Scheduling Tests for Stacked 3D Chips under Power Constraints

Sengupta, Breeta; Ingelsson, Urban; Larsson, Erik

2010

Scheduling Tests for Stacked 3D Chips under Power Constraints

Breeta SenGupta Urban Ingelsson Erik Larsson

Department of Computer and Information Science
Linköping University
SE-581 83 LINKÖPING, SWEDEN
Email: {g-brese, urbin, erila}@ida.liu.se

***Abstract*- This paper addresses test application time (TAT) reduction for core-based stacked 3D chips. In contrast to the traditional method of testing non-stacked chips where the same test schedule is applied both at wafer test and at final test, stacked 3D chips need a pre-bond test schedule for each individual chip and a different post-bond test schedule where all chips are jointly tested. We consider a system of core-based chips where each core is tested with a dedicated Built-In Self-Test (BIST) engine and define an algorithm that defines each pre-bond test schedule and the post-bond test schedule such that the overall TAT is minimized and power constraints are met. The cost due to the number of BIST control-lines is also taken into account. Experiments with the proposed algorithm show significant savings in TAT.**

I. INTRODUCTION

Integrated circuits (ICs) with multiple chips (dies), so called stacked 3D chips, have recently attracted a fair amount of research [3-6]. A 3D chip is obtained by stacking and bonding individual chips. There are several techniques for the bonding process [3, 4]. Due to imperfections in IC manufacturing, each individual IC must be tested. This is true both for stacked 3D chips and traditional non-stacked chips. Because IC packaging is costly, each chip is tested twice; first at wafer sort where the bare die is tested and then at final test where the packaged IC is tested. For non-stacked chips, the same test schedule is applied first at wafer sort and then at final test. However, for stacked 3D chips the process is very different. First each chip must be tested individually (pre-bond test) and then the complete stacked 3D chip is tested (post-bond test). As will be discussed in this paper, a single test schedule cannot be used for both pre-bond and post-bond test. As test application time (TAT) is a major part of the overall test cost, it is important to schedule the tests for stacked 3D chips, such that the total 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 [1, 2]. 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 [2]. For core-based systems where each core has a dedicated Built-In Self-Test (BIST) engine, Chou et al. [2] proposed a method to schedule the tests in sessions while taking test conflicts and power consumption into account. Muresan et al. [1] proposed a heuristic to schedule the tests in sessions such that TAT is minimized while meeting test power constraints. A session is a group of tests that start at the same time. A single control line can be employed to initiate the session. As a rule, a low number of sessions is good, since it leads to a low number of control lines and implies that several tests are performed concurrently, leading to a low TAT. The studies in [1, 2] address test scheduling for non-stacked chips under power constraints. However, very little work has addressed the test scheduling for 3D stacked chips under test power constraints, which is the topic of this paper. We propose a test scheduling method which considers a two-chip stacked 3D design, consisting of cores, each equipped with a dedicated BIST engine. There is a BIST controller that is connected to each core by a control line and implements the test schedule by sending signals to initiate the core tests. In this context we present an analysis of the test scheduling problem in Section II leading to a procedure in Section III. The experimental results are in Section IV and the conclusions are in Section V.

II. PROBLEM ANALYSIS

Prior to bonding chips into stacked 3D design, each chip can be considered as individual non-stacked chips and the methods in [1, 2] apply for generating the pre-bond test schedules. Fig.1 shows an example of the pre-bond test schedules for two chips, Chip1 and Chip2. The test schedule for Chip1 contains three sessions and the test schedule for Chip2 contains two sessions. The pre-bond tests have been scheduled as per [2]. 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. Two types of constraints control the test schedule: resource constraints can determine that two tests are not to be performed concurrently and a constraint regarding the maximum power consumption, P_{max}, cannot be exceeded. In Fig.1, P_{max} is indicated by a horizontal line. The test time for the schedules as obtained by [2] are C1 and C2 for Chip1 and Chip2 respectively.

