Test-Architecture Optimization for TSV-Based 3D Stacked ICs*

Brandon Noia[1], Sandeep Kumar Goel[1], Krishnendu Chakrabarty[1], Erik Jan Marinissen[2], and Jouke Verbree[3]

[1]Dept. Electrical and Computer Engineering
Duke University
Durham, NC 27708, USA
{brn2, skgoel, krish}@ee.duke.edu

[2] IMEC
Kapeldreef 75
B-3001 Leuven, Belgium
erik.jan.marinissen@imec.be

[3]Dept. of Computer Engineering
Delft University of Technology
Mekelweg 4, 2628CD Delft, The Netherlands
j.verbree@student.tudelft.nl

Abstract

Testing of 3D stacked ICs (SICs) is becoming increasingly important in the semiconductor industry. In this paper, we address the problem of test architecture optimization for 3D stacked ICs implemented using Through-Silicon Vias (TSVs) technology. We consider 3D-SICs with both fixed given and yet-to-be-designed test architectures on each die and show that both corresponding problem variants are $\mathcal{NP}$-hard. We next present mathematical programming techniques to derive optimal solutions for these problems. Experimental results for three handcrafted 3D-SICs of various SOCs from the ITC'02 SOC test benchmarks show that compared to the baseline method of sequentially testing all dies in a stack, the proposed solutions can achieve up to a 57% reduction in test time. We also show that increasing the number of test pins provides a greater reduction in test time compared to an increase in the number of TSVs. Furthermore, it is shown that 3D stacks with large and complex dies at lower layers require less test time than stacks with complex dies at higher layers.

1 Introduction

The semiconductor industry is pushing relentlessly for high-performance and low-power chips. Recent advances in semiconductor manufacturing technology have enabled the creation of complete systems with direct stacking and bonding of die-on-die. These system chips are commonly referred to as three-dimensional (3D) stacked ICs (SICs). In this work, we focus on 3D-SICs implemented using Through-Silicon Via (TSV) vertical interconnects. Using this technology, 3D-SICs are created by attaching multiple device layers to each other through wafer or die stacking, and connecting metal layers between dies using vertical TSVs [5]. Compared to traditional two-dimensional SOCs, 3D-SICs provide greater design flexibility [1, 2], higher on-chip data bandwidth, reduction in average interconnect length, and alleviation of problems associated with long global interconnects [3, 4, 5].

Testing core-based dies in 3D-SICs brings forward new challenges [6, 7]. In order to test the dies and associated cores, a Test Access Mechanism (TAM) must be included on the dies to transport test data to the cores, and a 3D TAM is needed to transfer test data to the dies from the stack input/output pins. TAM design in 3D-SICs involves additional challenges compared to TAM design for 2D SOCs. In a 3D-SIC, a test architecture must be able to support testing of individual dies as well as testing of partial and complete stacks [7]. Furthermore, test architecture optimization must not only minimize the test time (test length), but it also needs to minimize the number of TSVs used to route the 3D TAM; as each TSV has area costs associated with it and is a potential source of defects in an 3D-SIC.

In this paper, we address the problem of test-architecture optimization of 3D-SICs with (1) *hard dies*, in which a test architecture already exists, and (2) *soft dies*, for which we also design the test architecture for each die. In addition to minimizing the test time for each soft die, we minimize the test time for the complete stack in both problem instances. While it is theoretically possible to have multiple dies on a given layer in a stack, we only consider one die per layer in a stack. Also, a core is considered to be part of a single die only, i.e., we do not consider '3D cores' as they are not likely in the immediate future of 3D-SICs.

The rest of the paper is organized as follows. Section 2 provides an overview of the related prior work. Section 3 uses a simple example to motivate this work and formally describes the two problems addressed in this paper. Section 4 presents integer linear programming (ILP) models to solve the test-architecture optimization problems described in Section 3. Section 5 presents experimental results for various 3D-SICs constructed using SOCs from the ITC'02 SOC test benchmarks [8]. Finally, Section 6 concludes the paper.

2 Related Prior Work

Testing of 2D SOCs and the optimization of related test-access architectures have been well studied [9, 10, 11, 12]. Optimization methods have included integer linear programming (ILP) [9], rectangle packing [9, 13], iterative refinement [11], and other heuristics [12, 14]. However, all these methods were originally developed for 2D SOCs, and the added test complexities related to 3D technology were not considered.

Recently, some early work has been reported on testing of 3D-SICs. Heuristic methods for designing core wrappers in 3D-SICs were developed in [15]. ILP models for test architecture design for each die in a stack is presented in [16]. While these ILP models take into account some of the constraints related to 3D-SIC testing such as a TSV limit, this approach does not consider the reuse of die-level TAMs. A TAM wire-length minimization technique based on simulated annealing is presented in [17]. Heuristic methods for reducing weighted test cost while taking into account the constraints on test pin widths in pre-bond and post-bond tests are described in [18]. In most prior work on 3D-SIC testing, TAM optimization is performed at die-level only, which leads to inefficient TAMs and non-optimal test schedules for partial/complete stack test. Furthermore, all previous methods assume that the designer can create TAM architectures on each die during optimization, which may not be possible in all cases. This paper considers test-architecture optimization for the entire stack and considers 3D stacks with both hard and soft dies. We also explore the effect of available test pins and TSVs used on TAM design and test scheduling.

