LUND UNIVERSITY

Scheduling Tests for 3D Stacked Chips under Power Constraints

Sengupta, Breeta; Ingelsson, Urban; Larsson, Erik

Published in:
2011 Sixth IEEE International Symposium on Electronic Design, Test and Application

DOI:
10.1109/DELTA.2011.23

2011

Link to publication

Citation for published version (APA):
Sengupta, B., Ingelsson, U., & Larsson, E. (2011). Scheduling Tests for 3D Stacked Chips under Power Constraints. In *2011 Sixth IEEE International Symposium on Electronic Design, Test and Application* (pp. 72-77). IEEE - Institute of Electrical and Electronics Engineers Inc.. https://doi.org/10.1109/DELTA.2011.23

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

Download date: 19. Dec. 2025

2011 Sixth IEEE International Symposium on Electronic Design, Test and Application

Scheduling Tests for 3D Stacked Chips under Power Constraints

BreetaSenGupta Urban Ingelsson Erik Larsson
Department of Computer and Information Science
Linköping University
SE-581 83 LINKÖPING, SWEDEN
Email: (breeta.sengupta, urban.ingelsson, erik.larsson) @liu.se

***Abstract*- This paper addresses Test Application Time (TAT) reduction for core-based 3D Stacked ICs (SICs). Applying traditional test scheduling methods used for non-stacked chip testing where the same test schedule is applied both at wafer test and at final test to SICs, leads to unnecessarily high TAT. This is because the final test of 3D-SICs includes the testing of all the stacked chips. A key challenge in 3D-SIC testing is to reduce TAT by co-optimizing the wafer test and the final test while meeting power constraints. We consider a system of chips with cores equipped with dedicated Built-In-Self-Test (BIST)-engines and propose a test scheduling approach to reduce TAT while meeting the power constraints. Depending on the test schedule, the control lines that are required for BIST can be shared among several BIST engines. This is taken into account in the test scheduling approach and experiments show significant savings in TAT.**

I. INTRODUCTION

Integrated circuits (ICs) with multiple chips (dies), so called 3D Stacked ICs (SICs), have recently attracted a fair amount of research [1-5]. A 3D-SIC is obtained by stacking and bonding individual chips, which are connected by Through-Silicon Vias (TSVs). Due to imperfections in IC manufacturing, each individual chip must be tested. Recent research has addressed test architecture design for 3D-SICs [6], testing the TSVs [1-6] and 3D-SIC-specific defects [1, 2].

Testing each individual chip is required for both 3D-SICs and traditional non-stacked ICs. Because, IC packaging is costly [7], each chip is tested twice: (1) in wafer sort test, where the bare die is tested (pre-bond test), and (2) in final test where the packaged IC is tested (post-bond test). For non-stacked chips the same test schedule is applied in both pre-bond and post-bond test. However, for a 3D-SIC the process is different. As will be discussed in this paper, applying the same test schedule for both pre-bond and post-bond tests in a 3D-SIC leads to sub-optimal Test Application Time (TAT). TAT is defined as the sum of the testing times for pre-bond tests and post-bond tests. TAT is a major part of the overall test cost. Hence it is important to schedule the tests for 3D-SIC so that TAT is minimized, which is addressed in this paper.

Much work has addressed test scheduling for non-stacked chips with the objective of minimizing TAT [8, 9]. For core-based systems where each core is to be tested, the main method of reducing TAT is to perform core tests concurrently. However, performing tests concurrently leads to higher power consumption than performing them sequentially. The test power consumption must be kept under control [9], to avoid false test positives due to voltage drop and damage due to overheating. For core-based systems, Chou *et al.*[9] proposed a method to schedule tests in sessions while taking resource conflicts and power consumption into account. A session is a group of tests that start at the same time. In the context of systems where each core has a dedicated Built-In Self Test (BIST) engine, all the core tests that are scheduled in the same session can be initiated using a single control line. As a rule, a low number of sessions is beneficial, since it leads to a low number of control lines and implies that several tests are performed concurrently, leading to a low TAT [8, 9, 10]. The studies in [8, 9, 10] address test scheduling for non-stacked chips under power constraints. However, no work has yet addressed test scheduling for 3D-SICs under power constraints, which is the topic of this paper. We propose a power constrained test scheduling approach, which considers a two-chip 3D-SIC design, consisting of cores, each equipped with a dedicated BIST-engine. In this context we present an analysis of the test scheduling problem in Section II leading to an approach in Section III. The experimental results are in Section IV and the conclusions are in Section V.

