Autonomous Testing for 3D-ICs with IEEE Std. 1687

Ye, Jin-Cun; Kochte, Michael A.; Lee, Kuen-Jong; Wunderlich, Hans-Joachim

Abstract: IEEE Std. 1687, or IJTAG, defines flexible serial scan-based architectures for accessing embedded instruments efficiently. In this paper, we present a novel test architecture that employs IEEE Std. 1687 together with an efficient test controller to carry out 3D-IC testing autonomously. The test controller can deliver parallel test data for the IEEE Std. 1687 structures and the cores under test, and provide required control signals to control the whole test procedure. This design can achieve at-speed, autonomous and programmable testing in 3D-ICs. Experimental results show that the additional area and test cycle overhead of this architecture is small considering its autonomous test capability.

Preprint

Autonomous Testing for 3D-ICs with IEEE Std. 1687

Jin-Cun Ye[1], Michael A. Kochte[1,2], Kuen-Jong Lee[1], and Hans-Joachim Wunderlich[2]

[1] Dept. EE, National Cheng Kung University, Tainan, Taiwan
[2] ITI, University of Stuttgart, Germany

Abstract— **IEEE Std. 1687, or IJTAG, defines flexible serial scan-based architectures for accessing embedded instruments efficiently. In this paper, we present a novel test architecture that employs IEEE Std. 1687 together with an efficient test controller to carry out 3D-IC testing autonomously. The test controller can deliver parallel test data for the IEEE Std. 1687 structures and the cores under test, and provide required control signals to control the whole test procedure. This design can achieve at-speed, autonomous and programmable testing in 3D-ICs. Experimental results show that the additional area and test cycle overhead of this architecture is small considering its autonomous test capability.**

Index Terms—**IEEE Std. 1687, IJTAG, reconfigurable scan network, autonomous testing, 3D-ICs, DFT.**

I. INTRODUCTION

3D-ICs are stacked dies with vertical interconnection using through-silicon vias (TSVs). They provide increased communication bandwidth and allow the integration of heterogeneous dies. The effective and efficient test of such 3D-ICs requires novel design-for-test (DFT) architectures for pre- and post-bond testing of both stacked dies and TSVs [1], [12]. The use of the common test standards IEEE Std. 1149.1, IEEE Std. 1500, the ongoing proposal IEEE P1838 [13], as well as novel standards for highly reconfigurable scan based access such as IEEE Std. 1687 [2] allows to construct flexible and highly scalable DFT architectures for 3D-ICs.

In [5], the authors proposed the use of die-level wrappers based on IEEE 1149.1 (JTAG) and IEEE 1500 for 3D-IC test supporting pre-bond, mid-bond, and post-bond testing. This architecture also supports parallel test access to the IEEE 1500 wrappers. The test access modes in [5] between the dies of a 3D-IC such as die access, die bypass, and turnaround mode are similar to those outlined in IEEE P1838 [13].

Reconfigurable scan networks have been recently standardized in IEEE 1687 [2] for serial scan-based access of on-chip instrumentation (on-chip test data registers) with low access time via a JTAG test access port (TAP). This standard, also called the IJTAG standard, allows hierarchical networks as well as irregular networks with complex sequential and combinational dependencies between instrument accesses [6]. The reconfigurability allows to change the scan path such that the targeted instruments can be accessed with a minimum number of shift cycles. The standard also defines the Instrument Connectivity Language (ICL) to describe the scan network structure, and the Procedural Description Language (PDL) to describe instrument accesses and access sequences. The case studies in [8] and [9] demonstrate the application of IEEE 1687 to setup test sessions in 2D designs and the achievable degree of automation when using ICL and PDL with commercial tools.

In [3], IEEE 1687 is adopted in a 3D-IC DFT architecture to distribute test data to cores and dies. The architecture supports pre-bond and post-bond test modes. It uses the die-detection mechanism of [4] to automatically determine whether another die is stacked and connected on top and accordingly establish the required communication paths. If a pre-bond test is performed, test data is provided from external test equipment via probe pads.

Test time and external ATE costs can be reduced by built-in test controllers that can drive the test application of complex VLSI systems autonomously [7], [10], [11]. Autonomous in-field testing is required for safety-critical systems to detect latent or emerging defects before they cause a failure during operation. In [7] and [11], an autonomous on-chip test platform based on IEEE 1500 is proposed that extracts test patterns from an external source, distributes the patterns to cores under test, and collects and compares the responses. It also generates the required control signals for the test procedure automatically. With this test platform, high speed and low-cost testing has been achieved for 2D designs.

