ABSTRACT

Test-data volume and test execution times are both costly commodities. To reduce the cost of test, previous studies have used test-data compression techniques on system-level to reduce the test-data volume or employed test architecture design for module-based SoCs to enable test schedules with low test execution time. Research on combining the two approaches is lacking. Therefore, this chapter studies how core-level test-data compression can be combined with test architecture design to reduce test cost further. The study is conducted in three steps. The first step analyzes how the TAM width influences three test-data compression techniques, namely Selective Encoding, Vector Repeat and the combination of the two. The second step investigates in what order to consider test architecture and test-data compression in the SOC design process to best reduce test cost. It is observed that test architecture design and test-data compression-technique selection should be performed in an integrated process. The third step presents a novel approach to integrate test-data compression-technique selection in the test architecture design process. Experiments on benchmarks with realistic cores show that the integrated approach achieves up to 32% reduction in test cost (7.8% on average) compared to non-integrated test architecture design and test-data compression technique selection.
1. INTRODUCTION

A trend in recent Integrated Circuit (IC) designs is to include billions of transistors in a single IC to provide higher performance and more functionality than in previous and smaller designs. Such high performance and multi-functionality ICs are extremely complex and time-consuming to design. To reduce the design complexity and meet short time-to-market requirements, there is a trend to employ a design approach where a system is composed of pre-designed and pre-verified blocks of logic called cores. The cores can be bought from core vendors and integrating them into an IC design is the task of a system integrator. Any core-based IC design that incorporates CPU, memory, etc., is often called System-on-Chip (SOC).

The IC fabrication process is not perfect and defects such as shorts and opens can occur and lead to faults. Therefore, each manufactured IC needs to be tested to determine if it is faulty. The general approach to test a circuit is to apply test stimuli at the inputs and compare the test responses at the outputs with the expected (fault-free) test responses. To generate the appropriate test stimuli and to apply them has become increasingly complicated due to the complexity of recent IC designs. Therefore, ICs and cores are designed with additional hardware to support testing, called design-for-testability (DFT). An important example of DFT is to insert scan-chains into sequential designs. Scan chains connect all flip-flops into shift-registers, which are activated during test and make it possible to shift-in (scan-in) and apply test stimuli inside a core. In the same way, scan-chains make it possible to capture test responses and shift-out (scan-out) these test responses for evaluation. This chapter assumes that all sequential core designs are equipped with scan-chains.

The cost for IC testing is increasing relative to the cost of manufacturing transistors and is becoming a significant part of the total IC manufacturing cost. The increase in IC testing cost is in part due to a huge test-data volume (number of bits), in terms of test stimuli and expected test responses. A design with almost two gigabits of test-data volume is mentioned in (Wang & Chakrabarty, 2005). The huge test-data volume requires large amounts of tester memory. Furthermore, in recent IC designs, the relation between the number of transistors and the number of I/O pins is one to a million. Because of this, there is a limited number of dedicated I/O pins for transporting test-data. This leads to a high test execution time since the high volume of test-data has to be transported through a limited number of I/O pins. In fact, the test execution time is another factor in the cost of testing ICs and therefore, it is important to minimize the test execution time.

This chapter aims to reduce the cost of testing SOCs with multiple cores. Some important concepts in the context of testing SOCs are discussed in the following, including modular testing with test-architecture design (Section 2.1), scheduling of core tests (Section 2.2) and test-data compression (Section 3). This chapter describes a study of how the cost of testing SOCs can be reduced by combining test-architecture design with test-data compression techniques. Previous studies have considered test-architecture design and test-data compression separately but studies on combining test-architecture design and test-data compression are lacking. A thorough analysis is presented in this chapter (Section 5), regarding the impact of the number of available wires for test-data transport on the test-data compression ratio of three compression techniques, namely Selective Encoding (Wang & Chakrabarty, 2005), Vector Repeat (Barnhart, Brunkhorst, Distler, Farnsworth, Keller, & Koenemann, 2001) and the combination of the two. Further experiments described in this chapter analyze the test cost for three strategies to combined test-architecture design and test-data compression. These three strategies are (1) test-data compression technique selection for each core followed by test-architecture design, (2)