*The work of Brandon Noia was supported in part by a Master's Scholarship from the Semiconductor Research Corporation (SRC).

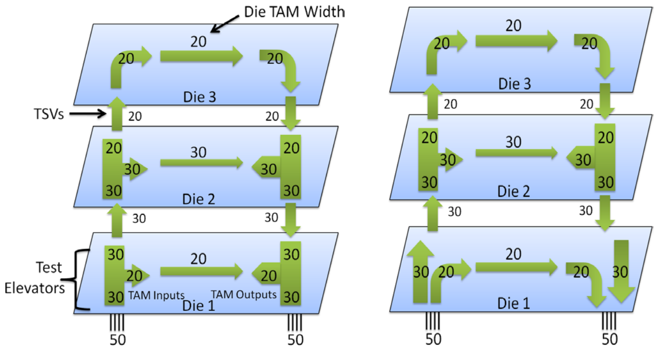
978-1-4244-5833-2/10/$26.00 ©2010 IEEE

3 Problem Definition

In a 3D-SIC, the lowest die is usually directly connected to chip I/O pins, therefore it can be tested using test pins. To test the non-bottom dies in the stack, test data must enter through the test pins on the lowest die. Therefore, to test other dies in the stack, the test access mechanism (TAM) must be extended to all die in the stack through the test pins at the lowest die. To transport test data up and down the stack, "TestElevators" [19] need to be included on each die except for the highest die in the stack [7]. The number of test pins and TestElevators as well as the number of TSVs used affect the total test time for the stack. Currently, stacks consist of anywhere from two to eight dies.

Consider a simple 3D-SIC with three dies with given test access architectures as shown in Figure 1. Suppose the test times for Die 1, Die 2, and Die 3 are 300, 800, and 600 clock cycles respectively. The total number of available test pins at the bottom die is 100. Die 1 requires 40 test pins (TAM width of 20), and dies 2 and 3 require 60 TestElevators and 40 TestElevators, respectively. The test time for each die is determined by its test architecture.

?



(a) (b)

Figure 1: Example 3D-SIC with three hard dies.

doubt

Figure 1(a) shows the TestElevator widths and the number of TSVs used if all dies are tested serially. In this case, a total of 100 TSVs are used and only 60 out of the available 100 test pins are necessary. The total test time for the stack is the sum of the test times of the individual dies, i.e., 1700 cycles. Figure 1(b) shows the test architecture required if Die 1 and Die 2 are tested in parallel. In this case, the number of TSVs used is the same as in Figure 1(a). However, all 100 test pins are required to test Die 1 and Die 2 in parallel. Also, 60 TestElevators must pass between Die 1 and Die 2 in order to pass a separate 30-bit wide TAM to Die 2 for parallel testing. For this case, the total test time for the stack is 600 + max$\{800, 300\}$ = 1400 cycles (we assume session-based test scheduling [20]). This example clearly shows that there is a trade-off between test time and the number of test pins and TSVs used. Therefore, a test-architecture optimization algorithm for 3D-SICs has to minimize the test time while taking into account upper limits on the number of test pins and TSVs used.

The problem of test-architecture optimization for 3D-SICs with hard dies can be defined as follows.

Problem 1 [3D-SIC with Hard Dies (PSHD)]
Given a stack with a set M of dies, total number of test pins W_{max} available for test, and a maximum number of TSVs (TSV_{max}) that can be used for TAM design. For each die $m \in M$, the die's number corresponds to its teir in the stack (die 1 is the bottom die, die 2 is next, and so forth), and we are given the number of test pins on the bottom die w_m ($w_m \leq W_{max}$) required to test the die, and the associated test time t_m (since the test architecture per die is given, t_m is also given). Determine an optimal TAM design and test schedule for the stack such that the total test time T for the stack is minimized and the number of TSVs used does not exceed TSV_{max}. □

The problem statement is different for a 3D-SIC with soft dies. In the case of soft dies, the test architecture for each die is not pre-defined, but it is determined during the test-architecture design for the stack. This scenario provides greater flexibility in terms of test time optimization. The problem of test-architecture optimization for 3D-SICs with soft dies can be formally defined as follows.

