



# LUND UNIVERSITY

## Test Planning for Core-based 3D Stacked ICs with Through-Silicon Vias

Sengupta, Breetta; Ingelsson, Urban; Larsson, Erik

*Published in:*  
[Host publication title missing]

*DOI:*  
[10.1109/VLSID.2012.111](https://doi.org/10.1109/VLSID.2012.111)

2012

[Link to publication](#)

*Citation for published version (APA):*  
Sengupta, B., Ingelsson, U., & Larsson, E. (2012). Test Planning for Core-based 3D Stacked ICs with Through-Silicon Vias. In *[Host publication title missing]* (pp. 442-447). IEEE - Institute of Electrical and Electronics Engineers Inc.. <https://doi.org/10.1109/VLSID.2012.111>

*Total number of authors:*  
3

### General rights

Unless other specific re-use rights are stated the following general rights apply:  
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: <https://creativecommons.org/licenses/>

### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117  
221 00 Lund  
+46 46-222 00 00

# Test Planning for Core-based 3D Stacked ICs with Through-Silicon Vias

Breeta SenGupta      Urban Ingelsson      Erik Larsson  
 Department of Computer and Information Science  
 Linköping University, SE-581 83 Linköping, Sweden  
 Email: {breeta.sengupta, urban.ingelsson, erik.larsson} (at) liu.se

**Abstract**—Test planning for core-based 3D stacked ICs with through-silicon vias (3D TSV-SIC) is different from test planning for non-stacked ICs as the same test schedule cannot be applied both at wafer sort and package test. In this paper, we assume a test flow where each chip is tested individually at wafer sort and jointly at package test. We define cost functions and test planning optimization algorithms for non-stacked ICs, 3D TSV-SICs with two chips and 3D TSV-SICs with an arbitrary number of chips. We have implemented our techniques and experiments show significant reduction of test cost.

**Index Terms**—Test Scheduling, 3D stacked IC, JTAG, Test Architecture, Through Silicon Via.

## I. INTRODUCTION

3D stacked ICs with through-silicon vias (3D TSV-SICs) are emerging and have attracted a fair amount of research [1]–[6]. As the cost of test, which is highly related to test time and the additional design-for-test (DfT) hardware, accounts for a considerable part of the total manufacturing cost, it is important to develop a test plan minimizing the overall test cost. The testing of non-stacked ICs is well-defined; each IC is tested twice during the manufacturing process: during wafer sort, the bare chip (die) is tested, and during package test, the packaged IC is tested. For non-stacked ICs, the same tests are applied to the chip both during wafer sort and package test; hence, the same test schedule is used twice. However, for testing 3D TSV-SICs it is different. First, the test-flow is not well-defined. For 3D TSV-SICs, there are more test alternatives; testing can be performed on each individual IC, partial stacks, and/or the final stack [7]. Second, as the number of tests are different in each of these steps, test schedules are to be developed for each step (each individual IC, partial stacks, and the final stack), which is the focus of this paper.

Much work on test scheduling for non-stacked ICs have been performed [8]–[11]. For example, Chou *et al.* proposed a test scheduling technique that organize the tests in sessions such that the test time is minimized while power constraints are met [9]. Muresan *et al.* [8] proposed a test scheduling technique with the same optimization goal as Chou *et al.* While, the test architecture is unclear in the approach by Muresan *et al.* [8], Iyengar *et al.* [12]–[14] and Marinissen *et al.* [15] proposed test scheduling techniques and test architecture optimization for IEEE 1500. However, no work has addressed test scheduling in an IEEE 1149.1 environment. An increasing amount of work address testing of 3D TSV-SICs [1]–[4], [7], [16], [17].

In our previous work [7], we have defined a cost efficient test flow, while maximizing the yield. The scheme proposes that each individual IC is tested and then the complete stack is tested [7]. Marinissen *et al.* accounted for the variations in hardware required for various test schedules, although the overall test cost has not been optimized [16]. DfT hardware optimization has been addressed in [15], [18]–[20]. However, no work has addressed test scheduling for scan tested core based ICs. And, no work has defined test cost models and test planning algorithms that optimizes the overall test cost for 3D TSV-SICs in an IEEE 1149.1 environment.

In this paper, we assume the test flow that we introduced in our previous work [7], an IEEE 1149.1 environment, and we define test cost functions and test planning optimization algorithms for non-stacked ICs, 3D TSV-SICs with two chips and 3D TSV-SICs with an arbitrary number of chips.