II. PROBLEM ANALYSIS

Figure 1 shows a chip with three cores where each core is tested by its BIST test. Two parameters are associated with each test: test time and power consumption. The test controller, which is a Finite State Machine (FSM) determines when the test for each core is initiated. Figure 2 shows a test schedule for the tests of the three cores of the Figure 1, which have been scheduled as per [8] where the TAT is minimized and the power consumption at any moment is less than the maximal allowed power consumption P_{max}, which is indicated by a horizontal line. The test schedules are represented with blocks for the core tests, where the height of a block is the power consumption for the test and the width of the block is the test time. The x-axis shows the time taken to perform the tests, and the y-axis marks the power consumption. Two types of constraints are considered for the test schedule. The first constraint type is a resource constraint which determines that two tests are not to be performed concurrently and the second constraint type is a constraint regarding the maximum power consumption, P_{max}, which cannot be exceeded. The test schedule contains three sessions: Session1, Session2 and Session3, as marked in the figure. This chip is a single-die IC, so the same test schedule is applied at pre-bond test (wafer sort) and post-bond test (final test). $TAT = C1 + C1$, as the same test schedule is run twice.

Figure 3 shows a 3D-SIC where Chip1 (from Figure 1) is stacked on top of Chip2. The testing of the 3D-SIC requires pre-bond test of Chip1 and Chip2 and a post-bond test of the stacked chip that tests the whole SIC by including tests for the cores in Chip1 and Chip2. While testing of TSVs is important, the actual test time is fixed and relatively low. In this paper, the testing of TSVs is not addressed. The test durations and

978-0-7695-4306-2/11 $26.00 © 2011 IEEE
DOI 10.1109/DELTA.2011.23

IEEE computer society

power consumption values for each core tests are provided in Table 1. The power constraint value is, P_{max} = 20 units.

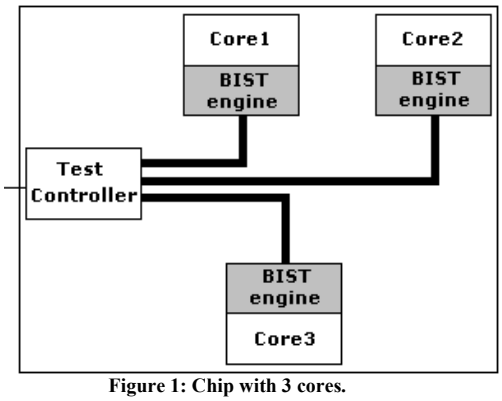

Figure 1: Chip with 3 cores.

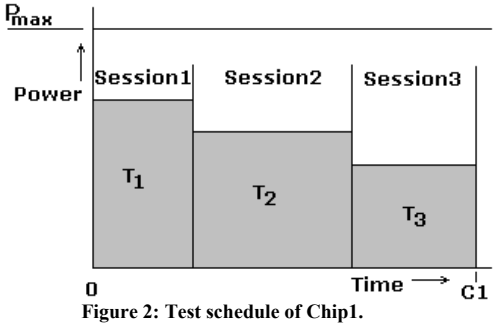

Figure 2: Test schedule of Chip1.

Prior to bonding chips into 3D-SIC, each chip can be considered as individual non-stacked chips and the methods in [8, 9] apply for generating the pre-bond test schedules. Figure 4 shows an example of the pre-bond test schedules for the two chips, Chip1 and Chip2, from Table 1. The test schedule for Chip1 contains three sessions (Session–1, Session–2 and Session–3) and the test schedule for Chip2 contains two sessions (Session–4 and Session–5) as shown in the figure. The test time for the schedules as obtained by [8] are *C1* and *C2* for Chip1 and Chip2, respectively.

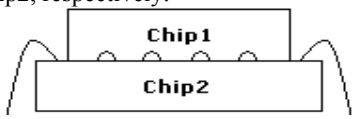

Figure 3: 3D-SIC of Chip1 and Chip2.

Table 1: Test time and power consumption for core tests in Chip1 and Chip2.

Chips	Tests	Duration	Power
Chip1	T1	5	15
	T2	8	12
	T3	6	9
Chip2	T4	2	7
	T5	7	8
	T6	5	9

Once the chips have been stacked, each core of the chips again requires testing. We define three different approaches for test scheduling depending on the available knowledge from the pre-bond test. In this paper, the three approaches are called Serial Processing (SP), Partial Overlapping (PO) and ReScheduling (RS).