In [14], the authors proposed test scheduling algorithms to compute optimal session-based or session-less test schedules in an IEEE 1687 architecture that consider resource conflicts between tests in a session and power constraints.

In this work, we present an architecture for autonomous testing of 3D-ICs with the following features:

1. A hardware-efficient test architecture for 3D-ICs based on IEEE 1687 for test access configuration and IEEE 1500 for core and die test with low access and reconfiguration time.
2. A controller for autonomous testing which extends the work of [7] to support IEEE 1687 so as to facilitate the required test access modes for 3D-IC testing. The architecture allows both ATE-driven test for high-volume production test as well as autonomous in-field test. Patterns can be accessed from external equipment such as a generic ATE or a low-cost ATE, or directly from a storage device such as FLASH or SD-card (for autonomous in-field testing).
3. The accessed test data can be formatted and applied to the CUTs directly, or buffered in shared system memory or local memory using a direct memory access (DMA) mechanism and then applied to the CUTs. The required memory space can be reduced by splitting the test into test sessions. The required sessions and test access configurations across the dies are setup using IEEE 1687.

4. A modified core test wrapper architecture to reduce the area overhead of a dedicated die-level wrapper for TSV test with low access time.
5. Support of wrapper scan cells with one or two flip-flops depending on the required test modes.

The next section describes the proposed architecture and its components. Section III explains the autonomous test procedure in the 3D-IC. Experimental results are discussed in Section IV, followed by the conclusion.

II. DFT ARCHITECTURE FOR AUTONOMOUS 3D-IC TEST

A. 3D-IC DFT architecture based on IEEE Std. 1687

Figures 1 shows the 3D DFT architecture for a stack of three dies. *Die0* at the bottom and *Die1* in the middle are both logic dies, and *Die2* on the top is a memory die. Every die contains four test components: 1) an IEEE 1687 based Scan Path Control unit (called "remotely controlled scan mux architecture" in IEEE 1687 [2]), 2) the scan chains of the cores under test that are connected into a number of parallel daisy-chains, 3) a TAP-controller (TAPC) to control the parallel daisy-chains and Scan Path Control unit, and 4) two top-level multiplexers (T0, T1) to determine the test data paths. The bottom die further contains a test controller, called Test Access Mechanism controller (TAMC).

The parallel daisy-chains are constructed by serially connecting the parallel scan chains of cores in each die with multiplexers (see Figure 1). Note that we add bypass registers on the bypass paths to avoid a long combinational path. The multiplexers and test enable signal of cores are controlled by the Scan Path Control unit. This allows to setup any combination of parallel scan chains of cores under test in a test session by shifting appropriate configuration setup data into the Scan Path Control unit and enables flexible test schedules in a complex die since the tested and non-tested cores can be quickly and centrally changed. By use of IEEE 1687 compliant structures, this setup can be easily controlled and the required shift patterns can be generated/retargeted at higher level using the procedure description language PDL [2] and EDA tool support.

The architecture supports both pre-bond and post-bond testing. For the post-bond test, the top-level multiplexers T0 and T1, which control the flow of test data from daisy-chains of cores at die-level, are added and controlled by the Scan Path Control unit. With these two multiplexers, we can configure the following 3D-IC test paths: die-bypass, elevator, or turnaround [2]. The cores in a die can be either included or bypassed in the configured test path. When a die is in the elevator mode, the test data will be forwarded to the die above and test responses will be received from the upper die. In the turnaround mode, the test data will not be delivered to the upper die, but turned back to the lower die (either including or bypassing the current die). Table 1 summarizes the supported 3D-IC post-bond test modes of the proposed architecture and the corresponding multiplexer control signals.

The TAP-controller (TAPC) in each die is used to control the JTAG-compliant operations including shift, capture and update. The port signals TAP-CTRL in Figure 1, which come from either the TAMC or probe pads, include TCK, TMS, and TRST.