Problem 2 [3D-SIC with Soft Dies (PSSD)]
Given a stack with a set M of dies, the total number of test pins W_{max} available for test at the lowest die, and a maximum number of TSVs (TSV_{max}) that can be used for TAM design. For each die $m \in M$, we are given the total number of cores c_m. Furthermore, for each core n, the number of inputs i_n, outputs o_n, total number of test patterns p_n, total number of scan chains s_n, and for each scan chain k, the length of the scan chain in flip flops $l_{n,k}$ are given. Determine an optimal TAM design and test schedule for the stack, as well as for each die, such that the total test time T for the stack is minimized and the number of TSVs used does not exceed TSV_{max}. □

Both Problem 1 and Problem 2 are $\mathcal{NP}$-hard ("proof by restriction"), as they can be reduced using standard techniques to the rectangle packing problem, which is known to be $\mathcal{NP}$-hard [21]. For example, for PSHD, if we remove the constraints related to maximum number of TSVs, each die can be represented as a rectangle with width equal to its test time and height equal to the number of required test pins. Now we need to pack all these rectangles (dies) into a bin with width equal to the total number of test pins and height equal to the total test time for the stack, which needs to be minimized. Similarly, for PSSD, a rectangle must also be selected for each die from a set of rectangles with different widths and heights, but a special case of the scenario is identical to PSHD. Despite the $\mathcal{NP}$-hard nature of these problems, they can be solved optimally since the number of layers in a 3D-SIC is expected to be limited, e.g., up to four layers have been predicted for logic stacks [22].

4 Test-Architecture Optimization

In this paper, we use integer linear programming (ILP) to solve the problems defined in the previous section. Although ILP methods do not scale well with problem instance size, the problem instance sizes for P_{SHD} and P_{SSD} are relatively small for realistic stacks, therefore ILP methods are good candidates for solving them.

4.1 ILP Formulation for PSHD

To create an ILP model for this problem, we need to define the set of variables and constraints. We first define a binary variable x_{ij}, which is equal to 1 if die i is tested in parallel with die j, and 0 otherwise.

Constraints on variable x_{ij} can be defined as follows:

$$x_{ii} = 1 \qquad \forall i \tag{4.1}$$

$$x_{ij} = x_{ji} \qquad \forall i, j \tag{4.2}$$

$$1 - x_{ij} \geq x_{ik} - x_{jk} \geq x_{ij} - 1 \qquad \forall i \neq j \neq k \tag{4.3}$$

The first constraint indicates that every die is always considered to be tested with itself. The second constraint states that if die i is tested in parallel with die j, then die j is also tested in parallel with die i. The last constraint ensures that if die i is tested in parallel with die j, then it must also be tested in parallel with all other dies that are tested in parallel with die j.

Next, we define a second binary variable y_i, which is equal to 0 if die i is tested in parallel with die j on a lower layer ($l_i > l_j$), and 1 otherwise. The total test time T for the stack is the sum of test times of all dies that are tested in series plus the maximum of the test times for each of the sets of parallel tested dies. Using variables x_{ij} and y_i, the total test time T for a stack with set of dies M can be defined as follows.

$$T = \sum_{i=1}^{|M|} y_i \cdot \max_{j=i..|M|} \{x_{ij} \cdot t_j\} \tag{4.4}$$

It should be noted that Equation (4.4) has two non-linear elements, the max function, and the product of variable y_i and the max function. We linearize this by introducing two new variables. The variable c_i takes the value of the max function for each die i and the variable u_i represents the product $y_i \cdot c_i$. The variables u_i and c_i are defined using standard linearization techniques as shown in Figure 2. The linearized function for total test time can be written as follows.

$$T = \sum_{i=1}^{|M|} u_i \tag{4.5}$$

As number of test pins used for parallel testing of dies should not exceed the given test pins W_{max}, a constraint on the total number of pins used to test all dies in a parallel set can be defined as follows.

$$\sum_{j=1}^{|M|} x_{ij} \cdot w_j \leq W_{max} \qquad \forall i \tag{4.6}$$

Similarly, the total number of used TSVs should not exceed the given TSV limit TSV_{max}. The number of TSVs used to connect layer i to layer $i - 1$ is the maximum of the number of pins required by the layer at or above layer i that takes the most test pin connections, and the sum of parallel tested die at or above layer i in the same parallel tested set. Based on this, we can define the constraint on the total number of TSVs used in a test architecture as follows.

$$\sum_{i=2}^{|M|} \{\max_{k=i}^{|M|} \{w_k, \sum_{j=k}^{|M|} w_j \cdot x_{kj}\}\} \leq TSV_{max} \tag{4.7}$$

We can linearize the above set of constraints by representing the max function by a variable d_i. Finally, to complete the ILP model for PSHD, we must define constraints on binary variable y_i and the relationship between binary variable y_i and x_{ij}. For this purpose, we first define a constant C that approaches but is less than 1. We then define y_i as follows:

$$y_1 = 1 \tag{4.8}$$

$$y_i \geq \frac{1}{1-i} \sum_{j=1}^{i-1} (x_{ij} - 1) - C \qquad \forall i > 1 \tag{4.9}$$

The first equation forces y_1 to 1, since the lowest layer cannot be tested in parallel with any layer lower than itself. Constraint 4.9 defines y_i for the other layers. To understand this constraint, we first make the observation that the objective function (as shown in Equation (4.4)) would be minimized if each y_i is zero. This would make the objective function value equal to 0, which is an absolute minimum test time. Thus, we only need to restrict y_i to 1 where it is absolutely necessary, and then we can rely on the objective function to assign a value 0 to all unrestricted y_i variables. This equation considers the range of values that the sum of x_{ij} can take. The fraction in the equation normalizes the sum to a value between 0 and 1 inclusive, while the summation considers all possible cases for a die being tested in parallel with a die below it. The complete ILP model is shown in Figure 2.