In case the only knowledge of the pre-bond test schedules consist of the test time for the schedules and the fact that the pre-bond test schedules are within the power constraint, the limited knowledge available restricts the test schedules that are possible. In this case the post-bond tests are scheduled by *Serial Processing*, which is illustrated in Figure 4. With Serial Processing we mean that the test schedules of individual chips are run serially during post-bond testing. It should be noted that no tests from *different* chips are performed concurrently, because, otherwise we would risk exceeding the power limit. For Serial Processing, the time taken to run the post-bond test schedule is equal to the sum of the time taken to test the individual chips, which has been denoted by TAT_{SP} in the figure. For the schedule in Figure 4, $TAT_{SP} = C1 + C1 + C2 + C2$.

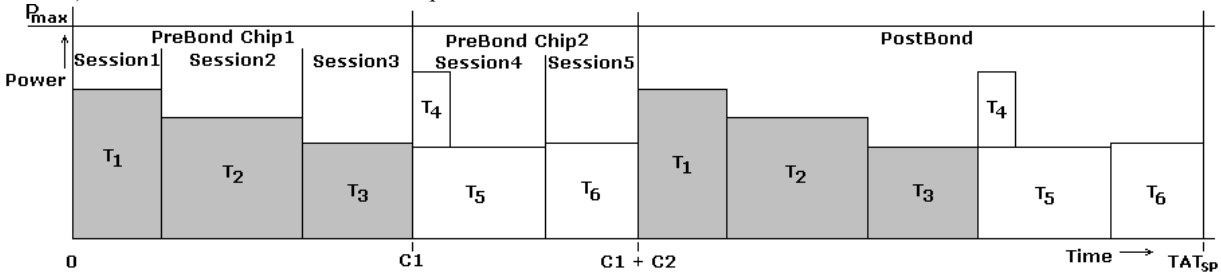

Figure 4: Serial Processing.

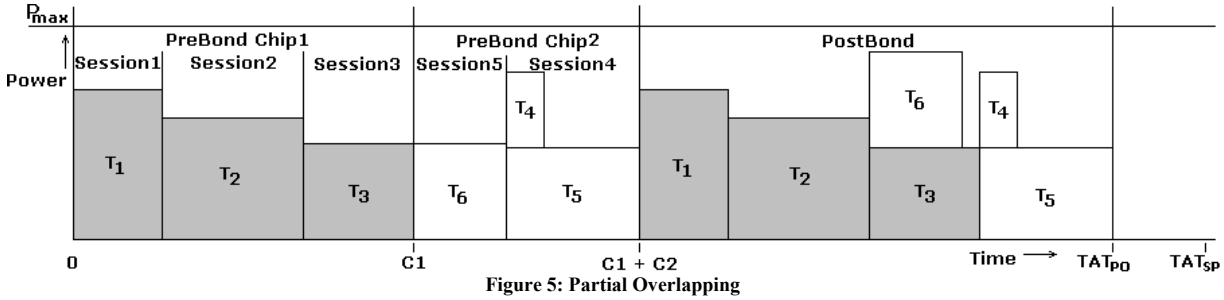

Figure 5: Partial Overlapping

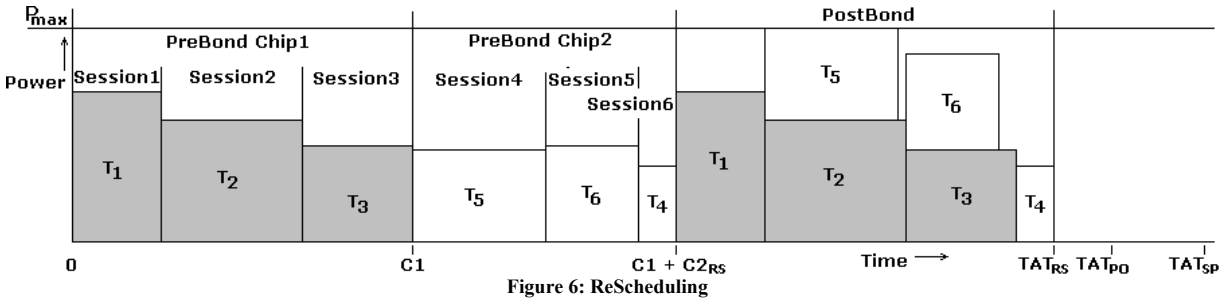

Figure 6: ReScheduling

If the maximum power reached by individual sessions and the test time for the sessions are known, post-bond scheduling by *Partial Overlapping* is possible. In Partial Overlapping, we utilize the knowledge of the test sessions to determine the power compatible test sessions of different chips that can be performed concurrently without exceeding the power constraint. Figure 5 shows the Partial Overlapping test schedule. In the post-bond test schedule, test T_3 of Chip1 (Session3) and test T_6 of Chip2 (Session4) are performed concurrently because they are power compatible. The pre-bond schedule of the chips remain unchanged, but there is a reduction in the TAT equal to the length of test T_6 (Session4) and the resulting TAT_{PO} is lesser than TAT_{SP}.