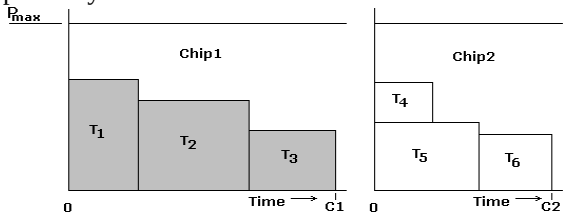

Fig.1. Pre-Bond Test Schedule of Chips.

Once the chips have been stacked, each chip again requires testing, called the post-bond test. We define three different types of test scheduling depending on the available knowledge. In this paper, the three types are called Serial Processing, Partial Overlap and ReScheduling.

In case no knowledge of the pre-bond test schedules is available, tests are scheduled by *Serial Processing*, which is illustrated in Fig.2, for the example from Fig.1 (assuming that the two chips are stacked). 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 run concurrently, 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.

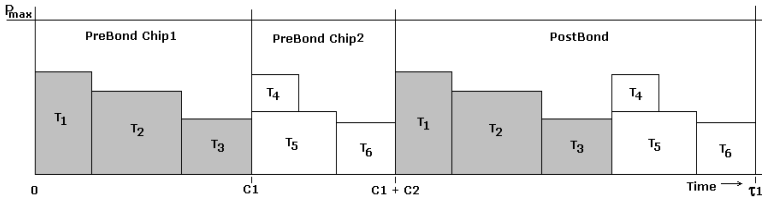

Fig.2. Serial Processing.

If the knowledge of the maximum power reached by individual sessions and the session lengths are provided, post-bond scheduling by *Partial Overlap* is possible. In Partial Overlap, we utilize the knowledge of the test sessions, to determine that power compatible test sessions of different chips can be run concurrently. Partial Overlap does not require altering the pre-bond schedules.

Fig.3. shows the Partial Overlap schedule. In the post-bond schedule, test T_3 of Chip1 and test T_6 of Chip2 are run concurrently. The pre-bond schedule of the chips remain unchanged, but there is a reduction in the total TAT equal to the length of test T_6 and the resulting TAT is τ_2.

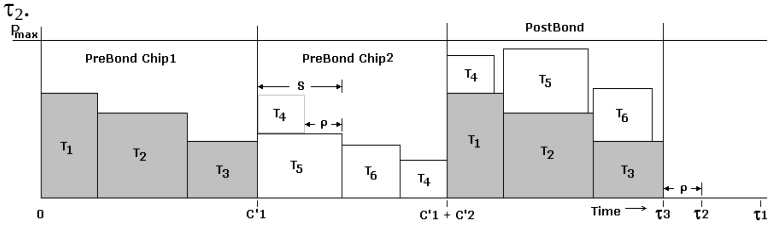

Fig.3. Partial Overlap.

When the full knowledge of individual tests and sessions of the pre-bond test schedules are available, total *ReScheduling* of the existing schedules can be done. In the ReScheduling approach, knowledge of the pre-bond test schedules is utilized to create a post-bond test schedule, and minimum possible changes are made to the pre-bond schedules to reduce the total TAT. A change in the pre-bond schedule in this context is to split a session and replace it with two new sessions which in turn can be scheduled concurrently with sessions of the other chip, if that reduces the total TAT.

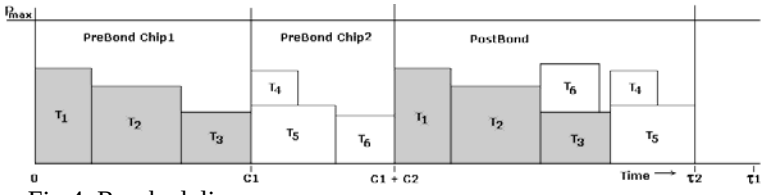

Fig.4. Rescheduling.

