ՀԱՅԱՍՏԱՆԻ ԳԻՏՈՒԹՅՈՒՆՆԵՐԻ ԱԶԳԱՅԻՆ ԱԿԱԴԵՄԻԱ
НАЦИОНАЛЬНАЯ АКАДЕМИЯ НАУК АРМЕНИИ
NATIONAL ACADEMY OF SCIENCES OF ARMENIA
ДОКЛАДЫ ԶԵԿՈՒՅՑՆԵՐ REPORTS

Հատոր Том Volume 118 2018 № 1

INFORMATICS

УДК 519.688

G. E. Harutyunyan

An Approach for Scheduling Parallel and Serial Testing of Embedded IP Cores in Nanoscale SoCs

(Submitted by academician S.K. Shoukourian 22/I 2018)

Keywords: *hierarchical test, system-on-chip, test pattern, test scenario, pattern porting.*

1. Introduction. Variety and complexity of used memories and IP cores, shrinking technologies and design complexity increasing in nanoscale systems-on-chips (SoC) make it crucial to have embedded in SoC test and repair solutions kept up with the advances in order to consistently and continuously provide chip quality and yield optimization. The embedded test approaches developed for designs just few years ago are not sufficient for today's designs, which are bigger, faster, hierarchical and much more sensitive to area, timing and power [1, 2]. Similarly, the embedded test solutions developed, for example, for 28-nm technology node will not deliver the same level of test quality, diagnosis accuracy and repair efficiency for 14-nm and below technology nodes, as defects and failure mechanisms change with process technologies shrink. From the other side, nanoscale SoCs are re-using multiple already designed sub-chips which means that multiple test infrastructures might be organized in some test hierarchy built according to suggested by the SoC developer [3, 4].

Figure 1 shows the evolution of the SoC test infrastructure during the last several years. The part (a) of the Figure 1 shows the stage when there was only one BIST (built-in self-test) scheme per SoC, while in Figure 1 (b) there are multiple BIST schemes and, the test configuration for the SoC has only one Server. This means that though multiple BIST schemes are used but SoC does not require a hierarchy of Servers. The part (c) of the Figure 1 shows modern SoCs with many memories and IP cores as well as containing a hierarchy of Servers, where there is a Server at top and there are Sub-Servers at the second level of hierarchy.

Usually, different approaches and standards are used for IP testing and integration into SoC and, at the same time, at the chip level the total number of

test channels is limited such that all core-level test channels cannot be accessed at the same time.

In general, hierarchical test gives designers flexibility to schedule test of individual interface IP cores and other cores for parallel and serial testing to optimize test time and power consumption during test [2, 5]. The flexible test schedule can significantly reduce test time, especially for designs with a large number of high-speed I/Os.

From the other side, there are certain limitations in SoC which should be taken into account when scheduling the test. Common types of SoC limitations are the following:

- Design limitation – e.g., limited number of test access mechanisms to test multiple IP cores;
- Test limitation – e.g., precedence constraint (e.g., test of IP2 should be run only after completing the test run on IP1);
- Resource limitation – e.g., power constraint (limitation on SoC consumed power when testing multiple IP cores in parallel).