When full knowledge is available concerning individual tests and sessions of the pre-bond test schedules, *ReScheduling* of the existing schedules can be performed. In the ReScheduling approach, knowledge of the pre-bond test schedules is utilized to create a post-bond test schedule to reduce test time. ReScheduling may cause changes to the pre-bond schedules. In this context, changing the pre-bond schedule means to split a session and replace it with two new sessions. The benefit of splitting a session is that the two new sessions can be scheduled concurrently with sessions of the other chip during post-bond test, if that reduces TAT. Figure 6 depicts the result of the ReScheduling approach. In the original pre-bond test schedule (Figure 4), Session4 consisted of tests T_4 and T_5. In the post-bond test schedule, after rescheduling, test T_4 is performed serially with test T_1, while test T_5 is performed together with test T_2. This results in a reduction of the post-bond test time equal to the duration of test T_5. ReScheduling results in splitting Session4 and renumbering the sessions, as shown in Figure 6, Session4 is test T_5, Session5 is test T_6 and Session6 is test T_4. But because of the splitting of the original Session4, there is an increase in the pre-bond test time for Chip2 from $C2$ to $C2_{RS}$. The increase is equal to the duration of test T_4, which is now performed serially with test T_5. Compared to SP, the reduction in TAT is equal to the sum of the durations of tests T_5 and T_6, minus the duration of test T_4. From the above, it can be seen that ReScheduling leads to lower TAT as compared to Serial Processing and Partial Overlapping, as is shown in Figure 6. However, in contrast to Serial Processing and Partial Overlapping, ReScheduling can lead to an increase in the number of control lines, as a result of splitting sessions. Our approach, detailed below, takes control lines into account.

III. PROPOSED APPROACHES

In this section we first detail the two approaches, Partial Overlapping (PO), and ReScheduling (RS), and then discuss the complexity of the approaches.

PO can be considered as a special case of RS. PO considers only the knowledge of individual sessions, and no sessions are split in the process. Hence, PO can be obtained by disregarding Step1 of RS.

RS is an approach in two steps as is described in the following. Before the first step, the initial pre-bond test for each chip is generated by the heuristic from [8]. Each session of the pre-bond test schedules is given a unique number.

Step1: Figure 7 shows an 11 stage process for implementing Step1 of RS. The key idea is to group the tests of two pre-bond test sessions from different chips in two post-bond test sessions such that the long tests are grouped together and the short tests are grouped together. This way, there will be one long test session and one short test session, instead of the previous two long sessions. The sum of the newly formed session duration is less than for the two original test sessions. In stage 1 of Figure 7, we consider two sessions, Sx and Sy, from the original pre-bond test schedules of two different chips, ChipX and ChipY, respectively. In stage 2, the tests of Sx and Sy are arranged in descending order of length in a list called *M*. A post-bond session, Sa, is produced in stage 3. Starting from the first test in *M*, *i.e.* the test with the longest test time, move the test from *M* to the post-bond session Sa, as shown in stage 4, and stage 5 checks if the total power of Sa is within the power constraint. Stage 4 and stage 5 are iterated as long as the power constraint is met. As soon as P_{max} is exceeded as a result of moving a test from *M* to Sa, that test is moved back to M (stage 6). A new post-bond session, Sb, is created with the remaining tests of list *M*, which is shown in stage 7. Post-bond sessions Sa and Sb contain all the tests from *M*. The pre-bond sessions Sx and Sy are split into test sessions (say Sxa, Sxb, Sya and Syb) according to how the tests were allocated in Sa and Sb. The modified TAT is calculated in stage 8. Stage 9 considers the TAT. If the new test schedule (pre-bond and post-bond) is shorter than the test schedule for SP, the value is included in Table 2 as in stage 10, as the entry for session Sx and session Sy. Otherwise, if there is no reduction the value is set to be zero.