Table 1 Supported 3D-IC post-bond test modes and control signals of multiplexers

Mode	T0	T1
DieBypass-Turnaround	1	0
DieBypass-Elevator	0	0
DieInclude-Turnaround	1	1
DieInclude-Elevator	0	1

The bottom die contains the TAMC. The TAMC can provide parallel test data through TAM wires to the cores under test and generate the required control signals for the whole test procedure of the stacked 3D-IC. With this controller, the test data can be read from an external source and buffered in the chip. Then the test application and examination procedures can be carried out autonomously without using external test equipment.
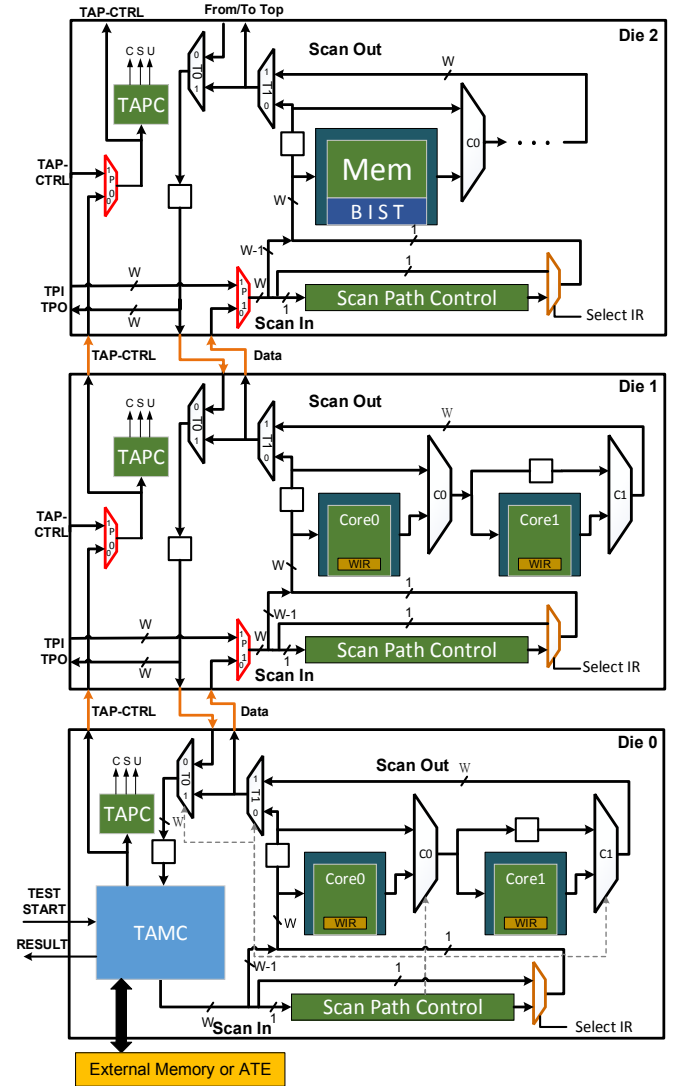

Figure 1. 3D DFT architecture for autonomous 3D-IC test

Since IEEE 1687 is a serial access architecture (except for a broadcast feature), it cannot be directly used as parallel TAM here. To distribute the parallel test data to the IEEE 1500 test wrappers of the cores, we use a dedicated parallel TAM and

control it by IEEE 1687 scan registers. IEEE 1687 is also used to configure the core wrappers.

In the upper dies, we add JTAG-compliant probe pads for the pre-bond test mode and two additional multiplexers (P0, P1) controlled by the automatic die detection mechanism proposed in [4] to switch the test path between pre-bond (P0=1, P1=1) and post-bond (P0=0, P1=0) test modes. Notice that the TPI and TPO are parallel I/Os in order to provide multiple-bit test data in pre-bond mode.

For memory testing, we can also use the proposed architecture to access and enable the memory BIST and shift out the result after the test is finished.

B. Scan Path Control unit

The main idea of reconfigurable scan networks based on IEEE 1687 is to use control registers and multiplexers on the scan paths so as to configure the scan paths in each die during testing. One widely used implementation based on IEEE 1687 is the Segment Insertion Bit (SIB)-based hierarchical structure, in which each SIB controls the access to the scan segments or chains of a core under test, and the SIBs themselves are part of the scan path. In [2], a "remotely controlled scan mux architecture (RCSMA)" is also suggested, which can reduce the number of shift cycles when accessing the scan chains compared to a SIB-based design. The basic concept of the RCSMA is shown in Fig 2.