Objective:

Minimize $\sum_{i=1}^{|M|} u_i$

Subject to:

$t_{max} = \max_{i=1}^{|M|} t_i$

$c_i \geq x_{ij} \cdot t_j \qquad \forall i, j = i..|M|$

$u_i \geq 0 \qquad \forall i$

$u_i - t_{max} \cdot y_i \leq 0 \qquad \forall i$

$u_i - c_i \leq 0 \qquad \forall i$

$c_i - u_i + t_{max} \cdot y_i \leq t_{max} \qquad \forall i$

$\sum_{j=1}^{|M|} x_{ij} \cdot w_j \leq W_{max} \qquad \forall i$

$x_{ii} = 1 \qquad \forall i$

$x_{ij} = x_{ji} \qquad \forall i, j$

$1 - x_{ij} \geq x_{ik} - x_{jk} \geq x_{ij} - 1 \qquad \forall i \neq j \neq k$

$\sum_{i=2}^{|M|} d_i \leq TSV_{max} \qquad \forall i$

$d_i \geq \sum_{j=k}^{|M|} w_j \cdot x_{kj} \qquad \forall i, k = i..|M|$

$d_i \geq w_j \qquad \forall i, j = i..|M|$

$y_1 = 1$

$y_i \geq \frac{1}{1-i} \sum_{j=1}^{i-1} (x_{ij} - 1) - C \qquad \forall i > 1$

Figure 2: ILP model for 3D TAM optimization PSHD.

4.2 ILP Formulation for PSSD

The ILP formulation for 3D-SICs with soft cores is derived in a similar manner as that for 3D-SICs with hard cores. In this case, the test time t_i for die i is a function of the TAM width w_i assigned to it. Using the variables x_{ij} and y_i as defined in Section 4.1, the total test time T for the stack with the set of soft dies M can be defined as follows.

$$T = \sum_{i=1}^{|M|} y_i \cdot \max_{j=i..|M|} \{x_{ij} \cdot t_j(w_j)\} \tag{4.10}$$

It should be noted that Equation (4.10) has several non-linear elements. To linearize this equation, first we must define the test time function. For this purpose, we introduce a binary variable $g_{in} = 1$ if $w_i = n$, and 0 otherwise. We then linearize this expression using the variable v_{ij} for $x_{ij} \cdot \sum_{n=1}^{k_i} (g_{jn} \cdot t_j(n))$. Similar to Equation (4.5), the variable c_i takes the value of the max function for each die i and the

Objective:

Minimize $\sum_{i=1}^{N} u_i$

Subject to:

$$t_{max} = \max_{i=1}^{|M|} t_i$$
$$c_i \geq v_{ij} \qquad \forall i, j = i..|M|$$
$$v_{ij} \geq 0 \qquad \forall i, j$$
$$v_{ij} - t_{max} \cdot x_{ij} \leq 0 \qquad \forall i, j = 1..|M|$$
$$-\sum_{n=1}^{k_i} (g_{jn} \cdot t_j(n)) + v_{ij} \leq 0 \qquad \forall i, j$$
$$\sum_{n=1}^{k_i} (g_{jn} \cdot t_j(n)) - v_{ij} + t_{max} \cdot x_{ij} \leq t_{max} \qquad \forall i, j$$
$$u_i \geq 0 \qquad \forall i$$
$$u_i - t_{max} \cdot y_i \leq 0 \qquad \forall i$$
$$u_i - c_i \leq 0 \qquad \forall i$$
$$c_i - u_i + t_{max} \cdot y_i \leq t_{max} \qquad \forall i$$
$$z_{ijk} \geq 0 \qquad \forall i, j, k$$
$$z_{ijk} - t_{max} \cdot x_{jk} \leq 0 \qquad \forall i, j, k$$
$$-wi + z_{ijk} \leq 0 \qquad \forall i, j, k$$
$$w_i - z_{ijk} + t_{max} \cdot x_{jk} \leq t_{max} \qquad \forall i, j, k$$
$$\sum_{i=2}^{|M|} d_i \leq TSV_{max}$$
$$d_i \geq \sum_{j=k}^{|M|} z_{jkj} \qquad \forall i, k = i..|M|$$
$$d_i \geq w_j \qquad \forall i, j = i..|M|$$
$$\sum_{j=1}^{|M|} z_{jij} \leq W_{max} \qquad \forall i$$
$$x_{ii} = 1 \qquad \forall i$$
$$x_{ij} = x_{ji} \qquad \forall i, j$$
$$1 - x_{ij} \geq x_{ik} - x_{jk} \geq x_{ij} - 1 \qquad \forall i \neq j \neq k$$
$$y_1 = 1$$
$$y_i \geq \frac{1}{1-i} \sum_{j=1}^{i-1} (x_{ij} - 1) - M \qquad \forall i > 1$$