The process described above is repeated for all possible combinations of two sessions from the pre-bond test schedules

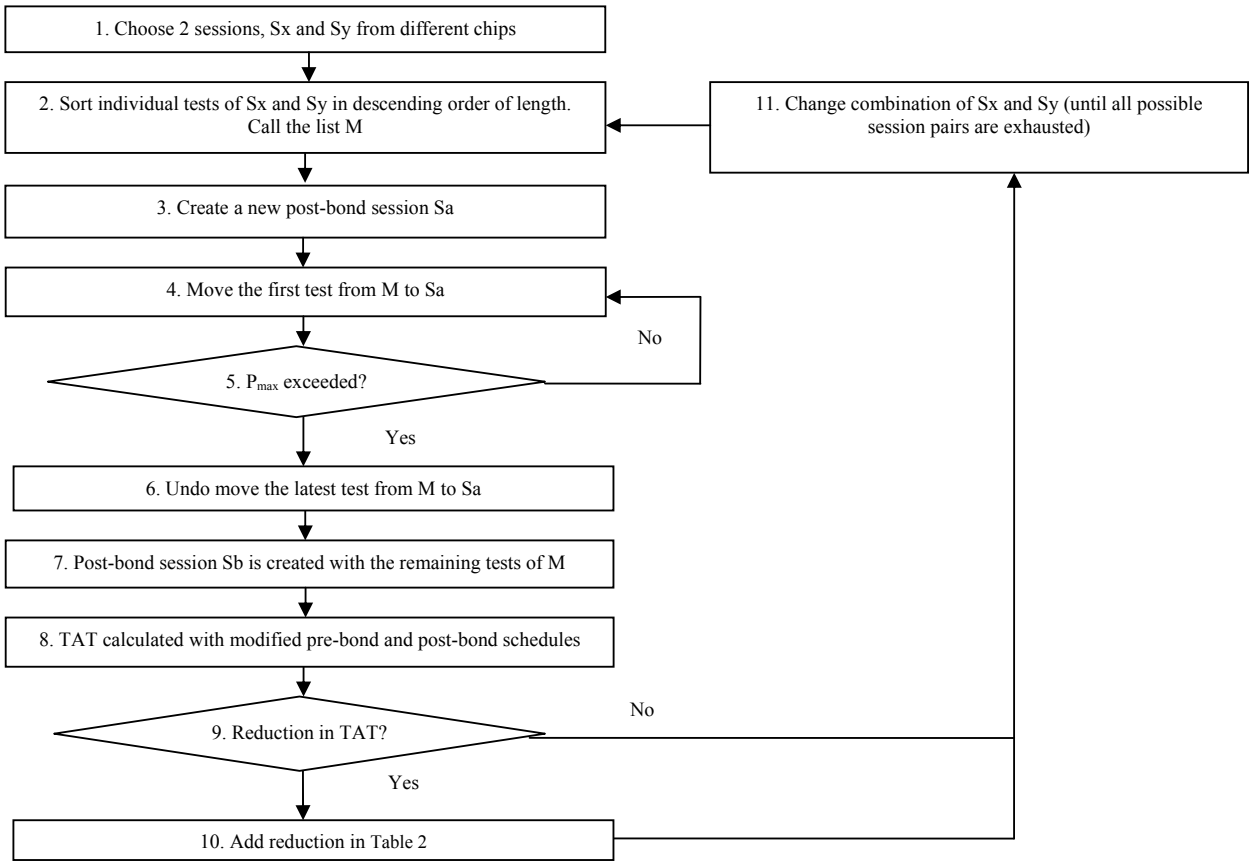

Figure 7 Flow diagram for test scheduling

of the two chips, as is shown in stage 11 of Figure 7.

A key observation from the above is that pairs of sessions can be handled independently. If combining a pair of sessions as described by stage 1 to stage 11 leads to a reduction in TAT compared to the test schedule in SP, a new test schedule can be constructed by combining several such session pairs. The total reduction in TAT can be summed up from the reductions in test time when all session pairs have been considered, while each session has been taken into account only once.

