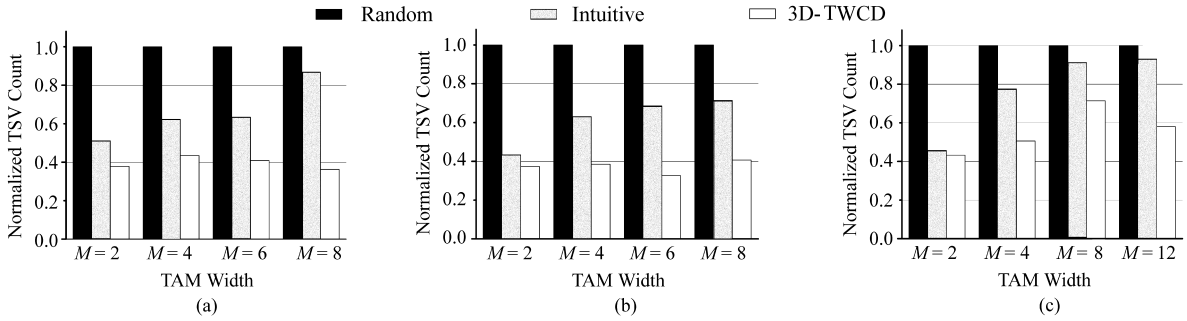


Fig.17. TSV count comparisons among three algorithms for (a) core 7 of d281, (b) core 13 of p93791 and (c) core 26 of p22810.

Both intuitive method and 3D-TWCD consist of two steps, i.e., scan chain allocation and functional input/output insertion. The optimization effect of the first step (scan chain allocation) is partly offset by the second step (functional input/output insertion) as mentioned above. Hence the combined optimization effect shown in Fig.17 is worse than that shown in Fig.15, which considers scan chains only. In addition, as TAM width increases, TSVs consumed by both the random and intuitive method also increase. Whereas, 3D-TWCD algorithm exhibits a relative steady trend, which validates its effectiveness and scalability.

5 Related Work

Due to the limitations of traditional design paradigm when entering the deep sub-micron regime, 3D integration was proposed as a promising technique to continue Moore's law. However, 3D stacking structure also brings several challenges. Among other things frequently used in papers to introduce the research point, test is believed to be one of the most important issues[3-4,15]. The new defect model incurred by the unique fabrication process of 3D ICs was examined in [16]. Test and DFT research in 3D realm can be classified in three categories corresponding to different design styles, i.e., technology-level deign, architecture-level design and circuit-partitioned level design[17]. At every level, the test methodology can be divided into pre-bond test and post-bond test further. At the architecture level, Lewis *et al.* explored the pre-bonding test problem and proposed the scan-island design method in [18]. As for the post-bond design, Marinissen *et al.* proposed a test framework similar to the IEEE 1500 style design to tackle the 3D SoC test issue[19]. Wu *et al.* proposed a scan chain design method for 3D integrated circuits[9] and optimized TAM structure for 3D SoC[20]. Jiang *et al.* optimized the test structure for 3D SoC with routing congestion consideration[21]. They also explored the test optimization under the test pin count constraint[22]. Huang *et al.* proposed the test method for 3D memory[23]. Although architecture level design introduces little modifications to the existing design methods, it cannot fully exploit the benefits of 3D technique. In addition, these researches focus on system-level optimization and do not consider the TSV optimization within each embedded IP core.

To exploit the potentials of 3D ICs to the extreme, circuit-level partitioning method would be adopted, which means that a single core may cross several ties. For circuit-partitioned 3D ICs, Lewis *et al.* explored the design for testability issue in [2]. Zhao *et al.* attacked the clock network design problem for pre-bonding test in 3D ICs[24]. However, test time optimization is not touched in these papers. Noia *et al.* optimized the test time of circuit-partitioned 3D SoC with the constraint of TSV count[11]. Unfortunately, due to TSV limits, the test time cannot be guaranteed to be globally optimal. Roy *et al.* optimized TSV count based on the work of [11] but they only considered the scenario that each scan chain only resides on one particular layer[25]. Different from the above work, we believe that test time is of the primary importance in 3D testing and should be optimized prior to TSV reduction.

