A Power Based Memory BIST Grouping Methodology

L. Martirosyan, G. Harutyunyan, S. Shoukourian, Y. Zorian
Synopsys
lemartir@synopsys.com, gharutyu@synopsys.com, samshouk@synopsys.com, zorian@synopsys.com

Abstract

For System-on-Chips (SoCs) one of the most critical design constraints is power consumption. This paper presents memory built-in self-test (BIST) grouping methodology which takes into account the given peak power, power domains based on Unified Power Format (UPF) and optimal test time. The mentioned grouping criteria enable to perform power-aware memory BIST design at early stages of SoC design. To apply this methodology, there is a need for a method to estimate the power consumption from a design description of the circuit at high level of abstraction. We propose a fast power estimation methodology for register-transfer level (RTL) compilers which is based on linear and polynomial approximation. The obtained approximate functions are embedded in a script developed for automated estimation of power consumption. Memory BIST grouping methodology is based on and uses the data of power consumption estimation script as input information.

1. Introduction

Embedded memories are among the most common cores in SoCs. The annual report of International Technology Roadmap of Semiconductors ITRS expects that the number of memories would exceed 95% of the global area of a SoC [1]. These memories have different types and sizes. As a result testing these memories is increasingly difficult. Today BIST is considered as one of the best solutions for testing embedded memories thanks to its low cost and effectiveness [1].

Testing all the memories in these SoCs serially would take a long time. Therefore, a memory BIST design that allows two or more memories to be tested simultaneously is needed. So in case when the memories are tested in parallel (concurrently) the consumed power is the sum of the powers consumed by each memory. Hence, parallel run of memories by the same BIST scheme will consume much power which means that there could be a limitation on sharing multiple memories by the same BIST scheme. Total test application time of a memory BIST is also calculated under maximum power constraint [2]. Consequently, power consumption is one of the most critical constraints for SoCs. Moreover, the increasing functionality also raises complexity for specification, design and verification of SoCs. Therefore, power aware design should be introduced at early stages of SoC design where it has the highest benefits for power reduction [3]. Approaches such as power gating and adaptive voltage scaling are widely used to reduce power in SoCs. The most basic form of these approaches is to partition the chip logic into multiple voltage regions or power domains, each with its own power supply and power control unit. Unified Power Format (UPF) enables defining and verifying the power intent that represents the power-management architecture (set of power domains, power switches, supply networks, etc.) and strategy [4]. Another bottleneck of designing complex SoCs is a test time. Reducing the test time has become the focus of attention of designing complex SoCs [5].

One of the most important points of designing embedded memories is the early design planning phase, considering power and test time schedule with power constraint. There are many works published related to this topic (e.g., [1], [3]-[8]). [1] presents a generic method for designing shared memory BIST systems. It accepts arbitrary sharing rules for grouping memories under a wrapper, and it takes individual values of BIST component area, memory test time and memory peak power during test as parameters. [3] uses the new IEEE 1801 (IEEE Standard for Design and Verification of Low-Power Integrated Circuits) standard to describe power aware design. [4] proposes an efficient methodology for making system power decisions at transaction level by adding and verifying power intent and management capabilities into transaction level (TL) models. [5] proposes an efficient approach based on particle swarm optimization (PSO) algorithm for the test scheduling problem of core-based SoCs with power constraint. [6] presents an approach

978-1-4673-7776-8/15/$31.00 ©2015 IEEE

for wrapper/TAM (test access mechanism) co-optimization based on a combination of integer linear programming (ILP) and exhaustive enumeration. [7] proposes session-based and session-less test scheduling, and resource and power-aware test scheduling algorithms for the detailed scheduling types. [8] presents SoC test scheduling method based on simulated annealing which is used to find the optimal test schedule by altering an initial sequence pair and changing the width of the core wrapper.