Step2: This step involves the calculation of the maximum reduction in TAT from SP to RS (or PO) by considering all possible session pairs. Table 2 shows the possible reduction in TAT as a result of rescheduling a session of ChipX, as denoted by the column number, with a session of ChipY of the corresponding row number. Given Table 2, a schedule is generated by rescheduling each session of one chip with different sessions of the other chip, such that every session is considered only once. The sessions that are not rescheduled are added to the final schedule without any modification. The objective is to find the combination of rescheduled session pairs, which give the minimum TAT. The values in Table 2 are obtained by rescheduling sessions of the example used during the problem formulation. For example, with respect to Figure 4, considering Session2 from Chip1 and Session4 from Chip2 results in a reduction of 3 time units on rescheduling (as discussed in Step-1), compared to the time required to perform the original Session–2 of Chip1 and Session–4 of Chip2 sequentially, as in SP. In case of PO, where no sessions are split, the values in the table would either be zero (when the sessions are not power compatible), or equal to the length of the smaller session. For example, it was not possible to reduce TAT by combining Session1 with Session4 as marked by 0 in Table 2. The test schedule and the total reduction in TAT are obtained by rescheduling each session of ChipY, (Chip2 in the example) with sessions of ChipX (Chip1 in the example). As discussed before, tests from Session-2 of Chip1 and tests from Session-4 of Chip2 upon rescheduling, result in a reduction of 3 time units, while, Session-5 of Chip2 with Session-3 of Chip1 give a reduction of 5 time units. The sessions that result from the marked session pairs are included in the post-bond test schedule with the summed total of test time reduction adding up to $3 + 5 = 8$ time units. The test time of the rescheduled session pairs are added to the remaining sessions to give TAT. Thus, the final post bond schedule has Session1 in series with the combination of Session2 with Session4 and Session3 with Session5. Thus TAT is 54 time units, obtained by reducing 8 time units from TAT_{SP}, which has 31 time units in both pre-bond and post-bond.

Table 2: Maximum possible time reduction of sessions where ChipX and ChipY refers to the algorithm and Chip1 and Chip2 refers to the example.

		ChipX(Chip1)		
		Session1	Session2	Session3
ChipY(Chip2)	Session4	0	**3**	2
	Session5	0	0	**5**

Finding the combination of session pairs that give the minimum TAT on rescheduling requires comparison of all possible

combinations of session pairs, which is complex. To arrive at the complexity of exploring all possible schedules from Table 2, say ChipX and ChipY have x and y number of sessions respectively, and that $x \geq y$. Then there are x columns and y rows. The first row has x values to choose from. Once a value is chosen, the row and column to which the value belongs are ignored and there remains $x - 1$ values to choose from in the second row. Thus, as we process each row, the number of choices decreases by one. This accounts for a factorial function that describes the number of possible sets of session pairs. But, when $y - 1$ rows have been traversed the last value can be chosen from the remaining $x - y + 1$ columns. Thus, the total number of ways, N, in which values can be selected from Table 2, with each value from a unique row or column, is given by $N = (x - y + 1)\, y!$. Hence, for a total number of ten sessions each in two chips, N becomes as large as 3628800. Thus, it can be seen that the problem of selecting session pairs from Table 2 to explore all possible test schedules is difficult.

Existing heuristics can be applied to obtain a schedule from Table 2. In the following we describe the greedy heuristic that has been used. Prior to applying the heuristic, the rows are first sorted in descending order of the highest value in each row and then the same procedure is applied for the columns. After the table has been sorted, an arbitrary starting value is chosen from the table. The highest value from the neighboring row is then considered along with it. The process is continued until all rows are exhausted. The sum of all the values corresponding to the session pairs considered for rescheduling give the net reduction in test time. Sessions that were not joined with other sessions are added to the list of session pairs to form the schedule. The particular combination of session pairs that lead to the schedule correspond directly to the pre-bond and post-bond test schedules for the stacked 3D design. The combination of session pairs that gives the largest reduction in terms of TAT and an acceptable number of BIST control lines, as determined by the designer of the stacked 3D chip, can be considered as the final schedule.

To arrive to the final schedule, the heuristic is iterated K times, where K is the sum of the number of rows and columns, with different session pairs as starting point to produce a number of solutions that can be evaluated by the designer of the stacked 3D chip with regard to the acceptable number of control lines. This results in Table 3 for the considered example. Schedule1 is the result of combining Session2 with Session4 as well as Session3 with Session5. Schedule 2 is the result of combining Session2 with Session5.

Table 3: TAT reduction versus increase in number of BIST control lines

Schedule	1	2	3	4	5
TAT Reduction	8	2	3	5	0
BIST Control Line Increase	1	1	1	0	0

ReScheduling of sessions resulting in a reduction of TAT can lead to a corresponding increase in the number of BIST control lines due to splitting of sessions. Table 3 shows an example providing the reduction in TAT and the number of additional control lines for five of the test schedules produced by the proposed RS approach.

Complexity of the approach: Here we study the complexity of the RS approach. The approach consists of two steps, Step1 and Step2.

In Step1 of the RS approach, the tests from two sessions are initially sorted by their test durations and stored in the list M. The average time complexity for quick-sort is $O(N \log N)$ for N tests.