Fig.4 depicts the result of the ReScheduling approach. In the post-bond schedule, the session comprising of tests T_4 and T_5 in the previous examples, is split, and test T_4 is run concurrently with test T_1, while test T_5 is run together with test T_2. This results in a reduction in the post-bond TAT equal to the length of test T_5, marked in Fig.4 as S. But because of the splitting of the session, there is an increase in the TAT in the pre-bond schedule from C2 to C'2. The increase is equal to the length of test T_4, which is now run serially with test T_5. Thus the overall reduction in the total TAT is the difference of the lengths of tests T_5 and T_4, equal to ρ and the reduced TAT is τ_3. From the above, it can be seen that ReScheduling leads to lower TAT as compared to Serial Processing and Partial Overlap, as is also shown in Fig.4. However, in contrast to Serial Processing and Partial Overlap, ReScheduling can lead to an increase in the number of control lines, as a result of splitting sessions.

III. PROPOSED APPROACH

In this section a procedure for ReScheduling is presented. A detailed step by step procedure for the scheduling of test in stacked ICs is hence provided.

The pre-bond test schedules are given, which are obtained by applying the heuristic discussed in [1], generating sessions. Each session of the individual chips are numbered serially.

Step1: In this step we discuss the method of rescheduling, which is an iterative method of rearranging the tests of two sessions from the pre-bond test schedules to produce a session for the post-bond test schedule, with the aim at reducing the total test time. We consider two sessions, Sx and Sy, from the pre-bond test schedules of two different chips, ChipX and ChipY, to form new sessions for the post-bond test schedule. Only two sessions are considered in each iteration, and they must be from different chips, because tests that belong to different sessions for the same chip have power and resource constraints that prevent rearranging of tests among them. This is because of how the pre-bond test schedules were originally generated, as described above. All tests of Sx and Sy are arranged in descending order of length in a single list called M. Tests with the same length are arranged in descending order according to their power consumption. A post-bond session, Sa, is produced as follows. Starting from the first, i.e. the longest, test in the list M, the tests are included serially in the post-bond session Sa, in decreasing order of lengths, until the power constraint is met. In this process, each test that is included in session Sa (in the post-bond test schedule) is also removed from its original session (either Sx or Sy) and added to a pre-bond session, called Sx' or Sy' depending on its original session. This move of a test from one pre-bond session to another pre-bond session represents the splitting of a session. If the first test that (if included in session Sa) would cause the power constraint to be broken in session Sa, belongs to session Sx (session Sy), no more tests from session Sx (session Sy) are considered for inclusion in session Sa. In this case, the remaining tests of session Sy (session Sx) are included serially in Sa, in decreasing order of length, until the power constraint is met. If all the tests of Sy (Sx) are contained in the post bond session Sa, then as many as possible of the remaining tests of Sx (Sy) are again included in the post-bond session Sa, until the power constraint is met. This ensures that the minimum number of tests are left out after the tests of sessions Sx and Sy are rearranged to form session Sa. The tests in session Sa, from ChipX and ChipY, constitute the rescheduled pre-bond sessions Sx and Sy of ChipX and ChipY, which are no longer considered for rescheduling during subsequent iterations.

The remaining tests of the original pre-bond sessions Sx and Sy, which are not included in session Sa, form two new pre-bond sessions Sx' of ChipX and Sy' of ChipY. Hence, seven sessions, Sx, Sy, Sa, Sx'(in pre-bond and post-bond) and Sy'(in pre-bond and post-bond) are obtained as a result of rescheduling two pre-bond sessions in the post-bond test schedule. It should be noted that some of these seven sessions may be empty.

The above mentioned process can be iterated with session Sx' of ChipX and any session of ChipY and vice-versa (with session Sy' of ChipY and any session of ChipX). A net reduction in TAT is obtained if the sum of the lengths of the sessions rescheduled is greater than the increase in TAT resulting from the splitting of the sessions.