6 Conclusions

As 3D technology was introduced to design more sophisticated SoCs, test challenge is becoming more severe than ever before, which requires some new DFT techniques. A method was proposed in this paper to guide the designer to derive an optimal test wrapper design for 3D SoCs. Firstly, due to the significant impact of test time on test cost and thermal issue, we optimized the test time using existing research efforts in 2D field. Then, a heuristic algorithm called 3D-TWCD was proposed to reduce the precious resources — TSVs without affecting test time negatively. Experimental results show that the number of test wrapper TSVs can be reduced dramatically compared with the random method and the intuitive method.

References

[1] Davis J A, Venkatesan R, Kaloyeros A *et al.* Interconnect limits on gigascale integration (GSI) in the 21st century. *Proceedings of the IEEE*, 2001, 89(3): 305-324.

[2] Lewis D L, Lee H H S. Test circuit-partitioned 3D IC designs. In *Proc. ISVLSI*, May 2009, pp.139-144.

[3] Lee H-H S, Chakrabarty K. Test challenges for 3D integrated circuits. *IEEE Design & Test of Computers*, 2009, 26(5): 26-35.

[4] Marinissen E J. Test challenges for 3D-SICs: All the old, most of the recent, and then some new! In *Proc. ITC*, Nov. 2009.

[5] Marinissen E J, Arendsen R, Bos G *et al.* A structured and scalable mechanism for test access to embedded resuable cores. In *Proc. ITC*, Oct. 1998, pp.284-293.

[6] Iyengar V, Chakrabarty K, Marinissen E J. Wrapper/TAM co-optimization, constraint-driven test scheduling, and tester data volume reduction for SOCs. In *Proc. the 39th DAC*, Jun. 2002, pp.685-690.

[7] Huang Y, Reddy S M, Cheng W T *et al.* Optimal core wrapper width selection and SOC test scheduling based on 3-D bin packing algorithm. In *Proc. ITC*, Oct. 2002, pp.74-82.

[8] Loi I, Mitra S, Lee T H, Fujita S, Benini L. A low-overhead fault tolerance scheme for TSV-based 3D network on chip links. In *Proc. ICCAD*, Nov. 2008, pp.598-602.

[9] Wu X, Falkenstern P, Xie Y. Scan chain design for three-dimensional integrated circuits (3D ICs). In *Proc. the 25th Int. Conf. Computer Design*, Oct. 2007, pp.208-214.

[10] Chandran U, Zhao D. Thermal driven test access routing in hyper-interconnected three-dimensional system-on-chip. In *Proc. the 24th Int. Symp. Defect and Fault Tolerance in VLSI Systems*, Oct. 2009, pp.410-418.

[11] Noia B, Charabarty K, Xie Y. Test-wrapper optimization for

embedded cores in TSV-based three-dimensional SOCs. In *Proc. ICCD*, Oct. 2009, pp.70-77.

[12] Iyengar V, Chakrabarty K, Marinissen E J. Test wrapper and test access mechanism co-optimization for system-on-chip. In *Proc. ITC*, Oct. 2001, pp.1023-1032.

[13] Marinissen E J, Goel S K, Lousberg M. Wrapper design for embedded core test. In *Proc. ITC*, Oct. 2000, pp.911-920.

[14] International technology roadmap for semiconductors. http://www.itrs.net/links/2009ITRS/Home2009.htm, July 2012.

[15] Mak T M. Test challenges for 3D circuits. In *Proc. the 12th IOLTS*, Jul. 2006, p.79.

[16] Chen P Y, Wu C W, Kwai D M. On-chip TSV testing for 3D IC before bonding using sense amplification. In *Proc. ATS*, Nov. 2009, pp.450-455.

[17] Lewis D L, Lee H H S. Test strategies for 3D die-stacked integrated circuits. In *Proc. DATE*, Apr. 2009.