Step2 of the problem involves obtaining the maximum sum of individual elements from the matrix, taking one element from each row or column. As discussed, the solution space is large, so the greedy heuristic has been applied, which has a average time complexity of $O(T \log T)$, where T is the number of elements in the matrix. Prior to applying the heuristic, the rows and columns were sorted in descending order of the value of individual elements. The sorting here is also done by quick-sort, which has a complexity of $O(\log T)$. Step 2 is iterated K times, where K is the sum of the number of rows and columns of the corresponding matrix.

Thus the overall complexity of the approach is $O(T \log T)$.

IV. EXPERIMENTAL RESULTS

To demonstrate the benefits of the proposed test scheduling approach, this section describes an experiment to compare TAT achieved by Partial Overlapping (PO) and ReScheduling (RS) with TAT achieved by the straight forward Serial Processing (SP) approach, which is used as baseline. In the experiment, the power constraint is met and the number of BIST control lines required by different test schedules is taken into account. As the RS approach yields a table such as Table 3 with several different test schedule solutions where the acceptable number of control lines determines the final test schedule selection, the experiment is performed with the test schedule that results in the largest TAT reduction (8 time units in the case of Table 3). The initial pre-bond test schedules were generated by the approach in [8] and our approaches where applied for generating the post bond test schedule. The approach proposed in Section III was used to find the maximum reductions in TAT while considering the number of BIST control lines as the number of sessions in the example designs were in a reasonable range.

The experiments are performed with the circuits ASIC Z [10], System L [10] and Muresan [8] (marked by Z, L and M respectively in Table 4) and these circuits were used to create 3D-SICs. These designs are seen as single-die chips and have 9 [10], 14 [11] and 10 [8] cores, respectively. To make a 3D-SIC, two of the three single-die chips are combined. Groups of columns marked Chip1 and Chip2 respectively, show the chips that are combined into 3D-SICs, and groups of columns marked Chip1 & Chip2 and TAT contain experimental results regarding the combined 3D-SICs. To combine the Muresan design with ASIC Z or System L, such as in the fifth row of results in Table 4, adjustments were made on parameters because the values of those parameters in the original designs were given in different orders of magnitude. In the cases marked M* and M**, the parameter values (test time, test power consumption and power constraint) were scaled such that the pair of single-die chips that are combined into 3D-SICs

Table 4: Maximum possible reduction in time with increase in number of control lines. In the table: Z: ASIC Z, L: System L, M:Muresans' Design, SP: Serial Processing Time, PO: Partial Overlapping Time, RS: ReScheduling Time, R(= $\frac{T_{SP}-T_{RS}}{T_{SP}}$): Reduction

	Chip1					Chip2				Chip1 & Chip2				TAT				Incr. in control lines
	Pre-bond Test					Pre-bond Test				Post-Bond Test				Pre-bond + Post-bond				
	T_{SP}	T_{PO}	T_{RS}	R (%)		T_{SP}	T_{PO}	T_{RS}	R (%)	T_{SP}	T_{PO}	T_{RS}	R (%)	T_{SP}	T_{PO}	T_{RS}	R (%)	%(orig)
Z	300	300	300	0	**Z**	300	300	300	0	600	560	560	6.7	1200	1160	1160	3.3	0 (6)
L	1374	1374	1374	0	**L**	1374	1374	1592	-15.9	2748	2107	1592	42.1	5496	4855	4558	17.1	3 (36)
M	26	26	27	-3.8	**M**	26	26	27	-3.8	52	52	48	7.7	104	104	102	1.9	20 (10)
Z	300	300	300	0	**L**	1374	1374	1374	0	1674	1374	1374	17.9	3348	3048	3048	9.0	0 (16)
Z	300	300	300	0	**M***	520	520	520	0	820	780	780	4.9	1640	1600	1600	2.4	0 (8)
L	1374	1374	1374	0	**M****	1040	1040	1040	0	2414	1824	1824	24.4	4828	4238	4238	12.2	0 (18)

have their parameter values in the same order of magnitude. It should be noted that this scaling of parameter values is only performed to enable the experiments. The results are collected in Table 4. The first group of four columns marked Pre-bond test for Chip1 show the test times for SP, PO and RS for the pre-bond schedules for Chip1. The fourth column in the group shows the relative reduction in pre-bond test time of RS compared to SP. It should be noted that a negative reduction is an increase. Similarly, the second group of four columns shows the Pre-bond test for Chip2. The third group of four columns marked Chip1 & Chip2, Post-bond test, show test time for the post-bond test schedule generated by the three approaches, and gives the relative amount of post-bond test time reduction achieved comparing the result for SP with the result for RS. The group of columns marked TAT includes the sum of the pre-bond test times and post-bond test times. The first three columns in the group of four show TAT for the SP, PO and RS approaches, respectively. The relative reduction in TAT is shown in the last of the four columns where RS is compared against SP. The right-most column of Table 4 shows the relative increase in the number of control lines that result from splitting sessions in the RS approach. The number of control lines for the SP approach is shown in parenthesis.