In this paper we propose memory built-in self-test (BIST) grouping methodology taking into account the given peak power, power domains (based on UPF) and optimal test time. To apply this methodology, there is a need for a method to estimate the power consumption from a design description of the circuit at high level of abstraction. Accurate power consumption estimation is done with special tools [9], but in case of RTL compilers, when many instances can be generated, it is a time-consuming and inefficient task. In [10] we represent a fast power consumption estimation methodology. The method was applied to industrial compilers and maximum achieved estimation error was 10%. A fast power consumption estimation methodology is essential for implementation of memory BIST grouping methodology.

2. Methodology

Figure 1 shows memory BIST architecture. BIST architecture is generally composed by a certain number of memory BISTs. Each BIST can test a certain number of memories.

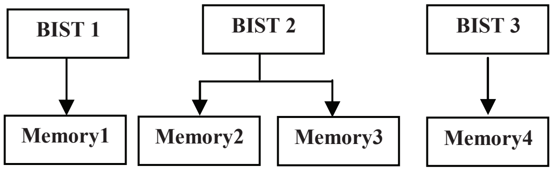

Figure 1. Memory BIST architecture

There are two memory connection methods for memory BIST logic sharing: parallel connection and serial connection. In Figure 1, Memor2 and Memory3 can be tested in parallel or serial mode. BIST1, BIST2 and BIST3 also can work in parallel or in serial mode.

In this section we will describe memory BIST grouping methodology taking into account the given peak power, power domains (based on UPF) and optimal test time.

Memory BIST grouping methodology taking into account the given peak power: In case BIST is testing memories serially (in the sequel will call it sequential BIST mode), the consumed power P_s is calculated by equation (1) and for parallel mode (in the sequel will call it parallel BIST mode) the consumed power P_P is calculated by equation (2) (see [1]).

$$P_S = Max\{P_i\}, i \in \{1,2,...,k\}, P_S <= P_{peak} (1)$$

$$P_p = \sum_{i=1}^{k} P_i , P_p <= P_{peak} (2)$$

where P_{peak} is the given peak power, P_i is the power consumed by memory i, k is the total number of memories.

Memory BIST grouping methodology taking into account optimal test time: Test time T_S for sequential BIST mode is calculated by equation (3), while test time T_P for parallel BIST mode is calculated by equation (4) (see [1]).

$$T_S = \sum_{i=1}^{k} T_i (3)$$

$$T_P = Max\{T_i\}, i \in \{1,2,..., k\}(4)$$

where T_i is the test time for memory i and k is the total number of memories.

Memory BIST grouping methodology taking into account power domains (based on UPF): In real world it is essential to save power in parts of chip that are not in use. Chips integrate several systems on a single chip (SoC). In order to save current consumption, each Intellectual Property (IP) can move between power modes (power-off, power-on, etc.). Each SoC is divided into power domains and those power domains can be turned on and off as well, according to the power-mode. The isolation cells keep the turned off IP outputs in a previously defined value, and this is how the shut-down IP does not corrupt other active IP functionality. The UPF [11] enables designers to easily specify the power intent early in the design process (e.g., in the RTL compiler) [12].

The power intent specification consists of power domains, supply networks, and power states.

Power Domains: The basic unit of power management strategy is the power domain. It is a collection of instances of the user design that share the same primary supplies, like power and ground. UPF provides a command *create_power_domain* for creating a power domain and associating the design instances with the extent of that power domain [12].

Supply network: In order to specify the low power design constraints, it is required to specify a power supply network that can control the distribution of that supply to minimize the energy consumption. Using UPF, one can easily specify the network at an abstract level. This network consists of supply ports, supply nets, and power switches, and is a high-level abstraction of the electrical network of the chip. Supply ports provide the supply interface to the power domain and switches, whereas supply nets connect the supply port [12].

Power states: Any power management strategy starts by defining the list of power states. These power states determine the mode of operation of the domains and their corresponding simulation semantics [12].