In this work, we adapt the RCSMA and call the control register for the scan paths on a die the *Scan Path Control unit*. The Scan Path Control unit and the scan chains in the cores of each die are on two separate paths. One can use a select signal to switch the active scan path between Scan Path Control and the scan chains of the cores. Before shifting in the test patterns, one has to select the Scan Path Control and shifts in appropriate setup data to determine the configuration of the test path. After setting up the configuration, the test patterns can be shifted into the desired scan chains for test application.
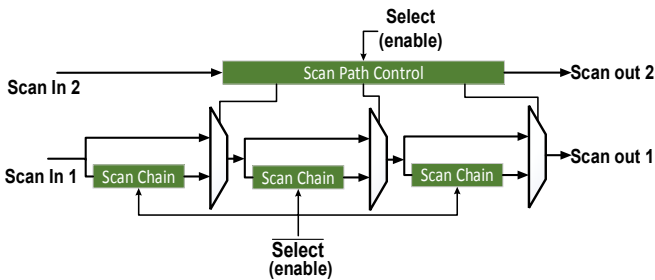

Figure 2. Remotely controlled scan mux architecture (RCSMA)

Our test architecture is based on the RCSMA because it is easier to control with the TAMC. However, with the RCSMA one needs 2 scan paths and one additional control signal, both leading to the increase of TSVs and probe pads (for pre-bond test) in 3D-ICs. In this work we address this problem such that no additional pin is needed. We modify RCSMA by reusing the selectIR signal from TAPC and sharing the test access mechanism (TAM) wires of the CUTs. Figure 3 shows how the Scan Path Control unit is operated after the modification. The Scan Path Control unit is placed in front of the first TAM wire,

TAM_0. When the TAPC is in the configuration stage (SelectIR=1), the Scan Path Control unit will be activated such that the configuration setup data can be shifted in serially and stored in the update register by an update operation. When the TAPC is in the test data application stage (SelectIR=0), the Scan Path Control unit is bypassed so it does not affect the pattern load/unload.

These configuration setup data determine which cores the test data will be delivered to (or bypassed) and which 3D-IC test path at die-level will be used. The length of the Scan Path Control unit depends on the number of cores and the test modes to be used in each die. Assume that the number of cores is N, then the length of the scan path control unit will be $N+3$ in this case, where the extra three bits denote the control signals of multiplexers T0 and T1, and the signal to enable the TSV testing mode of core wrappers as explained in Section II.D.
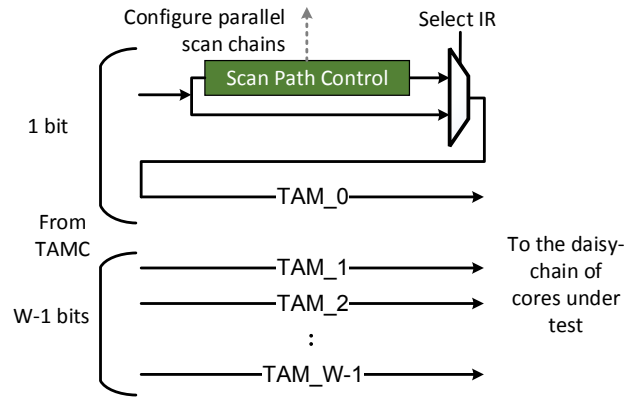

Figure 3. Scan path control unit and TAM signals

C. Test access mechanism controller (TAMC)

The TAMC is essential for autonomous testing in this work. It is connected to the cores under test through IEEE 1687. It can read deterministic test patterns from an external ATE through probe pads or pins in the manufacturing phase or from external memory like an SD-card through the system bus for in-field test. The TAMC distributes the patterns to the cores under test, and collects and evaluates test responses. It generates all the required control signals for the test architecture and test procedure on-chip. Upon test completion, the final results can be read and displayed, for instance, using an LED screen for in-field testing. Even in the pre-bond phase, we can still use a less complex ATE to send the plain test data to the TAMC through pads and test the bottom die.

With the TAMC, we achieve the following features:

1. Autonomous and at-speed testing.
2. The requirements on external ATE are significantly reduced for manufacturing test and can be eliminated for in-field testing.
3. Only a few pins are needed for accessing the TAMC. No extra test pins are required for parallel scan chains.

More information of the TAMC can be found in [7].

D. Modification of IEEE 1500 wrapper scan cells to test TSVs

In order to test the TSVs, 3D-ICs usually have a die-level wrapper or boundary scan chain in each die [1] so that every TSV or I/O can be easily accessed and tested. In a core-based SOC environment, there may already exist cores with core-level wrappers and some I/Os of cores may also connect to TSVs. If the core-level wrapper scan cells can be reused to access TSVs, the die-level wrapper may not be necessary, or the number of wrapper cells can be reduced.