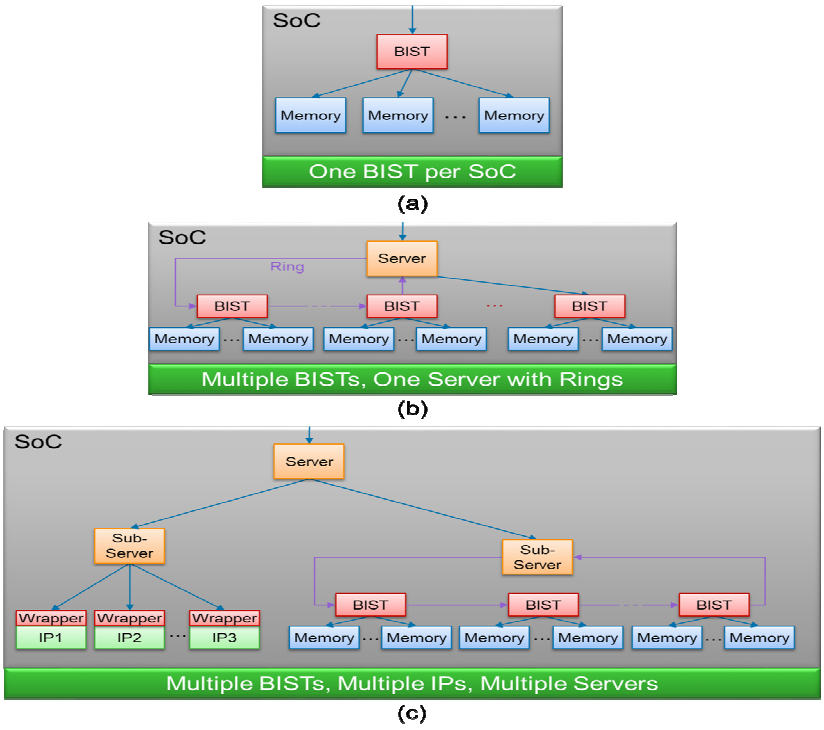

Fig. 1. SoC and its evolution

In this paper, a hierarchical test approach is proposed which takes into account all the above mentioned limitations and allows to do efficient test scheduling.

2. Hierarchical test system. Figure 2 shows a hierarchical test system [2-4] which is used to do scheduling of parallel and serial testing of IPs in SoC. At IP level it is based on IEEE 1500 standard [6] which provides unified access to different types of IP cores. At top level the test system consists of main Server and multiple Sub-Servers placed at the second level of hierarchy. Top level Server is connected to IEEE 1149.1 JTAG interface [7] which is in charge of providing test patterns from the outside world. Usually SoCs are consists of Sub-Chips and the test patterns applied to a Sub-Chip can be reused at top level by porting those test patterns from Sub-Chip level to SoC level.

The considered hierarchical test system has the following main capabilities:

- Unified test accessibility for different IP cores in SoC based on existing test standards (IEEE 1500 [6], IEEE 1149.1 (JTAG) [7], IEEE 1687 (IJTAG) [8]);
- Pattern porting from IP and Sub-Chip level to SoC level which allows to reduce the design and test times;
- Language for describing the structural models of memories and IP cores;
- Capability to create effective test scenarios under the presence of limited resources available in SoC.

The mentioned above language for structural models provides a set of parameters that are necessary to describe the structure of a given IP which is comprised of the following sections:

- Port description – Name of ports and attributes (function, direction, range, etc.);
- Core internal and external registers;
- Test patterns;
- Comments – line (//…) and block (/* … */) comments are supported.

3. Scheduling of Parallel and Serial Testing of IPs in SoC. Within the concept of the hierarchical test scheduling, in nowadays complex SoCs the test time is one of the important challenges for which usually concurrent test is used to minimize the test time. For thousands of cores in SoC comprised of multiple levels of hierarchy the following problem exists: **determination of an optimal scenario for concurrent test and its implementation in a test infrastructure.**

Figure 3 shows a ring architecture of the hierarchical test system which allows to do efficient test scheduling. Group of blocks connected serially is called ring. Sub-Server can have one or more rings (Ring 1, Ring 2, …, Ring K), and each ring can contain one or more blocks (e.g., Block 11, Block 12, …, Block $1N_1$), where a block can be an IP core or group of IPs (connected with hierarchical connections) or it can be another Sub-Server.

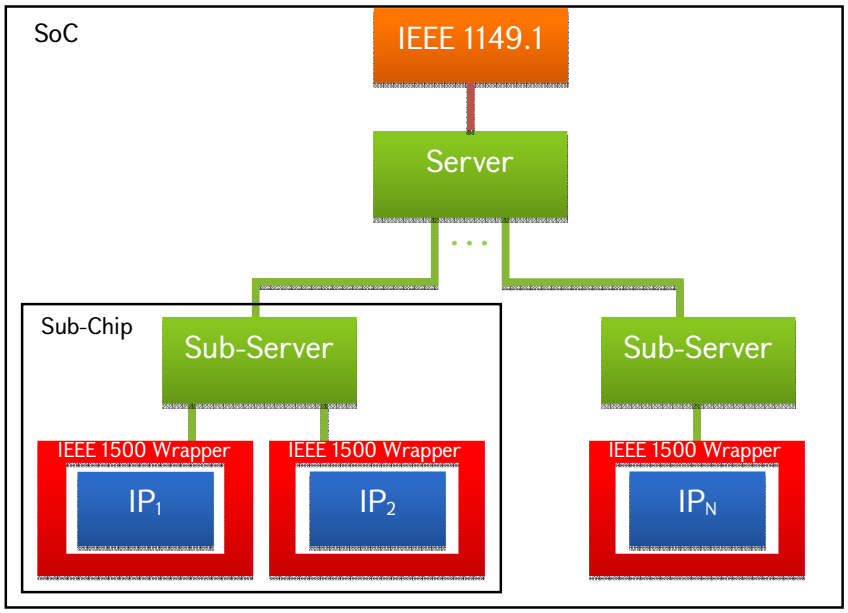

Fig. 2. Hierarchical Test System

When several IP cores are being tested in parallel, usually there is a limitation that the total consumed power should not exceed the given number (e.g., MAX_POWER) [9]. Figure 4 shows two scenarios for testing a given set of IP cores. Figure 4 (a) shows a non-efficient (poor) scheduling while in Figure 4 (b) an efficient scheduling scenario is shown. In both scenarios IP cores are divided to be tested in 3 sequential sessions where in each session the IP blocks