[18] Lewis D L, Lee H H S. A scan-island based design enabling pre-bond testability in die-stacked microprocessors. In *Proc. ITC*, Oct. 2007, pp.1-8.

[19] Marinissen E J, Zorian Y. Testing 3D chips containing through-silicon vias. In *Proc. ITC*, Nov. 2009, pp.1-11.

[20] Wu X, Chen Y, Chakrabarty K, Xie Y. Test-access mechanism optimization for core-based three-dimensional SOCs. In *Proc. ICCD*, Oct. 2008, pp.212-218.

[21] Jiang L, Huang L, Xu Q. Test architecture design and optimization for three-dimensional SoCs. In *Proc. DATE*, Apr. 2009, pp.220-225.

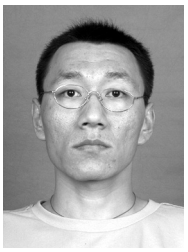
[22] Jiang L, Xu Q, Chakrabarty K, Mak T M. Layout-driven test-architecture design and optimization for 3D SoCs under prebond test-pin-count constraint. In *Proc. ICCAD*, Nov. 2009, pp.191-196.

[23] Huang Y J, Li J F. Testability exploration of 3-D RAMs and CAMs. In *Proc. ATS*, Nov. 2009, pp.397-402.

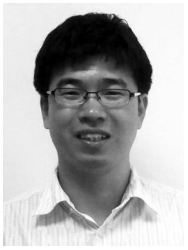
[24] Zhao X, Lewis D L, Lee H H S, Lim S K. Pre-bond testable low-power clock tree design for 3D stacked ICs. In *Proc. ICCAD*, Nov. 2009, pp.184-190.

[25] Roy S K, Ghosh S, Rahaman H, Giri C. Test wrapper design for 3D system-on-chip using optimized number of TSVs. In *Proc. ISED*, Dec. 2010, pp.197-202.

Yuan-Qing Cheng received his Ph.D. degree from Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing. He is currently a post-doc researcher at LIRMM, CNRS, France. His research interests include VLSI design for 3D integrated circuits considering thermal and defect issues, as well as DFT techniques for 3D SoC design.



Lei Zhang received his Ph.D. degree in computer architecture from ICT, CAS, Beijing, in 2008. He is currently an associate professor at the State Key Laboratory of Computer Architecture, ICT, CAS. His research interests include network-on-chip, 3D integration, fault-tolerant computing and multi-core/many-core processors.



Yin-He Han received the M.S. and Ph.D. degrees in computer science from ICT, CAS, in 2003 and 2006, respectively. He is currently an associate professor at ICT, CAS, His research interests include VLSI/SOC interconnection, testing and fault-tolerant. Dr. Han is a recipient of the Best Paper Award at Asian Test Symposium 2003. He is a member of CCF, ACM, IEEE. He was the program co-chair of Workshop of RTL and High Level Testing (WRTLT) in 2009, and serves on the Technical Program Committees of several IEEE and ACM conferences, including ATS, GVLSI, etc.

Xiao-Wei Li received his Ph.D. degree in computer science from ICT, CAS, in 1991. From 1991 to 2000, he was an assistant professor and associate professor (since 1993) in the Department of Computer Science, Peking University, China. He joined ICT, CAS as a professor in 2000. He is now the deputy (executive) director of the State Key Laboratory of Computer Architecture, ICT, CAS. He is a senior member of IEEE. Dr. Li's research interests include VLSI testing, design verification, dependable computing. He has co-published over 200 papers in academic journals and international conference, and holds 34 patents and 35 software copyrights. Dr. Li has served as Chair of CCF Technical Committee on Fault Tolerant Computing since 2008. He has served as IEEE Asian Pacific Regional TTTC (Test Technology Technical Council) Vice Chair since 2004. He has served as the Steering Committee Chair of IEEE Asian Test Symposium (ATS) since 2011. In addition, he serves on the Technical Program Committee of several IEEE and ACM conferences, including VTS, DATE, ASP-DAC, PRDC, etc. He also serves as member of editorial board of JCST, JETTA, JOLPE, etc.