In addition, the boundary scan chain through a core-level wrapper may be very long. To shorten the test access time to TSVs accessible from a core-level wrapper, we can bypass the cores that do not connect to TSVs using IEEE Std. 1687 structures. Figure 4(a) shows a core-level boundary scan chain. The red blocks denote I/Os that are connected to TSVs. To reduce the access time to the red scan cells, we can add multiplexers to bypass the cells that do not connect to TSVs as Figure 4(b) shows. When the TSV test signal from the Scan Path Control unit is active, the modified core-level wrapper only chains the boundary scan cells that connect to TSVs. Otherwise, the boundary scan chain contains all cells. In this way, we reuse the core-level wrappers for TSV testing and reduce the area and performance overhead of additional die-level wrappers. Furthermore, to reduce the area overhead we can use the one-flip-flop boundary scan cell as defined in IEEE Std. 1500. For complex test patterns, e.g., for TSV crosstalk faults, two-flip-flop boundary scan cells can be added individually. Figure 5 shows the two boundary scan cells.
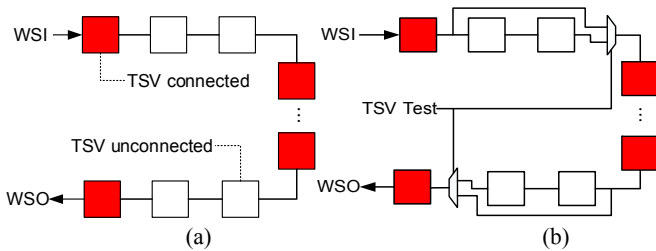

Figure 4.(a) Original boundary scan chain (b) Modified boundary scan chain with bypass for cells not associated with TSVs
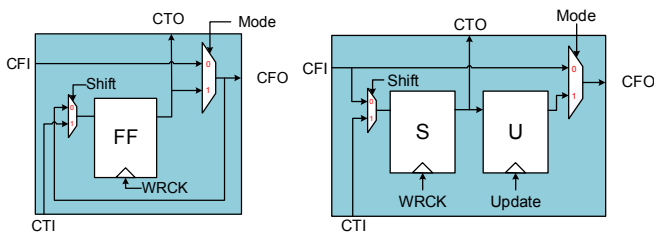

Figure 5. Wrapper scan cell with one or two flip-flops

III. TEST PROCEDURE

This section explains the test procedure executed by the TAMC. The data for the test application is stored in an external source (external memory or ATE). This data comprises setup data for the IJTAG architecture, test patterns, masks and expected test responses. The TAMC loads the setup information of the current test session from external source through the system bus or pins. These TAMC setup data include the number of test patterns, location of test patterns of the test session and shift length of the configuration. Then the TAMC resets the DFT architecture, applies the scan path configuration and starts reading and applying test data for the test session according to the setup information.

The test procedure contains a number of test sessions each of which consists of three steps: 1) configuration setup, 2) instruction register setup, and 3) test data application.

In the configuration setup step, the active scan path only contains the Scan Path Control unit in each die. The TAMC shifts in the configuration setup data into the Scan Path Control unit and captures it to the update register of the unit to setup the configuration of the current test session.

After the configuration setup, the configuration has been defined. The instruction register setup step is used to shift instructions into the test wrappers of each core under test and define their modes. Because the Scan Path Control unit shares the selectIR signal in our design in order to reduce test pins and TSVs (see Figure 3), the active scan path must contain the Scan Path Control unit and the instruction registers of wrappers of cores under test in each die for the current test session.

Finally, the TAMC starts to shift test patterns into the parallel scan chains of the cores under test and collects the test responses which are shifted out through the configured test path in the die stack. At the same time, the TAMC also reads the mask and expected responses and compares them with the captured responses.

The TAMC repeats these steps until every test session has been completed. The final test results can be output to external devices such as ATE or an LED display.

IV. EXPERIMENTAL RESULTS

In this section, we quantify the area overhead of the proposed 3D DFT architecture, estimate the required test time, and report the obtained area and test scheduling results for a constructed experimental 3D-IC. For this 3D-IC, we compare the test time overhead of a SIB-based test architecture and the proposed DFT architecture.