The rest of the paper is organized as follows. In Section II, the JTAG test architecture assumed in our work is detailed. The problem definition is in Section III. In Section IV, we show a motivational example on the test scheduling problem for 3D TSV-SICs. The proposed test scheduling techniques are in Section V. The paper is concluded with experimental results in Section VI and conclusions in Section VII.

## II. TEST ARCHITECTURE

The test architecture of a non-stacked IC, that has been assumed in this paper, is shown in Fig. 1. Here a chip is considered to consist of a number of cores that are accessed by an on-chip JTAG infrastructure [7]. The JTAG test access port (TAP) may have up to five terminals, namely Test Data Input (TDI), Test Data Output (TDO), Test Mode Select (TMS), Test Clock (TCK) and an optional Test Reset (TRST). In Fig. 1 only the TDI and TDO pins are shown, as the test interface terminals. Each core on a chip is accessed by the JTAG TAP via test data registers (TDRs). One TDR may be used to connect multiple cores on a single chip. In Fig. 1, the IC contains three cores: Core1, Core2 and Core3. Core1 and Core2 share a common TDR, while Core3 has an exclusive TDR. Only one TDR can be accessed at a time. Thus, if tests for more than one core of a chip are to be executed concurrently, in a session, as shown in Fig. 2, these cores are to be connected in series on the JTAG interface in one TDR. Since, Core1 and Core2 are tested in the same session as in Fig. 2, denoted by (1, 2), the two cores are connected

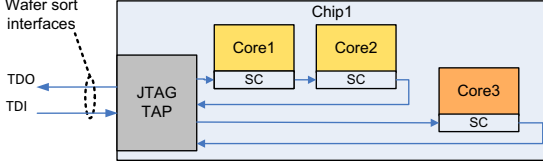


Fig. 1. Test architecture of a non-stacked chip with JTAG

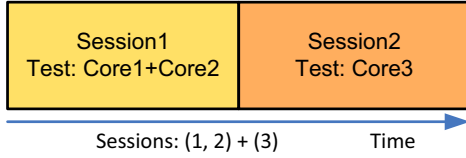


Fig. 2. Sessions formed by core tests

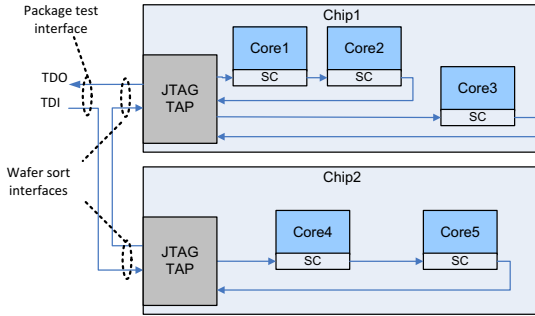


Fig. 3. Test architecture of 3D TSV-SIC with JTAG

to the JTAG TAP by the same TDR, as seen in Fig. 1. Correspondingly, in Session2, only Core3 is tested, denoted by (3) in Fig. 2, which is connected to the JTAG TAP by a single TDR.

During the package test of the 3D TSV-SIC, the TDO of the lower JTAG TAP in the stack serves as the TDI of the corresponding JTAG TAP of the chip on top of it. The TDO of the topmost chip is directed out via TSVs. The TDI of the lowermost chip and the TDO of the topmost chip serve as the package test interfaces as shown in Fig. 3. A session of tests from one chip can be performed concurrently with a session of tests from another chip by selecting the corresponding TDRs by the respective on-chip JTAG TAPs of to the two chips.

### III. PROBLEM DEFINITION

In this section the test cost for non-stacked IC, 3D TSV-SIC with two chips in the stack and 3D TSV-SIC with N chips in the stack, are defined. The overall objective is a test plan with a minimal cost in terms of test application time (TAT) and hardware (number of TDRs), defined as:

$$Cost(TAT, TDR) = \alpha \cdot TAT + \beta \cdot TDR \quad (1)$$

where,  $\alpha$  and  $\beta$  are constants set by the designer depending on the particular system.

#### A. Non-stacked IC

For a non-stacked IC with  $C$  cores, having a test schedule with  $S$  sessions, we assume for a core  $c_{ij}$ ,  $1 \leq i \leq C, 1 \leq j \leq S$ , having a scan chain of length  $l_{ij}$  and requiring  $p_{ij}$  test patterns. The test time for a core  $c_{ij}$  is given by:

$$Time(c_{ij}) = (l_{ij} + \delta) \cdot p_{ij} + l_{ij} \quad (2)$$

where,  $\delta$  accounts for the number of clock cycles required by the JTAG for apply and capture, which is equal to 5.

