Abstract

The test bench methodology helps the design engineer to structure the simulation of his circuit. As showed in this paper, the test bench methodology can further be developed in order to efficiently reuse simulation stimuli and response for the real device under test. As FPGAs are very often used to prototype an ASIC design, an easy switch between simulation and real hardware test is necessary to establish a rapid prototyping design and test environment. Our ProTest system closes the gap between the simulation and the test environment with a low cost and easy to use computer-aided-test environment.

1. Introduction

The technological evolution in microelectronics has lead to a permanent raise in complexity of VLSI systems integrated on high density chips. Since development and test time should not grow linearly with the complexity of VLSI chips new design and test methods are needed.

At the microelectronics laboratory of the Interdisciplinary Institute of Integrated Systems (MicroLab-I3S) a modern test bench design methodology is used to efficiently design VLSI circuits for research and industry projects. The following steps can be identified:

step 1: Technology independent design of digital systems using hardware description languages like VHDL or Verilog-HDL. Simulation of the digital system using the test bench methodology.

step 2: Synthesisation of the HDL description into a logical format for reconfigurable FPGA chips. Real time verification of FPGA chips with the already defined test bench using the low cost rapid prototyping test system ProTest.

step 3: Synthesisation of HDL description into target ASIC technology. Testing ASIC prototype chips with the same test bench procedure as used for simulation and for FPGAs on the ProTest environment.

If the objective is to design an FPGA, then the first two steps have to be performed. If the final goal is to get an ASIC or a masked programmable FPGA, step 3 has to be added. Synthesizing an FPGA in the ASIC design flow, described in intermediate step 2, has several advantages. This method allows to check for success of the design-for-testability at a much earlier stage. Also specification errors are discovered before producing the ASIC. This does lead to a much improved success rate for a correct design on the first attempt. Thus the presented design methodology is a consequent implementation of the state-of-the art design-for-test approach. The key element of this design methodology is the ProTest system with the universal test adapter for different devices under test (DUT) like our FPGAs or ASICs.

2. Test Bench Procedure

Fig. 1 shows a classical test bench used to simulate and verify a circuit model described with a hardware description language or with a schematic. The predefined ProTest monitor captures test patterns and simulation responses in a file during regular simulation. The ProTest monitor is activated by a VHDL or Verilog-HDL library call.

Once the digital system is designed and verified, synthesisation into an FPGA or an ASIC target technology can be performed. In order to guarantee a rapid prototyping, reuse of the simulation test bench is mandatory. The ProTest environment uses the same test bench as already developed in the HDL simulation and verification phase. Test patterns and simulation results captured by the ProTest monitor during simulation are used
to stimulate the real device-under-test. Fig. 2 illustrates the close relationship between the CAT-ProTest software and a test-machine.

Design engineers are used to work with CAD environments. Thus design and test procedures are faster if no additional tools have to be introduced to test the DUT. We reuse the test bench developed for circuit simulation for the real tests and result comparisons. Therefore the circuit model is simulated once again and the simulation results are immediately compared with the DUT's response in the CAD environment. In order to perform the comparison of the simulation data with the real chip test data, a simple switch of the ProTest monitor has to be changed from simulation to comparison. The switch causes the ProTest monitor to load the test result file into the users CAD environment in order to perform the comparison between simulation and DUT response obtained with a test machine. Fig. 3 illustrates the test bench used to read back the test results into the CAD environment.

3. ProTest System Description

The Computer-Aided-Test software CAT-ProTest acts as an interface between the CAD environment and the test machine. Thanks to the ProTest monitor and converter principle described above, all CAD-tools using Verilog-HDL or VHDL languages are supported. Currently drivers for the following two test machines are developed: the low-cost ProTest test machine and the HP16500 pattern generator and logic analyzer. The CAT-ProTest software is implemented as server client configuration, which allows to support multiple clients and test-machines simultaneously. Since the CAT-ProTest software is written in the platform independent Java language, test sessions can even be controlled via internet access.

The low-cost ProTest hardware test machine is able to generate test vectors and capture DUT responses with a resolution of 50ns. Up to six different clock signals per test cycle can be generated with the ProTest hardware. A library of universal test adapters allows the test of ASICs and FPGAs with standard device sockets. Using the HP16500 pattern generator and logic analyzer an improved resolution of 4ns is achievable.

4. Results and Conclusions

Surveys of HDL users have indicated that the generation of HDL test benches typically consumes 35% of the entire front-end ASIC design cycle. It is clear that the reuse of the test bench for the test of ASICs and FPGAs would significantly reduce the costs of HDL based designs. The presented ProTest system achieves the primary goal to merge design, rapid prototyping and test of ASICs and FPGAs. Due to the client server based principle of the CAT-ProTest system, student class exercises on design-for-test can easily be executed. Using an application specific test adapter PCB, one or even several integrated circuits can be grouped to form a DUT. In such a configuration the target integrated circuit can be tested in its application environment similar to a bread-board design.