The area overhead A_o for the IEEE Std. 1687 structures in our 3D DFT architecture can be estimated using the following equation: $A_o = f + W \times T_p + N \times (W \times C_p + C_r)$ (1), where f denotes the fixed area including TAP-controller, multiplexer P0, and the fixed number of register bits in the scan path control unit. W and N denote the TAM width and the number of cores in a die. T_p, C_p and C_r represent the area of top-level paths (including multiplexers T0, T1, P1 and die bypass registers), core-level paths (including multiplexers C_n and core bypass register of an individual core), and control bits of cores in scan path control unit, respectively. A_o depends on the number of cores and the TAM width, but is independent of the size and number of pins of cores.

In this work, we use a TSMC 90nm cell library to synthesize the structure. The area in gate count is as follows: $f = 165$, $T_p = 12.9$, $C_p = 8.6$, $C_r = 11$ and the gate count of the test access mechanism controller (TAMC) in the bottom die is 26525. Table 2 shows the area overhead of the DFT components for different

TAM widths (*W*) and number of cores (*N*). For medium or large circuits, the area cost is relatively small.

Table 2. Area overhead (in gate count) of IEEE Std. 1687 DFT components in one die for different TAM widths W and numbers of cores N (without TAMC)

N \ W	5	10	20	50	100
1	275.9	373.9	569.9	1157.9	2137.9
4	443.6	670.6	1124.6	2486.6	4756.6
8	667.2	1066.2	1864.2	4258.2	8248.2
16	1114.4	1857.4	3343.4	7801.4	15231.4

We now analyze the number of test cycles in the three steps of each test session. Assume that D is the number of dies included in a test session, N_i is the number of cores in the ith die. L_{ij} and Lr_{ij} denote the length of the scan chains and the length of instruction register of the jth core in the ith die, respectively, and Ls_i denotes the scan length of scan path control unit in the ith die. The test cycles of the three steps discussed in Section III can be estimated by the following equations:

$$\sum_{i=0}^{D-1}(Ls_i+2)+u \quad \text{(Step 1)}$$

$$\sum_{i=0}^{D-1}\left[Ls_i+d_i\sum_{j=0}^{N_i-1}(c_{ij}\cdot Lr_{ij}+\overline{c}_{ij}\cdot 1)+\overline{d}_i\cdot 1+1\right]+u \quad \text{(Step 2)}$$

$$\left\{\sum_{i=0}^{D-1}\left[d_i\cdot\sum_{j=0}^{N_i-1}(c_{ij}\cdot L_{ij}+\overline{c}_{ij}\cdot 1)+\overline{d}_i\cdot 1+1\right]\right\}(P+1)+P\cdot u \quad \text{(Step 3)}$$

where c_{ij} and d_i are Boolean variables representing whether the cores or dies are included in the scan path. P and u denote the numbers of test patterns in a test session and number of cycles of update and capture operations. The constant in each equation is due to the bypass register in each path. For a large value of P, the test cycles in Steps 1 and 2 are negligible, since L_{ij} is much larger than Lr_{ij} and Ls_i. For TSVs testing, we only need to replace the parameter L_{ij} in Step 3 with the length of boundary scan chains for TSVs access.

Note that all of the test data (including the setup data) can be provided to the chip using external test equipment for volume testing or directly from storage devices such as FLASH or SD card to facilitate autonomous in-field testing.

In the following experiment, we construct a two-die 3D-IC using ITC'99 and IWLS'05 benchmark circuits. The information and arrangement of these circuits is shown in Table 3. The columns of the table denote the name of cores, number of I/O ports, number of scan flip-flops, maximum length of scan chains,

Table 3. Benchmark circuits and arrangement in 3D-IC example

Core	# I/O	#SFF	SC_len	#Pattern	FC %
b19	51	6642	595	1069	99.59
b18	60	3320	298	764	100
b17	134	1415	129	669	99.98
des_perf	297	8808	801	55	97.48
ethernet	210	10544	959	674	99.53
Arrangement of Dies					
Die	Cores		Area	Max SC_len	
0	b19, b18, b17		273507	1023	
1	des_perf, ethernet		218549	1761	

number of test patterns, and fault coverage respectively. The patterns are generated using Synopsys TetraMax. We wrap the cores using IEEE Std. 1500 wrappers and add the required multiplexers such that they can be integrated into the proposed architecture for autonomous testing. The width of the TAM is twelve bits, including a boundary scan chain of the core wrapper.