Figure 3: ILP model for 3D TAM optimization PSSD.

variable u_i represents the product $y_i \cdot c_i$. Since w_j is now a decision variable, we linearize $x_{ij} \cdot w_j$ using a new variable z_{ijk} defined for all i, j, k. We represent the max function by the variable d_i as before. By using the variable z_{ijk}, the TAM width that can be given to each die can be constrained by an upper limit, which is the number of available test pins. We represent this with the following set of inequalities. The complete ILP model for PSSD is shown in Figure 3.

$$\sum_{j=1}^{|M|} z_{jij} \leq W_{max} \qquad \forall i \qquad (4.11)$$

5 Experimental Results

In this section, we present experimental results for the ILP models presented in the previous section. As benchmarks, we have handcrafted three 3D-SICs (as shown in Figure 4) out of several SOCs from the ITC'02 SOC Test Benchmarks as dies inside the 3D-SICs. The SOCs used are d695, f2126, p22810, p34292, and p93791. In SIC 1, the die are ordered such that the lowest die is the lease complex (d695), with die increasing in complexity as one moves higher in the stack. The order is reversed in SIC 2, while for SIC 3, the most complex die is placed in the middle of the stack, with die decreasing in complexity moving out from that die.

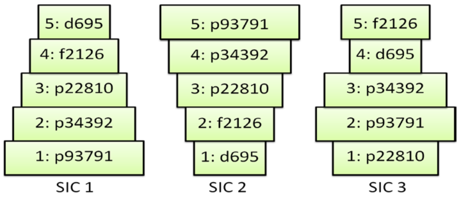


Figure 4: Three 3D-SIC benchmarks.

To determine the test architecture and test time for a given die (SOC) with a given TAM width, we have used the control-aware TAM design method [23] for hybrid TestRail architectures [11]. Control-aware TAM design takes into account the number of scan enable signals required for independent testing of TAMs in the architecture. For PSHD (3D-SIC with hard dies), the test times (cycles) and TAM widths for different dies are listed in Table 1. Note that test pins were assigned to dies based on their sizes in order to avoid very large test times for any individual die.

For a fixed TSV_{max} and range of W_{max}, Table 2 presents results for PSHD for the three benchmark 3D-SICs. The ILP models were ran using XPRESS-MP [24].

Die	**d695**	**f2126**	**p22810**	**p34392**	**p93791**
Test Length	96297	669329	651281	1384949	1947063
Test Pins	15	20	25	25	30

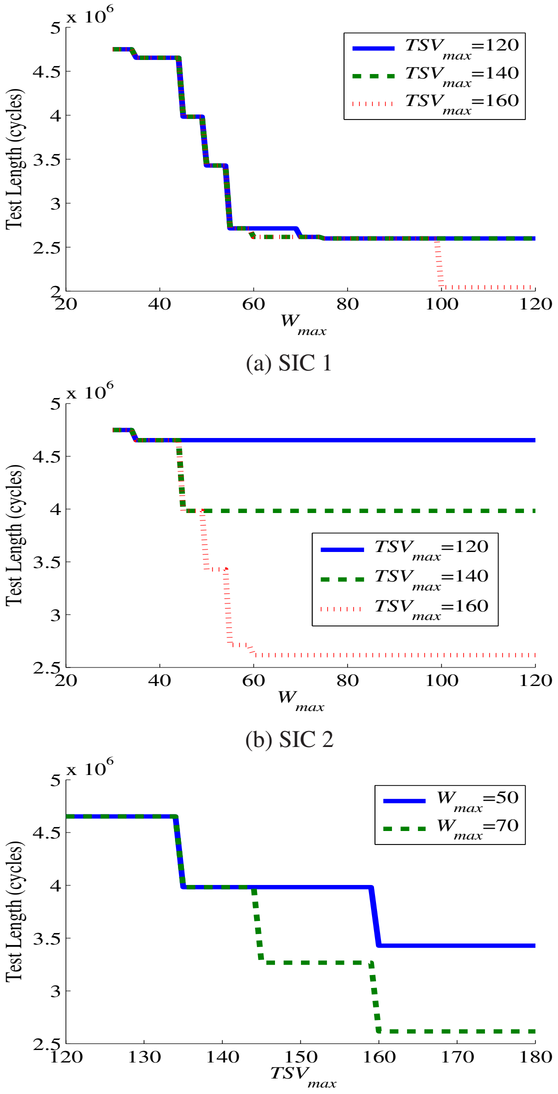
Table 1: Test lengths and number of test pins for dies as required in PSHD.