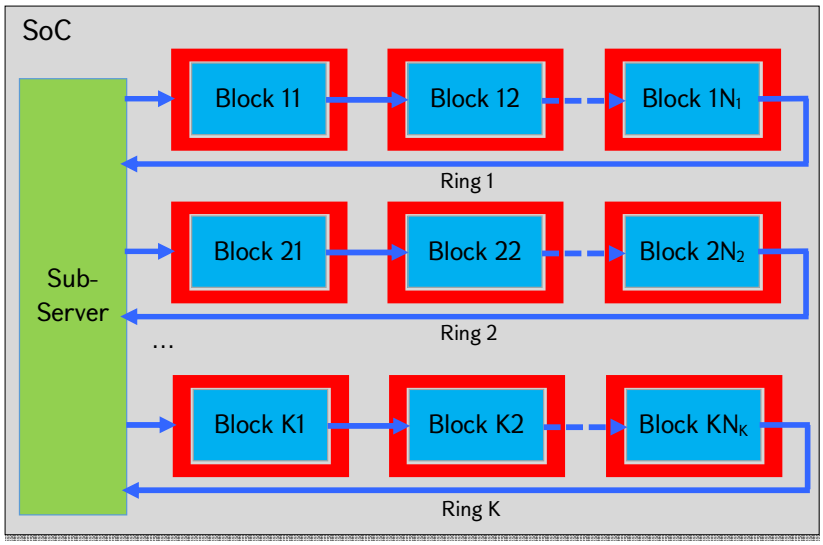

Fig. 3. Ring architecture

are being tested in parallel. In the figure, the «Idle time» shows how well or bad the available resources are used: better scheduling brings smaller «Idle time» which means the available resources are being used more efficiently and thus the overall test time is shorter.

Within the proposed architecture in Figure 3, the overall time for testing all the blocks is calculated in the following way: $T = T_{ring\_access} + T_{ring\_load} + T_{block\_test}$, where:

- T_{ring_access} – time needed to access rings;
- T_{ring_load} – time needed to load information into rings;
- T_{block_test} – time needed to test all the blocks.

Let assume there are K test sessions determined by one of well-known scheduling algorithms to get effective test concurrency for given design, test and resource limitations, e.g. Greedy/Rectangle Packing algorithm [10]. It is

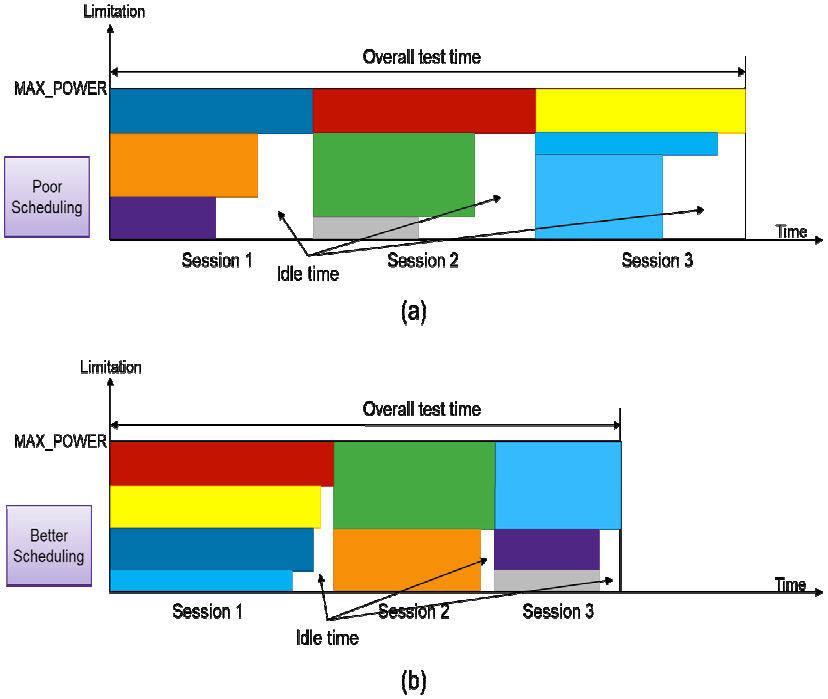

Fig. 4. Optimal scheduling scenario

proved that the following proposition is true: **it is necessary for the proposed test architecture to have K independent rings (one ring per session) to reach optimal test time.**