Circuits b19, b18, b17 are put in the bottom die with the TAMC test controller, and circuits des_perf and ethernet are put in the upper die. The area information is shown in Table 4. The columns in Table 4 denote the area of the scan inserted cores, the area of the core wrappers in gate count, and their percentage in the whole design respectively. Rows IJTAG_bot and IJTAG_up are the areas of the IJTAG architectures in the bottom die and upper die, respectively. The area overhead of the wrapper DFT and the IJTAG architecture is very small compared to the area of the cores. The area overhead of the TAMC test controller is 5.39%. The total area overhead of the proposed DFT architecture including the TAMC is about 7.69%. Since the area of the TAMC does not increase with the number or size of cores or dies, the area overhead can be even lower for large designs, e.g., less than 1% for a 4M-gate design.

Table 4. Area information for the 3D-IC example

Process : TSMC90nm		Unit : gate count		
Core	Area	Wrap or DFT Area	Area (%)	Wrap or DFT (%)
b19	144746	912	29.42	0.18
b18	72228	1055	14.68	0.21
b17	25524	1849	5.19	0.38
des_perf	109558	3621	22.26	0.74
Ethernet	102154	2667	20.76	0.54
IJTAG_bot	-	668	-	0.14
IJTAG_up	-	549	-	0.11
TAMC	-	26525	-	5.39
Sum	454210	37846	92.31	7.69
Total	492056		100	

We also calculate the number of test cycles of this design example based on the three test cycle calculation equations given earlier in this section. For simplicity, we employ the optimized session-less (OSL) scheduling algorithm proposed in [14] without power and resource constraints to create the desired test sessions. In this session-less schedule, the tests do not have to start in a synchronized manner, i.e., two cores with different numbers of test patterns can start their testing in one session and after the one with fewer test patterns is fully tested, the one with more test patterns can be further tested in other test session(s).

Table 5 shows the details of the computed test sessions for post-bond or in-field test. The columns in Table 5 denote the test sessions, tested cores, number of patterns and test cycles in different steps mentioned above. The results show the test cycle overhead of the proposed architecture (Steps 1 and 2 of the equations above), which comprises the additional cycles induced by the proposed test architecture during the test procedure for the setup and configuration of the test data paths. This overhead is very small compared to Step 3, the actual application time of the pattern data.

Table 5. Test sessions and test cycles

	Session	#Patterns	Cycles step (3)	Cycles step (1)+(2)
1	b19, b18, b17, des_perf, ethernet	55	156179	58
2	b19, b18, b17, ethernet	614	1223230	55
3	b19, b18, ethernet	5	11161	52
4	b19, b18	90	81895	34
5	b19	305	184513	31
	Total test cycles		1657208	
	Percentage (%)		99.99	0.01

We also estimate the reconfiguration overhead if the architecture uses SIB (Segment Insertion Bit, [2]) bypasses instead of the central Scan Path Control unit per die. Table 6 compares reconfiguration overhead in cycles of the SIB-based and the proposed Scan Path Control unit based architectures. The overhead of the SIB-based architecture is 35 times larger than in our architecture. This is because in the SIB-based architecture *every* pattern is shifted through additional SIB registers on the scan paths, which causes a relatively high overhead. In the proposed architecture, we only need to shift the data through the Scan Path Control unit twice per test session.

Table 6. Test cycle overhead for reconfiguration

3D DFT Architecture	Test Cycle Overhead	%
SIB-based	8473	100
Proposed architecture	230	2.71

Table 7 compares the test cycles of the proposed reconfigurable architecture and an architecture without reconfigurability, in which the test configuration cannot be changed during the whole test procedure. The results show that the proposed reconfigurable architecture reduces the test time by about 44.4% in this design. The reduction of test cycles is due to bypassing of the cores that already finished their tests and the dynamic minimization of the scan chain length in the tests. In a complex system with many cores with different test parameters, e.g., number of test patterns and length of scan chains, it is useful to use a reconfigurable scan architecture to dynamically change the scan paths for an optimal test procedure.

Table 7. Test time comparison

Architecture	Test Cycles	%
Without reconfigurability	2982085	100
Proposed architecture	1657208	55.6

The proposed architecture allows to execute an autonomous test of the whole 3D-IC with high quality stored test patterns. For the considered 3D design, this test completes in 1.66 million cycles, corresponding to a time period of only a few milliseconds.