A test schedule for the  $C$  cores consists of  $S$  sessions, where each core  $c_{ij}$  belongs to an unique session  $s_j$ ,  $1 \leq j \leq S$ . The number of cores that are tested in a session  $s_j$  is given by  $m_j$ . The test time  $t_j$  for a session  $s_j$  is denoted by:

$$t_j = \left( \delta + \sum_{\forall i \in m_j} l_{ij} \right) \cdot \max_{\forall i \in m_j} (p_{ij}) + \sum_{\forall i \in m_j} l_{ij} \quad (3)$$

The overall test time for a test schedule is given as:

$$Time = \sum_{j=1}^S t_j \quad (4)$$

The hardware cost is directly related to the number of sessions, since each session corresponds to a TDR; hence,  $TDR = S$ .

In the case of non-stacked ICs, the same schedule is applied at wafer sort and at package test; hence,  $TAT = 2 \cdot Time$ .

The cost function in Eq.1 is in the case of non-stacked ICs given as:

$$Cost(TAT, TDR) = \alpha \cdot TAT + \beta \cdot TDR \\ = \alpha \cdot 2 \cdot Time + \beta \cdot S \quad (5)$$

The problem is to find a test schedule such that the TAT and the number of TDRs required result in a minimized cost.

#### B. 3D TSV-SIC with two chips in the stack

For a 3D TSV-SIC design having a stack of two chips, Chip1 and Chip2, we assume that Chip1 and Chip2 have  $C_1$  and  $C_2$  cores, respectively. During wafer sort, Chip1 and Chip2 have test schedules with  $S_1$  and  $S_2$  sessions respectively. For each core  $c_{1im}$ ,  $1 \leq i \leq C_1, 1 \leq m \leq S_1$ , in Chip1, the length of the scan chain is  $l_{1im}$  and the number of patterns required is  $p_{1im}$ , while for each core  $c_{2jn}$ ,  $1 \leq j \leq C_2, 1 \leq n \leq S_2$ , in Chip2, the length of the scan chain is  $l_{2jn}$  and the number of patterns required is  $p_{2jn}$ . For wafer sort, Chip1 and Chip2 have test schedules with  $S_1$  and  $S_2$  sessions respectively. Each core  $c_{1im}$  belongs to an unique session  $s_{1m}$ , and each core in Chip2  $c_{2jn}$  belongs to an unique session  $s_{2n}$ . The number of cores that are tested in a session  $s_{1m}$  ( $s_{2n}$ ) is given by  $m_{1m}$  ( $m_{2n}$ ). The test time  $t_{1m}$  for a session  $s_{1m}$  session is denoted by:

$$t_{1m} = \left( \delta + \sum_{\forall i \in m_{1m}} l_{1im} \right) \cdot \max_{\forall i \in m_{1m}} (p_{1im}) + \sum_{\forall i \in m_{1m}} l_{1im} \quad (6)$$

and the test time  $T_{2n}$  for a session  $s_{2n}$  session is denoted by:

$$t_{2n} = \left( \delta + \sum_{\forall j \in m_{2n}} l_{2jn} \right) \cdot \max_{\forall j \in m_{2n}} (p_{2jn}) + \sum_{\forall j \in m_{2n}} l_{2jn} \quad (7)$$

Given Eq.6, the test time for wafer sort for Chip1 is given as:

$$t_{1WS} = \sum_{m=1}^{S_1} t_{1m} \quad (8)$$

and given Eq.7, the test time for wafer sort for Chip2 is given as:

$$t_{2WS} = \sum_{n=1}^{S_2} t_{2n} \quad (9)$$

The total time taken for wafer sort is:

$$t_{WS} = t_{1WS} + t_{2WS} \quad (10)$$

For package test of Chip1 and Chip2 a test schedule with  $S_3$  sessions is formed. Each core  $c_{1im}$  ( $c_{2jn}$ ) belongs to a unique session  $s_{3o}$ ,  $1 \leq o \leq S_3$ . The number of cores that are tested in a session  $s_{3o}$  is given by the set  $m_{3o}$ . The test time  $t_{3o}$  for a session  $s_{3o}$  is denoted by:

$$t_{3o} = \left( \delta + \sum_{\forall i,j \in m_{3o}} (l_{1im} + l_{2jn}) \right) \cdot \max_{\forall i,j \in m_{3o}} (p_{1im}, p_{2jn}) + \sum_{\forall i,j \in m_{3o}} (l_{1im} + l_{2jn}) \quad (11)$$