TSV_{max}	W_{pin}	SIC 1			SIC 2			SIC 3		
		Test Length (cycles)	Test Schedule	Reduction (%)	Test Length (cycles)	Test Schedule	Reduction (%)	Test Length (cycles)	Test Schedule	Reduction (%)
160	30	4748920	1,2,3,4,5	0.00	4748920	1,2,3,4,5	0.00	4748920	1,2,3,4,5	0.00
160	35	4652620	1,2,3,4||5	2.03	4652620	1||2,3,4,5	2.03	4652620	1,2,3,4||5	2.03
160	40	4652620	1,2,3,4||5	2.03	4652620	1||3,2,4,5	2.03	4652620	1,2,3,4||5	2.03
160	45	3983290	1||5,2||4,3	16.12	3983290	1||3,2||4,5	16.12	3983290	1,2||4,3||5	16.12
160	50	3428310	1||4,2||3,5	27.81	3428310	1,2||5,3||4	27.81	3428310	1||3,2||5,4	27.81
160	55	2712690	1||2,3||4,5	42.88	2712690	1,2||3,4||5	42.88	2712690	1||5,2||3,4	42.88
160	60	2616390	1||2,3||4||5	44.91	2616390	1||2||3,4||5	44.91	2616390	1||4||5,2||3	44.91
160	65	2616390	1||2,3||4||5	44.91	2616390	1||2||3,4||5	44.91	2616390	1||4||5,2||3	44.91
160	70	2616390	1||2||5,3||4	44.91	2616390	1||2||3,4||5	44.91	2616390	1||5,2||3||4	44.91
160	75	2598340	1||2||4,3||5	45.29	2616390	1||2||3,4||5	44.91	2598340	1||4,2||3||5	45.29
160	80	2598340	1||2||4,3||5	45.29	2616390	1||2||3,4||5	44.91	2598340	1||4,2||3||5	45.29
160	85	2598340	1||2||4,3||5	45.29	2616390	1||2||3,4||5	44.91	2598340	1||4,2||3||5	45.29
160	90	2598340	1||2||4,3||5	45.29	2616390	1||2||3,4||5	44.91	2598340	1||4,2||3||5	45.29
160	95	2598340	1||2||4,3||5	45.29	2616390	1||2||3,4||5	44.91	2598340	1||4,2||3||5	45.29
160	100	2043360	1||2||3||4,5	56.97	2616390	1||2||3,4||5	44.91	2043360	1||2||3||5,4	56.97
160	105	2043360	1||2||3||4,5	56.97	2616390	1||2||3,4||5	44.91	2043360	1||2||3||5,4	56.97

Table 2: Experimental results for PSHD.

In this table, Column 1 shows the maximum number of TSVs allowed (TSV_{max}), while Column 2 represents the number of available test pins W_{max}. Columns 3, 6 and 9 represent the total test length (cycles) for the stack for 3D-SIC 1, 2 and 3 respectively. Columns 4, 7, and 10 show the resulting test schedule for the 3D-SICs, where the symbol "$\|$" indicates parallel testing of dies, and a "," represents serial testing. Finally, Columns 5, 8, and 11 show the percent decrease in test time over the serial testing case for the three 3D-SICs. From Table 2 we can see that compared to serial testing of all dies (first row in the table), the proposed method obtains up to 57% reduction in test time. Note that although identical test times were obtained for SIC 1 and SIC 3 for $TSV_{max} = 160$, different TAM architectures and test schedules are obtained from the optimization algorithm (see Columns 4 and 10) and test times are better for SIC 1 than SIC 3 for tighter TSV constraints.

For a different number of TSVs (TSV_{max}), Figure 5(a) and Figure 5(b) show the variation in test time T with an increase in number of test pins W_{max} for SIC 1 and SIC 2.



(a) SIC 1

(b) SIC 2

(c) SIC 2

Figure 5: The test time with respect to TSV_{max} and W_{max} for SIC 1 and SIC 2 with hard dies.

From the figures we can see that both TSV_{max} and W_{max} determine which dies should be tested in parallel, and thus the total test time for the stack. For a given value of TSV_{max}, increasing W_{max} does not always decreases the test time. Similarly, increasing TSV_{max} for a given W_{max} does not always decreases the test time. These Pareto-optimal points are shown in Figure 5(c) for SIC 2.

Figure 6 shows the variation in test time for SIC 2 when both TSV_{max} and W_{max} are varied. From the figure, we can see that a small increase in the number of test pins W_{max} for a given TSV_{max} reduces the test time significantly, while to achieve the same reduction in test time with a fixed number of test pins W_{max}, a large increase in TSV_{max} is required.

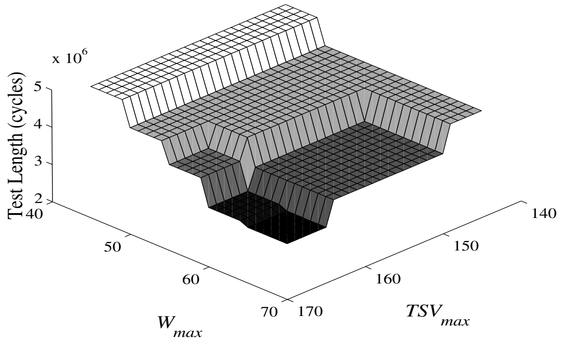


Figure 6: Variation in test time with W_{max} and TSV_{max} for SIC 2 with hard dies.