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

Step2: Table 1 shows the reduction in TAT as a result of rescheduling a session of ChipX, as denoted by the row number, with a session of ChipY of the corresponding column number. The new test schedules and the total reduction in TAT are obtained by rescheduling all sessions of ChipY (as it has a lower number of sessions) with a session of ChipX, with no two sessions of ChipY being rescheduled with the same session in ChipX. It should be noted that the reason why no two sessions of ChipY can be rescheduled with the same session of ChipX is, as mentioned above, is due to time and resource constraints.

Session number↓→		ChipX				
		1	**2**	**3**	**4**	**5**
ChipY	**1**	3	0	2	0	**3**
	2	**6**	0	0	5	0
	3	5	0	0	**6**	0

Table 1. Maximum possible time reduction of sessions.

An example of a rescheduling is shown in Table 1, marked by the highlighted values. In this example, tests from Session 1 of ChipX and tests from Session 2 of ChipY are used to form sessions in the post-bond test schedule (as discussed in Step 1) and the resulting reduction in the post-bond test time is 6 time units, compared to the time required to perform the original Session 1 of ChipX and Session 2 of ChipY sequentially. Correspondingly, Session 3 of ChipY is considered together with Session 4 of ChipX and Session 1 of ChipY is considered with Session 5 of ChipY for rescheduling. 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 6+6+3=15 time units. The remaining sessions of ChipY, Session 2 and Session 3, are also included in the post-bond test schedule, without any alteration, but for these sessions, there is no reduction in test time.

The total number, N, of ways in which values can be selected from Table 1, with each value from a unique row or column, is $N = (x - y + 1) * x!$ and $x \leq y$, for x and y number of sessions for ChipX and ChipY. Hence, for a total number of ten sessions each in two chips, N becomes as large as 3628800. From this reasoning, it can be seen that the problem of selecting session pairs from Table 1 to form the new test schedules is difficult. This problem can be mapped onto the well known Travelling Salesman Problem (TSP). To map the problem at hand to the TSP, each session can be considered as a city, and the time reduced by selecting a pair of sessions as in Table 1 can be seen as the cost of moving between the cities. Thus, the picture can be projected as to having sessions belonging to the respective chips can be projected as two sets of cities, and the Travelling Salesman can move between any two cities which belong to the two different sets, which are provided with the weighed cost. The objective is to find the maximum (instead of minimum, as in general) cost incurred while covering all the cities. Existing heuristics can be applied to obtain a solution to the problem at hand.

TAT redn	15	**14**	13	9	8	3
CL inc	9	**4**	3	5	7	2

Table 2. TAT reduction versus increase in BIST control lines.

Each 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. In this context, it can be noted that the solutions achieved by applying a heuristic for the TSP are not optimal, because it is possible that the best solution in terms of TAT would require an unacceptable increase in the number of control lines, and hence be rejected. The solution with the maximum 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 solution. Therefore, the proposed procedure is used a number of times 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. Table 2 shows an example providing the reduction in TAT and the number of additional control lines for a number of test schedules produced by the proposed procedure.

The particular combination of session pairs that lead to the solution correspond directly to the pre-bond and post-bond test schedules for the stacked 3D design.

IV. EXPERIMENTAL RESULTS

The test scheduling procedure in Section III was applied to stacked 3D designs that were constructed as shown in Column 1 and Column 2 of Table 3, by pairing the known benchmark designs ASIC Z [7], System L [8] and Muresan [1] (marked by Z, L and M respectively), effectively stacking single-die chips corresponding to the pair of designs into 3D chips. To combine the Muresan design with ASIC Z or System L to construct 3D designs it was required to adjust some parameters, because the parameter values in the original designs were given in different orders of magnitude. In the cases marked M* and M**, we have scaled the parameter values (the core test lengths, the core test power values and the power constraint of the design) so that the pair of designs that are used to construct a 3D design have their parameter values in the same order of magnitude. The results of the test scheduling procedure while choosing the largest TAT reduction achieved(Table 2) are shown in Table 3.