Given Eq.11, the test time for package test for Chip1 and Chip2 is given as:

$$t_{PT} = \sum_{t=1}^{S_3} t_{3o} \quad (12)$$

The TAT is given by

$$TAT_{2chip} = t_{1WS} + t_{2WS} + t_{PT} \quad (13)$$

The hardware required is the sum of the number of TDRs required during wafer sort of Chip1 and Chip2:

$$TDR = S_1 + S_2 \quad (14)$$

The overall test cost can be expressed by the following equation:

$$\begin{aligned} Cost_{2chip}(TAT, TDR) &= \alpha \cdot TAT + \beta \cdot TDR \\ &= \alpha \cdot TAT_{2chip} + \beta \cdot (S_1 + S_2) \end{aligned} \quad (15)$$

The problem is to find the test schedules for wafer sort of Chip1 and Chip2 individually, and package test for jointly testing Chip1 and Chip2 such that the TAT and the total number of TDRs required by Chip1 and Chip2 during wafer sort result in a minimized cost.

### C. 3D TSV-SIC with $N$ chips in the stack

The cost minimization problem for a 3D TSV-SIC with  $N$  chips forming the stack can be generalized from the two problems stated above. Any chip in the stack  $n_i$ ,  $1 \leq i \leq N$ , has  $C_i$  cores, each denoted by  $c_{ijk}$ ,  $1 \leq j \leq C_i$ ,  $1 \leq k \leq S_i$ , each having a scan chain of length  $l_{ijk}$ , and requiring  $p_{ijk}$  patterns. During wafer sort, the test schedule of a chip  $n_i$  has  $S_i$  sessions, each denoted by  $s_{ik}$ , with  $m_{ik}$  tests in each session. Then, the test time  $t_{ik}$  for a session  $s_{ik}$  is given by

$$t_{ik} = \left( \delta + \sum_{\forall j \in m_{ik}} l_{ijk} \right) \cdot \max_{\forall j \in m_{ik}} (p_{ijk}) + \sum_{\forall j \in m_{ik}} l_{ijk} \quad (16)$$

The time taken by each chip  $n_i$  during wafer sort is

$$t_{iWS} = \sum_{k=1}^{S_i} t_{ik} \quad (17)$$

Thus, the total time taken for wafer sort of the 3D TSV-SIC is

$$t_{N.WS} = \sum_{i=1}^N T_{iWS} = \sum_{i=1}^N \left( \sum_{k=1}^{S_i} t_{ik} \right) \quad (18)$$

For package test of the 3D TSV-SIC, a test schedule is formed with  $S_N$  sessions. Each core  $c_{ij}$  belongs to a unique session  $s_o$ ,  $1 \leq o \leq S_N$ . The number of cores that are tested in a session  $s_o$  is given by  $m_o$ . The test time  $t_o$  is denoted by:

$$t_o = \left( \delta + \sum_{\forall j \in m_o} \sum_{i=1}^N l_{ijo} \right) \cdot \max_{\forall j \in m_o} (p_{ijo}) + \sum_{\forall j \in m_o} \sum_{i=1}^N l_{ijo} \quad (19)$$

Given Eq.19, the test time for package test is given as:

$$t_{N.PT} = \sum_{o=1}^{S_N} t_o \quad (20)$$

Hence, the overall cost is

$$\begin{aligned} Cost_N(TAT, TDR) &= \alpha \cdot TAT + \beta \cdot TDR \\ &= \alpha \cdot t_{N.PT} + \beta \cdot \left( \sum_{\forall i \in N} S_i \right) \end{aligned} \quad (21)$$

The problem is to find the test schedules with  $S_1$  sessions for wafer sort of Chip1,  $S_2$  sessions for wafer sort of Chip2, and  $S_3$  sessions for package test for jointly testing of Chip1 and Chip2 such that the TAT and the total number of TDRs required by all the  $N$  chips during wafer sort result in a minimized cost.

TABLE I  
GIVEN L, P VALUES FOR EACH CORE OF THE 3D TSV-SIC

	Chip 1			Chip 2	
	Core1	Core2	Core3	Core4	Core5
Scan chain length ( $l_{ijk}$ )	50	40	30	20	10
Patterns required ( $p_{ijk}$ )	50	40	30	20	10

TABLE II  
TEST SESSION ALTERNATIVES

Cases	Wafer Sort ( $T_{ws}$ )		Package Test ( $T_{pt}$ )	Total Time	Cost	No. of TDRs
	Chip 1	Chip 2				
1	(1, 2, 3)	(4, 5)	(1, 2, 3)+(4, 5)	14200	15000	2
2	(1, 2, 3)	(4)+(5)	(1, 2, 3)+(4) + (5)	14100	15300	3
3	(1, 2)+(3)	(4, 5)	(1, 2)+(3)+(4, 5)	13300	14500	3
4	(1)+(2)+(3)	(4, 5)	(1)+(2)+(3)+(4, 5)	12900	14500	4
5	(1, 2)+(3)	(4)+(5)	(1, 2)+(3)+(4)+(5)	13200	14800	4
5	(1)+(2)+(3)	(4)+(5)	(1)+(2)+(3)+(4)+(5)	12800	14800	5