For PSSD (3D-SIC with soft dies), Pareto-optimality is almost nonexistent when W_{max} is varied; see Figure 7 and Figure 8. This is due to the fact that as dies in the stack are soft, it is always possible to find one die for which adding an extra test pin reduces the overall test time. In Figure 8, some Pareto-optimal points can be identified for SIC 2. This is because the most complex die in a stack tends to be the bottleneck in reducing test time. Since these dies are stacked toward the top of the stack in SIC 2, TSV constraints are more restrictive; the addition of test pins to these dies requires more TSVs and TestElevators throughout the stack. However, for PSSD, although varying W_{max} does not create Pareto-optimal points, varying TSV_{max} results in various Pareto-optimal points as shown in Figure 9.

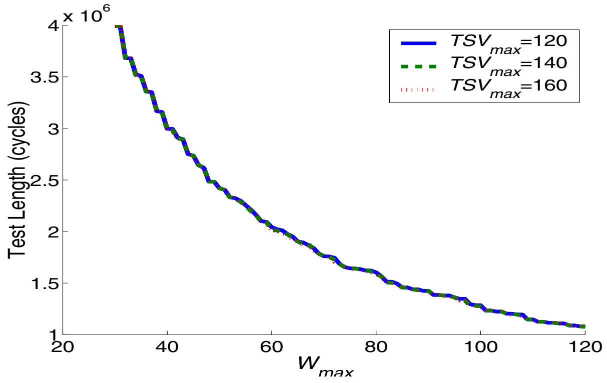


Figure 7: Variation in test time with W_{max} for SIC 1 with soft dies.

Note that this effect is more pronounced in SIC 2 than in the other 3D-SICs. This is because the addition of test pins to the bottleneck die (at the highest layer) introduces a larger TSV overhead than in the other 3D-SICs. Furthermore, as long as W_{max} is sufficient, TSV_{max} is the limiter on test time. For both PSHD and PSSD, the stack configuration (SIC 1) with the largest die at the lowest layer and the smallest die at the highest layer is the best for reducing test time while using the minimum number of TSVs.

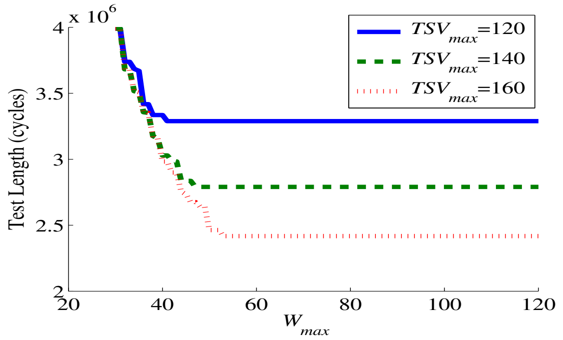


Figure 8: Variation in test time with W_{max} for SIC 2 with soft dies.

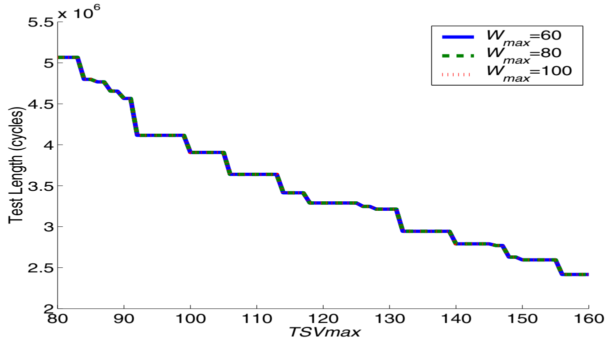


Figure 9: Variation in test time with TSV_{max} for SIC 2 with soft dies.

6 Conclusions

We have introduced the problem of test-architecture optimization for 3D stacked ICs with both hard and soft dies. In case of hard dies, the test architecture for each die is fixed and given, while for soft dies, the test architecture has to be determined while designing the test architecture for the entire stack. We have used ILP techniques to solve the above problems. We have considered constraints on the number of available test pins and the number of TSVs used. Results for three different stack configuration made up of five SOCs taken from the ITC'02 SOC Test Benchmarks show that Pareto-optimal points are present for stacks with hard dies, while for stacks with soft dies, test time is, in most cases, monotonically reduced with an increase in the number of test pins. Moreover, increasing the number of test pins provides a greater reduction in test time compared to an increase in the number of TSVs. Finally, stacks with large and complex dies at the lowest layers lead to lower test times compared to stacks with complex dies at the highest layers.

References

[1] K. Banerjee et al., "3-D ICs: a Novel Chip Design for Improving Deep-submicrometer Interconnect Performance and Systems-on-chip Integration", *Proc. IEEE*, vol. 89, no. 5, pp. 602-633, 2001.