The four columns marked Chip1 Pre-bond show how the proposed procedure affects the pre-bond test schedule for Chip1. The first three of the four columns show the TAT for the Serial Processing, Partial Overlap and ReScheduling. The fourth column in this group shows the

Chip1					**Chip2**					**3D design of Chip1 & Chip2**				**Total Test**				**Incr. in control lines**
	Pre-bond					**Pre-bond**				**Post-Bond**				**Pre-bond Chip1, Pre-bond Chip2 and Post-bond**				
	Serial	Partial Overlap	Re Sched.	Incr. I (%)		Serial	Partial Overlap	Re Sched.	Incr. I (%)	Serial	Partial Overlap	Re Sched.	Redu. R (%)	Serial	Partial Overlap	Re Sched.	Redu. 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)
Z	300	300	300	0	**L**	1374	1374	1374	0	1674	1374	1374	17.9%	3348	3048	3048	9.0%	0% (16)
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	**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)

Table 2. Maximum possible reduction in time with increase in number of control lines.

increase in TAT that results from splitting sessions in ReScheduling. The same applies to the next group of columns, marked Chip2 Pre-bond, but for Chip2. Similarly, the four columns marked 3D design of Chip1 & Chip2, post-bond, shows the TAT for the post-bond test schedule generated by the proposed procedure, and gives the relative amount of TAT reduction achieved, comparing the result for Serial Processing with the result for ReScheduling. The same way, the overall results, considering the total test, including both pre-bond tests and the post-bond test, are presented in the columns marked Total Test. The first three columns in this group of four, shows the sum of the TATs for the Serial Processing, Partial Overlap and ReScheduling approaches respectively. The overall relative reduction in TAT is shown in the last of the four columns, comparing Serial Processing to ReScheduling. The right-most column of Table 3 shows the relative increase in the number of control lines that result from splitting sessions in the ReScheduling approach. The number of control lines for the Serial Processing approach is shown in parenthesis.

From Table 3, it can be seen that the proposed procedure can achieve up to 42.1% reduction in the post-bond TAT (for the 3D design consisting of two SystemL chips). This result can be explained by a high power constraint, which enables a beneficial post-bond test schedule where parts of the pre-bond test schedules for the two chips are performed concurrently. In this case, a sessions was split, resulting in an additional control line and an increase in the pre-bond TAT. The net reduction in total TAT was 17.1%. It should be noted that other 3D designs consisting of two identical chips (such as the pair of ASIC Z chips) does not lead to the same result. For the 3D design made up by a pair of ASIC Z chips, the total TAT was reduced by 3.3% and ReScheduling and Partial Overlap achieved the same result. This corresponds to a case when it is not possible to reduce the total TAT by splitting sessions. In the six experiments for which Table 2 shows the results, only two experiments led to splitting of sessions. 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 test-scheduling with a power constraint for a stacked 3D design has been discussed. The chips are core-based and each core is tested by one BIST test. Three approaches are discussed, Serial Processing, Partial Overlap and ReScheduling. These approaches depend on different levels of available information regarding the 3D design. The ReScheduling approach can be applied when full knowledge of the 3D design is available. The ReScheduling approach relies on previously existing methods to generate schedules for testing prior to the bonding of the chips that make up the 3D design. To reduce the total TAT, the approach generates a schedule for testing after the bonding (post-bond) and reduces the TAT for this post-bond test schedule at the cost of increasing the TAT in the pre-bond test schedules and at the cost of additional control lines. The ReScheduling approach is discussed in detail and it is shown how it can be combined with a solver for the Traveling Salesman Problem. The test scheduling problem solved by the ReScheduling approach has not been considered in prior work, since no previous power-constrained test scheduling approach has considered the challenge of scheduling tests for stacked 3D chips. Experimental results demonstrate an average reduction of 7.7% in TAT with a 3.8% increase in the number of BIST control lines. Reduction in TAT is up to 17.1% compared to the test schedule that is a sequential application of the pre-bond test schedules.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] 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.

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

[6] 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*, 2008.

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

[8] Erik Larsson and Zebo 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), August 2002, pages 385-400.