V. CONCLUSIONS

Safety-critical systems require in-field testing in addition to thorough manufacturing test. In this work we describe a 3D-IC test architecture that employs features of the recent IEEE Std.

1687 for scan chain reconfiguration, modify IEEE Std. 1500 wrappers for high efficiency parallel scan and TSV test, and employ an embedded controller for autonomous in-field testing. It can efficiently and flexibly execute the test of a 3D-IC with very little or even no use of external test equipment so as to reduce test cost. The advantages of such a test architecture include: 1) facilitation of in-field autonomous testing; 2) support of highly flexible test scheduling via IEEE Std. 1687 structures; 3) low area overhead; 4) very small test cycle overhead for reconfiguration; and 5) support of pre-bond, post-bond and TSV testing.

REFERENCES

[1] E.J. Marinissen, "Challenges in testing TSV-based 3D stacked ICs: Test flows, test contents, and test access," in *Proc. 2010 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS),* pp. 544-547, 6-9 Dec. 2010.

[2] IEEE Standard for Access and Control of Instrumentation Embedded within a Semiconductor Device," *IEEE Std 1687-2014*, pp.1-283, Dec. 5 2014.

[3] Y. Fkih, P. Vivet, M. L. Flottes, B. Rouzeyre, G. D. Natale and J. Schloeffel, "3D DFT Challenges and Solutions," in *Proc. 2015 IEEE Computer Society Annual Symposium on VLSI (ISVLSI),* pp. 603-608, 8-10 July 2015.

[4] Y. Fkih, P. Vivet, B. Rouzeyre, M. l. Flottes and G. Di Natale, "A JTAG based 3D DFT architecture using automatic die detection," *IEEE PRIME,* pp. 341–344, 2013.

[5] E.J. Marinissen, C.-C Chi, M. Konijnenburg, J. Verbree, "A DfT Architecture for 3D-SICs Based on a Standardizable Die Wrapper," *Journal of Electronic Testing: Theory and Applications (JETTA)*, vol. 28(1), February 2012.

[6] R. Baranowski, M.A. Kochte, H.-J. Wunderlich, "Modeling, verification and pattern generation for reconfigurable scan networks," in *Proc. IEEE Int'l Test Conference (ITC),* paper 8.2, pp.1-9, 5-8 Nov. 2012.

[7] K.-J. Lee, T.-Y. Hsieh, C.-Y. Chang, Y.-T. Hong and W.-C. Huang, "On-Chip SOC Test Platform Design Based on IEEE 1500 Standard," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems,* vol.18, no.7, pp.1134-1139, July 2010.

[8] M. Keim, T. Waayers, R. Morren, F. Hapke and R. Krenz-Baath, "Industrial Application of IEEE P1687 for an Automotive Product," in *Proc. Euromicro Conf.* on *Digital System Design (DSD),* pp. 453-461, 4-6 Sept. 2013.

[9] T. Payakapan, S. Kan, K. Pham, K. Yang, J.-F. Cote, M. Keim, J. Dworak, "A case study: Leverage IEEE 1687 based method to automate modeling, verification, and test access for embedded instruments in a server processor," in *Proc. IEEE Int'l Test Conference (ITC),* pp. 1-10, 2015.

[10] Y. Li, S. Makar, S. Mitra: "CASP: Concurrent Autonomous Chip Self-Test Using Stored Test Patterns." in *Proc. Design Automation and Test in Europe Conf. (DATE),* 2008, pp. 885-890.

[11] K.-J. Lee, C.-Y. Chu, Y.-T. Hong, "An embedded processor based SOC test platform," in *Proc. ISCAS,* 2005, pp. 2983-2986.

[12] H. Lee, K. Chakrabarty, "Test Challenges for 3D Integrated Circuits," *IEEE Design & Test of Computers*, vol. 26, no. 5, pp.26-35, 2009.

[13] E.J. Marinissen, T. McLaurin, H. Jiao, "IEEE Std P1838: - DFT Standard-under-Development for 2.5D-, 3D-, and 5.5D-SICs," in *Proc. European Test Symposium, 2016.*

[14] F. G. Zadegan, U. Ingelsson, G. Asani, G. Carlsson and E. Larsson, "Test Scheduling in an IEEE P1687 Environment with Resource and Power Constraints," in *Proc. Asian Test Symposium*, 2011, pp. 525-53