#### IV. MOTIVATIONAL EXAMPLE

Here we present an example to demonstrate the variation of cost incurred due to the trade-off between test time and hardware required. Given is a 3D SIC with two chips in the stack, illustrated in Fig. 3. The lengths of the scan chains and the number of patterns required for each core is listed in Table I. We assume that the cost of a single TDR is equivalent to 400 time units.

The time taken for wafer sort,  $t_{WS}$ , for the configuration shown, as in case 3 in Table II, *i.e.*, Core1 and Core2 with a common TDR, forming session  $s_{11}$ , Core3 forming session  $s_{12}$ , Core4 and Core5: session  $s_{21}$  is:

$$\begin{aligned}
 t_{WS} &= t_{11} + t_{12} + t_{21} \\
 &= \max(p_{111}, p_{121}) \cdot (l_{111} + l_{121} + 5) + (l_{111} + l_{121}) \\
 &\quad + (l_{132} + 5) \cdot p_{132} + l_{132} \\
 &\quad + \max(p_{241}, p_{251}) \cdot (l_{241} + l_{251} + 5) + (l_{241} + l_{251}) \\
 &= 50 \cdot 95 + 90 + 30 \cdot 35 + 30 + 20 \cdot 35 + 30 \\
 &= 6650 \text{ time units (t.u.)}
 \end{aligned}$$

Performing the tests in the same order on package test as in wafer sort would result in this case

$$T_{ws} = T_{pt} \quad (22)$$

Therefore the total test time becomes,

$$T = T_{ws} + T_{pt} = 6650 + 6650 = 13300 \text{ t.u.} \quad (23)$$

In this case we require three TDRs for testing the chip. Hence, we can calculate the total test cost from Eq.1:

$$\begin{aligned}
 Cost_{case3} &= \alpha \cdot TAT + \beta \cdot TDR \\
 &= 13300 + 400 \cdot 3 \\
 &= 14500 \text{ units}
 \end{aligned}$$

Similarly, considering separate TDRs for all five cores would give,  $T = 12800 \text{ t.u.}$ , as shown in case6 in Table II. But, the schedule results in more sessions, thus an increased

hardware cost. The total cost incurred in case6 is  $Cost_{case6} = 14800 \text{ units}$ .

The minimum number of sessions is obtained when during wafer sort Core1, Core2 and Core3 are in  $s_{11}$  and Core4 and Core5 are in  $s_{21}$ , while during package test all five cores are in the same session. The total time leads to  $T = 14200 \text{ t.u.}$ , which is significantly higher than the alternative distribution of sessions discussed above. Although, in this case, the hardware requirement is minimum. The overall cost incurred in case1 is  $Cost_{case1} = 15000$ , which is higher than case3 and case6 discussed above.

In case2, where Core1, Core2 and Core3 are tested in session  $s_{11}$ , while Core4 is tested in session  $s_{21}$  and Core5 in session  $s_{22}$ , the cost incurred is  $Cost_{case2} = 15300 \text{ units}$ .

In case4, where Core1, Core2 and Core3 are tested in three different sessions, while Core4 and Core5 are tested in the same session, the total test cost is  $Cost_{case4} = 14500 \text{ units}$ . We can see that the cost incurred in performing case4 is minimum compared to the rest of the five cases in Table II.

Therefore, from the above studies on the distribution of TDRs in a 3D SIC it was seen that the test time can be reduced by increasing the number of TDRs, thereby increasing the number of sessions. Although, an increased number of sessions implies increased hardware cost. Hence, in this paper, we try to obtain a trade-off between the hardware cost and the test time, in order to give the minimum total effective cost.

#### V. PROPOSED APPROACHES

In this section we propose three algorithms, for non-stacked IC, 3D TSV-SIC with two chips in the stack and 3D TSV-SICs with any number of chips in the stack, to arrive at a test plan which requires minimal overall test cost, in terms of TAT and the number of TDR, as defined in Eq.1.

##### A. Non-stacked IC

By the following steps of the algorithm we arrive at the reduced cost for non-stacked ICs.