[2] R. Weerasekera et al., "Extending Systems-on-chip to the Third Dimension: Performance, Cost and Technological Tradeoffs", in *Int. Conference on Computer-Aided Design*, pp.212-219, 2007.

[3] G. Loh, Y. Xie, and B. Black, "Processor Design in 3D Die Stacking Technologies", *IEEE Micro* Vol 27, No. 3, 2007, pp.31-48

[4] W.R. Davis et al., "Demystifying 3D ICs: the Pros and Cons of Going Vertical", *IEEE Design and Test of Computers*, vol. 22, no. 6, pp. 498-510, 2005.

[5] Y. Xie, G. H. Loh, and K. Bernstein, "Design Space Exploration for 3D Architectures", J. *Emerg. Technol. Comput. Syst.*, 2(2):65-103, 2006.

[6] H.-H.S. Lee and K. Chakrabarty, "Test challenges for 3D integrated circuits", *IEEE Design & Test of Computers*, vol. 26, pp. 26-35, September/October 2009

[7] E.J. Marinissen and Y. Zorian, "Testing 3D Chips Containing Through-Silicon Vias", *International Test Conference*, E 1.1, 2009.

[8] E.J. Marinissen, V. Iyengar, and K. Chakrabarty. "A Set of Benchmarks for Modular Testing of SOCs", In *International Test Conference*, pp. 519-528, Oct. 2002.

[9] V. Iyengar, K. Chakrabarty, and E.J. Marinissen, "Test Wrapper and Test Access Mechanism Co-optimization for System-on-chip", *Journal of Electronic Testing, Theory, and Applications*, vol. 18, pp. 213-230, 2002.

[10] E.J. Marinissen, S.K. Goel, and M. Lousberg, "Wrapper Design for Embedded Core Test", In *International Test Conference*, pp. 911-920, 2000.

[11] S.K. Goel and E.J. Marinissen. "SOC Test Architecture Design for Efficient Utilization of Test Bandwidth", *ACM Transactions on Design Automation of Electronic Systems*, 8(4):399429, 2003.

[12] E. Larsson, K. Arvidsson, H. Fujiwara, and Z. Peng, "Efficient Test Solutions for Core-based Designs", *Transactions on Computer Aided Design*, vol. 23, no. 5, pp. 758-775, 2004.

[13] Y. Huang et al., "Optimal Core Wrapper width Selection and SOC Test Scheduling based on 3-D Bin Packing Algorithm", In *IEEE International Conference on Computer Design*, pp. 74-82, 2009.

[14] Q. Xu and N. Nicolici, "Resource-Constrained System-on-a-Chip Test: A Survey", *IEE Proceedings: Computers and Digital Techniques*. vol. 152, pp. 67-81, Jan. 2005.

[15] B. Noia, K. Chakrabarty, and Y. Xie, "Test-Wrapper Optimization for Embedded Cores in TSV-Based Three-Dimensional SOCs", *Proc. IEEE International Conference on Computer Design*, pp. 70-77, 2009.

[16] X. Wu, Y. Chen K. Chakrabarty, and Y. Xie, "Test-access Mechanism Optimization for Core-based Three-dimensional SOCs", *IEEE International Conference on Computer Design*, pp.212-218, 2008.

[17] L. Jiang, L. Huang, and Q. Xu, "Test Architecture Design and Optimization for Three-dimensional SoCs", *Design, Automation, and Test in Europe*, pp. 220-225, 2009.

[18] L. Jiang, Q. Xu, K. Chakrabarty and T.M. Mak, "Layout-driven test-architecture design and optimization for 3D SoCs under pre-bond test-pin-count constraint", *Proc. IEEE International Conference on Computer-Aided Design*, pp.191-196, 2009.

[19] E.J. Marinissen, J. Verbree, and M. Konijnenburg, "A Structured and Scalable Test Access Architecture for TSV-Based 3D Stacked ICs", *IEEE VLSI Test Symposium*, Santa Cruz, California, April 2010

[20] M.L. Flottes, J. Pouget, and B. Rouzeyre, "Sessionless Test Scheme: Power-constrained Test Scheduling for System-on-a- Chip", *Proceedings of the 11th IFIP on VLSI-SoC*, pp. 105110, 2001.

[21] E.G. Coffman, Jr., M.R. Garey, D.S. Johnson and R.E. Tarjan. "Performance bounds for level-oriented two-dimensional packing algorithms", *SIAM J. Computing*, vol. 9, pp. 809826, 1980.

[22] X. Dong and Y. Xie. "System-level Cost Analysis and Design Exploration for 3D ICs", *Proceedings of Asia-South Pacific Design Automation Conference (ASP-DAC)*, pp.234-241, Jan. 2009.

[23] S.K. Goel, E.J. Marinissen, "Control-Aware Test Architecture Design for Modular SOC Testing", *European Test Workshop*, pp. 57-62, 2003.

[24] FICO. Xpress-MP.
http://www.fico.com/en/Products/DMTools/Pages/FICO-Xpress-Optimization-Suite.aspx