From Table 4, it can be seen that RS can achieve up to 42.1% reduction in the post-bond test schedule time in comparison to SP, when two chips of System L are stacked to form the 3D-SIC. This is for the 3D-SIC consisting of two System L chips. This result can be explained by a high power constraint, which enables a beneficial post-bond test schedule where many core tests are performed concurrently. In particular for the design with two System L chips, one session was split, resulting in an additional control line and an increase in the pre-bond test schedule duration. The reduction in TAT was 17.1%. It should be noted that other 3D-SICs consisting of two identical chips (such as the pair of ASIC Z chips) does not lead to the same result. For the 3D-SIC design made up by a pair of ASIC Z chips, TAT was reduced by 3.3% and RS and PO achieved the same result. This corresponds to a case when it is not possible to reduce TAT by splitting sessions. In the six experiments, only two experiments led to splitting of sessions, which increased the number of BIST control lines. For the other four experiments, the reduction in TAT was achieved without splitting sessions and the best result achieved without splitting sessions was 12.2% reduction in TAT.

V. CONCLUSION

In this paper, the problem of power-constrained test scheduling for 3D Stacked Integrated Circuits (SICs) has been addressed for the first time. It is shown that the test planning for 3D-SICs is different, compared to the test planning for non-stacked ICs, and requires specific test scheduling solutions. The paper proposes two test scheduling approaches, Partial Overlapping and ReScheduling, that minimize test application time while taking power-constraints and the number of BIST control lines required to implement a test schedule into account. The two scheduling approaches and a straight forward approach (Serial Processing) have been implemented and experiments with several benchmarks show up to 17.1% reduction in test application time and an average reduction of 7.7% in test application time with a 3.8% average increase in the number of BIST control lines over the Serial Processing scheme.

REFERENCES

[1] E. J. Marinissen, Y. Zorian. Testing 3D Chips Containing Through-Silicon Vias. *IEEE ITC, paper ET1.1,* pp. 1-11, 2009.

[2] H.-H. S. Lee and K. Chakrabarty. Test Challenges for 3D Integrated Circuits. *IEEE Design and Test of Computers, Special Issue on 3D IC Design and Test*, pp. 26-35, Oct 2009.

[3] D. L. Lewis and H.-H. S. Lee. A Scan-Island Based Design Enabling Pre-bond Testability in Die-Stacked Microprocessors. *IEEE ITC*, paper 21.2, pp. 1-8, 2007.

[4] X. Wu, P. Falkenstern, and Y. Xie. Scan Chain Design for Three-Dimensional Integrated Circuits (3D ICs). *ICCD*, pp. 208–214, 2007.

[5] Y.-J. Lee and S. K. Lim. Co-Optimization of Signal, Power, and Thermal Distribution Networks for 3D ICs. *Electrical Design of Advanced Packaging and Systems Symposium*, pp. 163-166, 2008.

[6] B. Noia, S. K. Goel, K. Chakrabarty, E. J. Marinissen and J. Verbree. Test-Architecture Optimization for TSV-Based 3D Stacked ICs. *IEEE ETS*, pp. 24-29, May 2010.

[7] J. Verbree, E. J. Marinissen, P. Roussel and D. Velenis. On the Cost-Effectiveness of Matching Repositories of Pre-Tested Wafers for Wafer-to-Wafer 3D Chip Stacking. *IEEE ETS*, pp. 36-41, May 2010.

[8] V. Muresan, X. Wang, V. Muresan and M. Vladutiu. Greedy Tree Growing Heuristics on Block-Test Scheduling Under Power Constraints, *JETTA*, pp. 61-78, 2004.

[9] R. M. Chou, K. K. Saluja and V. D. Agrawal. Scheduling tests for VLSI systems under power constraints. *IEEE Trans. VLSI Systems,* vol. 5, no.2, pp. 175-185, June 1997.

[10] Y. Zorian. A Distributed BIST Control Scheme for Complex VLSI devices. *IEEE VTS,* pp. 6–11, April 1993.

[11] E. Larsson and Z. Peng. An Integrated Framework for the Design and Optimization of SOC Test Solutions, *JETTA*, *Special Issue on Plug-and-Play Test Automation for System-on-a-Chip*, vol. 18, no. 4, pp. 385-400, August 2002.