- Given is the list of  $C$  cores  $c_{ij}$ ,  $1 \leq i \leq C$ ,  $1 \leq j \leq S$ , in a chip, sorted by the number of patterns required  $p_{ij}$ . The length of the scan chains are denoted by  $l_{ij}$ .
- The constants of the cost function defined by Eq.1,  $\alpha$  and  $\beta$  are also provided.
- Initially, TAT is set equal to the test time of core  $c_{11}$ .
- The number of sessions,  $S$  is initially set equal to one. The first session,  $s_1$ , in the test schedule contains the test of core  $c_{11}$ . Core  $c_{11}$  is then removed from the sorted list.
- Each core  $c_{ij}$ , remaining in the sorted list, is descended in the following way:

The increase in TAT for each core  $c_{ij}$  is calculated by including it in all existing sessions. If the cost of a single TDR is less than the cost incurred by including the core test in any of the existing sessions due to the increased test time, the core test forms a new session.

Once the core is assigned a session, it is excluded from the sorted list.

TABLE III  
TAT AND TDR FOR NON-STACKED IC

No.	Design	Minimal Test Cost			Cost with Maximum TDR (= No. of cores)				Cost with Minimum TDR (= 1)			
		Cost	TAT	TDR	Cost	TAT	TDR	Cost	Inc.(%)	TAT	TDR	Cost
1	<b>p22810</b>	501490	7	534250	474489	22	577449	8.1	2022377	1	2027057	279.4
2	<b>p93791</b>	614233	4	633701	589394	13	652665	3.0	1990806	1	1995673	214.9
3	<b>g1023</b>	46885	4	51813	42429	12	57213	10.4	137727	1	138959	168.2
4	<b>d695</b>	35757	4	40689	34331	8	44195	8.6	80369	1	81602	100.6
5	<b>h953</b>	271381	2	305483	230771	7	350128	14.6	418607	1	435658	42.6
6	<b>d281</b>	117946	2	144992	97310	5	164925	13.8	186458	1	199981	37.9

- The test plan is achieved when test of each core  $c_{ij}$ , has been assigned its respective session  $s_j$ .

### B. 3D TSV-SIC with two chips in the stack

The wafer sort test schedules for the two chips forming the 3D TSV-SIC, Chip1 and Chip2 are obtained by applying the algorithm for test scheduling of non-stacked ICs. The test planning algorithm for package test is discussed below:

- Given is the list of sessions  $S_1$  of Chip1 and sessions  $S_2$  of Chip2, denoted by  $s_{1m}$  and  $s_{2n}$  respectively. The lists of sessions of Chip1 and Chip2,  $s_{1m}$  and  $s_{2n}$ , are sorted in descending order of their test times,  $t_{1m}$  and  $t_{2n}$ .
- The test schedule for the package test is obtained by simultaneously initiating the sessions  $s_{1m}$  and  $s_{2n}$  for all  $m = n$ . The total number of sessions during package test is  $S_1$  if  $S_1 > S_2$ , and  $S_2$  otherwise.
- The reduction in test time for each new session formed during package test of the two chip 3D TSV-SIC is the test time of the session  $s_{1m}$ , if  $s_{1m} < s_{2n}$  and  $s_{2n}$  otherwise. The sum of the reduction in test time over all the sessions formed during package test gives the overall reduction in the TAT.

### C. 3D TSV-SIC with N chips in the stack

The algorithm used for scheduling tests for 3D TSV-SICs with two chips in the stack can be extended for 3D TSV-SICs with N chips in the stack.

- Given is the list of sessions  $S_i$  of each chip  $n_i$ , each denoted by  $s_{ik}$ ,  $1 \leq k \leq S_i$ . All the sessions of each chip  $n_i$  are sorted in descending order of their test times.
- The test schedule for the package test is obtained by simultaneously initiating the  $k^{th}$  session,  $S_{ik}$  of each chip  $n_i$ ,  $\forall i \in (1toN)$ . The total number of sessions during the package test of the 3D TSV-SIC with N chips in the stack is  $\max(S_i)$  and the time taken by each session is  $\max(t_{ik})$ ,  $1 \leq k \leq \max(S_i)$ .

## VI. EXPERIMENTAL RESULTS

In this section we illustrate the benefits of the proposed approach on the three configurations described earlier, namely,

TABLE IV  
REDUCTION IN TAT FOR 3D TSV-SIC FOR 2, 3 AND 4 CHIPS

No. of chips	Design nos.	Cost Naive Approach	Cost Reduced	Percentage Reduction
2	<b>1,2</b>	4029631	1185943	70.57
	<b>2,3</b>	2143565	730389	65.93
	<b>3,4</b>	222748	94954	57.37
	<b>4,5</b>	503976	284600	43.53
	<b>5,6</b>	649111	475849	26.69
3	<b>1,2,3</b>	3298487	1951685	40.83
	<b>2,3,4</b>	2605603	1292321	50.40
	<b>3,4,5</b>	1759284	737453	58.08
	<b>4,5,6</b>	2925324	727092	75.15
4	<b>1,2,3,4</b>	3897924	1951685	49.93
	<b>2,3,4,5</b>	3051605	1292321	57.65
	<b>3,4,5,6</b>	3639169	737453	79.73

non-stacked IC, 3D TSV-SIC with two chips in the stack and 3D TSV-SIC with N chips in the stack.