Proof. All other cases will lead to having non-optimal test time. There are the following two cases:

- If blocks of the same session are distributed in different rings, then for testing those blocks in parallel there is a need to access more than one ring, which increases the overall ring access time (T_{ring_access}).

- If a ring contains blocks from different sessions, then for testing blocks of one session there is a need to bypass the blocks that are in that ring but are out of that session, which will increase the overall ring load time (T_{ring_load}).

T_{block_test} depends on the scheduling algorithm and is independent from the test architecture, thus it is not impacted by how the blocks are distributed in the rings.

Experiments showed the same results, i.e., optimal test time is obtained when the blocks of the same session belong to the same ring and there are no other blocks in the ring.

4. Conclusions. In this paper, an approach for parallel and serial testing of embedded IP cores is proposed allowing:

- to take into account SoC design, test and resource limitations during scheduling;
- to provide capability for creation of optimal test scenarios under the presence of limited resources available in SoC.

Yerevant State University
e-mail: gurgen.harut@gmail.com

G. E. Harutyunyan

An Approach for Scheduling Parallel and Serial Testing of Embedded IP Cores in Nanoscale SoCs

Modern nanoscale chips are increasingly growing and involve more memories and other design blocks. As a result, the test process of such chips becomes essentially difficult. In this paper, an efficient approach is proposed which provides capability to create optimal test scenarios under the presence of limited resources available in SoC.

Գ. Է. Հարությունյան

Պլանավորման մոտեցում նանոչափական բյուրեղներում ներկառուցված նախագծման բլոկների զուգահեռ և հաջորդական թեստավորման համար

Արդի նանոչափական բյուրեղները (չիպերը) աստիճանաբար մեծանում են՝ ընդգրկելով ավելի շատ հիշող սարքեր և այլ նախագծման բլոկներ։ Արդյունքում էապես դժվարանում է այդ բյուրեղների թեստավորման գործընթացը։ Այս աշխատանքում առաջարկված է մի արդյունավետ մեթոդ, որը հնարավորություն է տալիս ստեղծելու օպտիմալ թեստային սցենարներ՝ բյուրեղում սահմանափակ ռեսուրսների առկայության դեպքում։

Г. Э. Арутюнян

Подход к планированию параллельного и последовательного тестирования блоков проектирования в наномерных системах на кристалле

С постоянным увеличением размеров и сложности наномерных систем на кристалле усложняется процесс их тестирования. Предлагается эффективный подход к планированию параллельного и последовательного тестирования блоков проектирования, который предоставляет возможность создания оптимальных тестовых сценариев в условиях жестких ограничений на используемые ресурсы.

References

1. *Han D., Lee Y., Kang S.* - ETRI Journal, 2014. V. 36, N 2. P. 293-300.
2. *Zorian Y., Shoukourian S.* - International Conference on Computer Science and Information Technologies. 2013. P. 1-3.
3. *Kogan T., Abotbol Y., Boschi G., Harutyunyan G., et al.* - IEEE International Test Conference. 2017. Paper 14.2. P. 1-6.
4. *Eychenne Ch., Harutyunyan G., Zorian Y.* - IEEE International Workshop on Automotive Reliability & Test. 2017. Paper S01-01. P. 1-4.
5. *Keller B., Chakravadhanula K., Foutz B., Chickermane V., et al.* - IEEE International Test Conference. 2014. P. 1-10.
6. *DaSilva F., Zorian Y., Whetsel L., Arabi K., Kapur R.* - IEEE International Test Conference. 2003. P. 988-997.
7. IEEE Std. 1149.1, IEEE Standard for Test Access Port and Boundary-Scan Architecture. 2001.
8. 1687-2014 - IEEE Standard for Access and Control of Instrumentation Embedded within a Semiconductor Device.
9. *Martirosyan L., Harutyunyan G., Shoukourian S., Zorian Y.* - IEEE East-West Design and Test Symposium. 2015. P. 27-30.
10. *Iyengary V., Chakrabarty K., Marinissen E. J.* - IEEE VLSI Test Symposium. 2002. P. 253-258.