The proposed fast power consumption estimation methodology [10] enables to estimate the power consumption of scheme generated by RTL compilers for chosen values of parameters. Memory BIST grouping methodology takes information about power consumption from power consumption estimation script.

3. Implementation details

In [10] we have presented a fast power consumption estimation methodology for BIST architecture of embedded memories. Power consumption depends on functional and scalability parameters of memory BIST. Power consumption consists of two main components static power and dynamic power. The impact of changes due to static power is small hence we explore only the impact of changes by dynamic power. Experimental results are obtained through logic synthesis. The sets of parameter values are used as interpolation points (see Figure 2). As the analytical representation of a function is unknown, we proposed to apply corresponding approximation function.

Corresponding approximation function is constructed after linear and polynomial interpolation. Obtained approximation function is embedded in power consumption estimation script. Figure 3 represents block-scheme of proposed methodology [10].

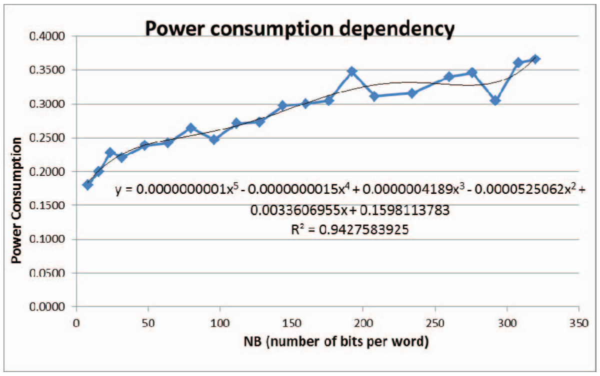

Figure 2. BIST power consumption dependency from NB (number of bits per word) parameter

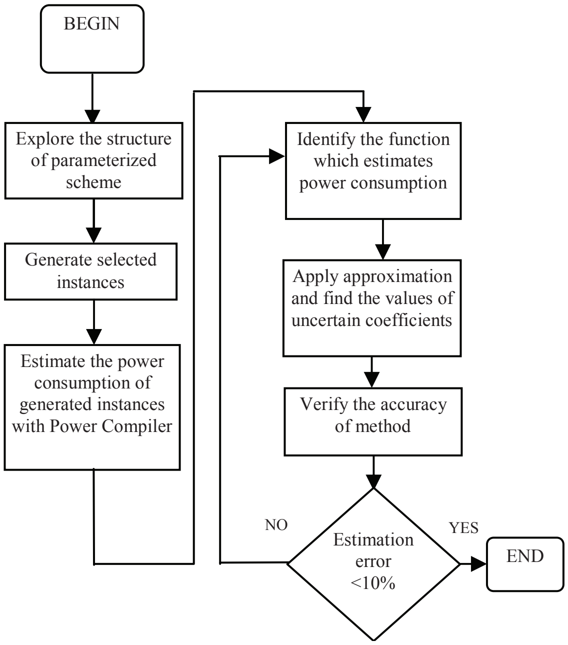
BEGIN
Explore the structure of parameterized scheme
Generate selected instances
Estimate the power consumption of generated instances with Power Compiler
Identify the function which estimates power consumption
Apply approximation and find the values of uncertain coefficients
Verify the accuracy of method
Estimation error <10%
NO
YES
END

Figure 3. Block scheme of proposed method

4. Experimental results

Below we present experimental results for memory BIST grouping methodology taking into account the given peak power, optimal test time and power domains (based on UPF).

Experimental results for peak power: Power consumption estimation and test time data for BIST architecture (Figure 1) is represented in Table 1. From the table it can be observed that only BIST1 and BIST2 can be run in parallel in order not to exceed 0.0003mv peak power limit.

Table 1. Given peak power for parallel mode is 0.0003mv

BIST Name	Power Consumption	Test Time
BIST1	0.0001mv	19.6ms
BIST2	0.0002mv	25.7ms
BIST3	0.0003mv	28.9ms

Experimental results taking into account optimal test time: According to equation (3) the test time for sequential BIST mode will be 74.2ms while for parallel BIST mode it is 28.9ms.