Experiments have been performed on the six ITC'02 benchmark system on chip (SOC) designs mentioned below: p22810, p93791, g1023, d695, h953 and d281.

The following assumptions were made when constructing 3D TSV-SICs from the non-stacked SOC benchmarks :

- The modules in the benchmark SOC designs are projected as cores in a non-stacked IC
- All scan elements (inputs, outputs, and scan cells) at a core are connected to a single scan-chain
- 3D TSV-SICs are constructed by vertically stacking any number of the non-stacked designs
- The constant  $\alpha$  in Eq.1 for all designs is considered to be one
- The constant  $\beta$  in Eq.1 for all designs is calculated by dividing the test time of the first core in the sorted list,  $Time(c_1)$ , by the number of cores,  $C$ .

### A. Non-stacked IC

Table III compares the minimized overall cost for non-stacked ICs to the overall cost when the test time cost is minimal and to the overall cost when the cost of hardware is minimum, *i.e.*, there is only one TDR. In Table III, each row corresponds to a SOC benchmark design, which is shown in the second column. The costs of three different test schedules are compared in the following three groups of columns. The first group of three columns shows the minimal test cost of the respective designs as obtained by the algorithm proposed in Section V. Next is the cost incurred when the TAT is minimum; in other words the hardware cost is maximum,

with the number of TDRs equal to the number of cores in the IC. The last column in the group of four columns evaluates the increase in the test cost *wrt* the minimal test cost. The rightmost group of four columns shows the test cost when all cores share a common TDR, thereby maximizing TAT. In Table III, it can be seen that the maximum reduction in cost *wrt* minimized TAT is up to 15% for h953 and *wrt* minimized number of TDRs is up to 280% for p22810.

In Table IV, the package test cost for various 3D TSV-SIC designs made by stacking the six benchmark designs in Table III are shown. The number of chips that have been stacked to make the 3D TSV-SIC is shown in the leftmost column. The group of five rows have 3D TSV-SICs with two chips in the stack, followed by a group of four rows having three chips in the stack and the group of three rows at the bottom are designs made by stacking four chips. The second column from left shows the benchmark designs that have been used to make the stack, which correspond to the serial number used in Table III. For instance, the first 3D TSV-SIC design contains two chips in the stack, 1 and 2, which refers to p22810 and p93791 respectively. The third column lists the test times obtained by summing up the test times, of each design forming the stack, corresponding to the minimal cost, as obtained in Table III. The next column shows the reduced test time by applying the algorithm proposed in Section V. In the rightmost column, the relative reduction in the test time is evaluated. We can see that the test time can reduce up to 75%, when chips g1023, d695, h953 and d281 are stacked.

## VII. CONCLUSION

In this paper, we define test cost as a function of TAT and the number of TDRs for non-stacked ICs, 3D TSV-SIC with two chips in the stack and 3D TSV-SIC with N chips in the stack. The test cost is minimized by co-optimizing TAT and the number of TDRs. We propose an algorithm for scheduling tests, which addresses the following three problems:

- 1) For a non-stacked IC, in an IEEE 1149.1 environment, where the same test schedule is applied during wafer sort and package tests, the tests of all the cores are grouped in sessions such that the cost is minimized by co-optimizing the TAT and the number of TDRs required. We find that the cost can increase by 280%, when either one of the variables are minimized.
- 2) For a 3D TSV-SIC, having two chips, each chip is tested individually during wafer sort and jointly during package test. The cost is minimized by forming sessions from different chips concurrently during the package test. Results show that by applying the algorithm, the test time can be reduced by up to 70%.
- 3) The algorithm for test scheduling of 3D TSV-SICs with two chips is extended to 3D TSV-SICs with any number of chips forming the stack. Experimental results show significant reductions in the overall test cost. The reduction in test time is up to 75%.

## REFERENCES

- [1] E. J. Marinissen and Y. Zorian, "Testing 3D Chips Containing Through-Silicon Vias," in *IEEE International Test Conference (ITC)*, 2009, pp. 1–11.