Experimental results taking into account power domains (based on UPF): The power management module controls power-down and power-up sequences. Memory BIST is divided into power domains. BISTs can be switched on and off in order to reduce their current consumption.

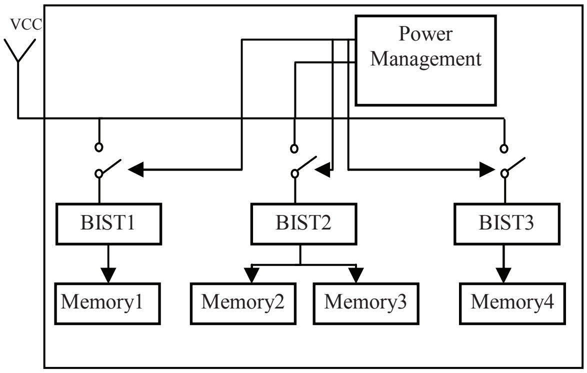

Figure 4. Memory BIST architecture based on UPF

Memory BIST grouping methodology takes information about power domains from UPF script (see Figure 4).

5. Conclusion

The paper presents memory BIST grouping methodology which is based on grouping memories taking into account the given peak power, power domains (based on UPF) and optimal test time. This methodology enables to perform power aware design at early stages of SoC design.

6. References

[1] L. Zaourar, Y. Kieffer, A. Wenzel, “A Complete Methodology for Determining Memory BIST Optimization under Wrappers Sharing Constraints”, 3rd Asia Symposium on Quality Electronic Design, 2011, pp. 46-54.

[2] M. Miyazaki, T. Yoneda, H. Fujiwara, “A Memory Grouping Method for Sharing Memory BIST logic”, Asia and South Pacific Conference, 2006, pp. 671-676.

[3] Ch. Trummer, Ch. Kirchsteiger, Ch. Steger, W. Reinhold, D. Dalton, M. Pistauer, “Simulation-based Verification of Power Aware System-on-Chip Designs UPF IEEE 1801”, Norchip, 2009.

[4] O. Mbarek, A. Pegatoquet, M. Auguin, “Using unified power format standard concepts for power-aware design and verification of systems-on-chip at transaction level”, IET Circuits Devices System, 2012, pp. 287-296.

[5] Xu Chuan-pei, Hu Hong-bo, Niu Jun-hao, “Test Scheduling of SOC with Power Constraint Based on Particle Swarm Optimization Algorithm”, Third International Conference on Genetic and Evolutionary Computing, 2009, pp. 611-614.

[6] V. Iyengar, K. Chakrabarty, E. Marinissen, “On Using Rectangle Packing for SOC Wrapper/TAM Co-Optimization”, VLSI Test Symposium, 2002, pp. 253-258.

[7] F.G. Zadegan, U. Inglesson, G. Asani, G. Carlsson, E. Larsson, “Test Scheduling in an IEEE P1687 Environment with Resource and Power Constraints”, Asian Test Simposium, 2011, pp. 525-531.

[8] Zou Wei, M. Reddy Sudhakar, Pomeranz Irith, Huang Yu, “SOC Test Scheduling Using Simulated Annealing”, 21 st VLSI Test Symposium, 2003.

[9] Power Compiler™ User Guid, SYNOPSYS-Version F-2011.09-SP4, March, 2012.

[10] L. Martirosyan, “A Quick Power Consumption Estimation Method for RTL Compilers”, Computer Science and Information Technologies (CSIT), 2013.

[11] Unified Power Format, IEEE Draft Standard for Design and Verification of Low Power Integrated Circuits, IEEE P1801 Std 1801™, 2013.

[12] F. Bembaron, S. Kakkar, R. Mukherjee, A. Srivastava, “Low Power Verification Methodology Using UPF”, pp. 228-233.