- [2] H.-H. S. Lee and K. Chakrabarty, "Test Challenges for 3D Integrated Circuits," in *IEEE Design and Test of Computers, Special Issue on 3D IC Design and Test*, Oct. 2009, pp. 26–35.
- [3] D. L. Lewis and H.-H. S. Lee, "A Scan-Island Based Design Enabling Pre-bond Testability in Die-Stacked Microprocessors," in *IEEE International Test Conference (ITC)*, 2007, pp. 1–8.
- [4] X. Wu, P. Falkenstern, and Y. Xie, "Scan Chain Design for Three-Dimensional Integrated Circuits (3D ICs)," in *International Conference on Computer Design (ICCD)*, 2007, pp. 208–214.
- [5] Y.-J. Lee and S. K. Lim, "Co-Optimization of Signal, Power, and Thermal Distribution Networks for 3D ICs," in *Electrical Design of Advanced Packaging and Systems Symposium*, 2008, pp. 163–166.
- [6] R. Weerasekera, "System Interconnection Design Trade-offs in Three-Dimensional (3-D) Integrated Circuits," in *KTH Information and Communication Technology*, 2008.
- [7] B. SenGupta, U. Ingelsson, and E. Larsson, "Scheduling Tests for 3D Stacked Chips under Power Constraints," in *Accepted to be published in Journal of Electronic Testing: Theory and Applications (JETTA)*, 2011.
- [8] V. Muresan, X. Wang, V. Muresan, and M. Vladutiu, "Greedy Tree Growing Heuristics on Block-Test Scheduling Under Power Constraints," in *Journal of Electronic Testing: Theory and Applications*, 2004, pp. 61–78.
- [9] R. M. Chou, K. K. Saluja, and V. D. Agrawal, "Scheduling tests for VLSI systems under power constraints," in *IEEE Transactions on VLSI Systems*, vol. 5, no. 2, Jun. 1997, pp. 175–185.
- [10] Y. Zorian, "A Distributed BIST Control Scheme for Complex VLSI devices," in *IEEE VLSI Test Symposium (VTS)*, Apr. 1993, pp. 6–11.
- [11] E. Larsson and Z. Peng, "An Integrated Framework for the Design and Optimization of SOC Test Solutions," in *Journal of Electronic Testing: Theory and Applications, Special Issue on Plug-and-Play Test Automation for System-on-a-Chip*, vol. 18, no. 4, Aug. 2002, pp. 385–400.
- [12] V. Iyengar, K. Chakrabarty, and E. J. Marinissen, "Wrapper/TAM Co-Optimization, Constraint-Driven Test Scheduling, and Tester Data Volume Reduction for SOCs," in *IEEE VLSI Test Symposium (VTS)*, no. 44.3, Jun. 2002, pp. 685–690.
- [13] —, "Test Access Mechanism Optimization, Test Scheduling, and Tester Data Volume Reduction for System-on-Chip," in *IEEE Transactions on Computers*, vol. 52, no. 12, Dec. 2003, pp. 1619–1632.
- [14] —, "Test Wrapper and Test Access Mechanism Co-Optimization for System-on-Chip," in *Journal of Electronic Testing: Theory and Applications*, vol. 18, 2002, pp. 213–230.
- [15] E. J. Marinissen, R. Kapur, M. Lousberg, T. McLaurin, M. Richetti, and Y. Zorian, "On IEEE P1500s Standard for Embedded Core Test," in *Journal of Electronic Testing: Theory and Applications (JETTA)*, vol. 18, 2002, pp. 365–383.
- [16] E. J. Marinissen, J. Verbree, and M. Konijnenburg, "A Structured and Scalable Test Access Architecture for TSV-Based 3D Stacked ICs," in *IEEE VLSI Test Symposium (VTS)*, Apr. 2010, pp. 1–6.
- [17] B. Noia, S. K. Goel, K. Chakrabarty, E. J. Marinissen, and J. Verbree, "Test-Architecture Optimization for TSV-Based 3D Stacked ICs," in *IEEE European Test Symposium (ETS)*, May 2010, pp. 24–29.
- [18] E. J. Marinissen, K. Chakrabarty, and V. Iyengar, "A Set of Benchmarks for Modular Testing of SOCs," in *International Test Conference (ITC)*, no. 19.1, 2002, pp. 519–528.
- [19] L.-T. Wang, C.-W. Wu, C.-W. Wu, and X. Wen, "VLSI test principles and architectures: design for testability," in *Academic Press*, 2006.
- [20] S. Goel, "Test-Access Planning and Test Scheduling for Embedded Core-Based System Chips," in *University Press, Eindhoven, The Netherlands*, 2005.