

**TESTING OF HARDWARE BY USING A
HARDWARE SYSTEM ENVIRONMENT
THAT MIMICS A VIRTUAL SYSTEM
ENVIRONMENT**

FIELD OF THE INVENTION

The present invention relates to the field of integrated circuit design, simulation and testing. More particularly, the present invention relates to an improved technique for testing a hardware implementation of an integrated circuit design using test algorithms developed for verifying a software simulation of the integrated circuit design.

BACKGROUND OF THE INVENTION

Designing an integrated circuit typically involves creating a software model for the integrated circuit. The software model is a software algorithm that emulates the behavior of a design for the integrated circuit. To analyze and verify the design, diagnostic tests can be performed on the software model and, based upon the results, the design can be modified. Using a software model allows the design for the integrated circuit to be analyzed and verified before the integrated circuit is actually implemented in hardware. This is especially advantageous when the integrated circuit has a high density of logic elements and a broad range of functions which can make the integrated circuit relatively expensive to implement in hardware.

After the design for the integrated circuit has been verified using the software model, the design is implemented in hardware. Preferably, the hardware implementation is also tested to ensure that it performs as expected. Due to differences between the software model and the hardware implementation, such as in speed of processing capability, and differences in how each interacts with its respective environment, diagnostic tests utilized for testing the software implementation are generally not applicable for testing the hardware implementation, model and for testing the hardware implementation. This approach is disadvantageous due to the time and expense required to formulate these separate tests. In addition, because separate diagnostic tests are utilized, their results can be difficult to correlate, thus, making the debugging process difficult and time consuming.

Court et al., in U.S. Pat. No. 5,483,673, entitled: "Automatic Interface For CPU Real Machine and Logic Simulator Diagnostics" suggest the desirability of re-using diagnostics that exist for testing hardware of a prior generation of a system design for testing a software simulation of a next generation of the design. Court et al. recognize, however, that doing so is not practical because the software simulation runs on the order of a million times slower than the actual hardware. Therefore, Court et al. teach a method to speed up the execution of diagnostics in software simulations. The method involves identifying those functions which should be run on the real-machine and those that need to be run on the software simulator; executing those functions that must be run in the software simulator in the software simulator, while executing at least some of those functions that may be run on the real-machine on the real-machine; and coordinating the functions executed between the real-machine and the simulator to provide a diagnostic check-out of the system.

The approach taken by Court et al. has a drawback in that it assumes, as a starting point, that diagnostics for testing the hardware implementation already exist as having been developed for a previous generation of the system design and also assumes that the prior hardware also exists.

Depending upon the differences between the prior design and the new design, however, modifications to the prior diagnostics could be required and, when a completely new system is designed, no such diagnostics will exist. Further, the method requires coordinating results obtained from testing portions of the simulation with results obtained from testing portions of the hardware, which could increase the difficulty of determining the source of faults.

Therefore, what is needed is an improved technique for testing a software model of a design for an integrated circuit and for testing a hardware implementation of the design that does not have the disadvantages associated with the prior art. In particular, what is needed is a technique for testing a software model of a design for an integrated circuit that does not require separate diagnostic tests for the software model and for the hardware implementation and that does not depend upon the existence of a prior generation of the design.

SUMMARY OF THE INVENTION

The invention is a technique for testing a hardware implementation of an integrated circuit design using test algorithms developed for testing a software model of the integrated circuit design. A software model is created for a design of the integrated circuit. The software model is a software algorithm that emulates behavior of the integrated circuit under design. In addition, a virtual environment for the software model is also constructed. The virtual environment is a software algorithm that emulates an actual environment anticipated for the integrated circuit. A series of diagnostic tests are performed on the software model while it is operating in the virtual environment. These tests are used to verify and analyze the design for the integrated circuit. Based upon the results of the diagnostic tests, the design can be modified as necessary.

Once the design for the integrated circuit has been verified by testing the software model, an actual hardware circuit is constructed that implements the software model. In addition, an actual hardware environment is constructed that implements the virtual environment. Accordingly, the software model and the virtual environment are transformed into equivalent hardware circuits. Because of the equivalency between the software model and the hardware circuit and between the virtual environment with the hardware environment, the same diagnostic tests that were used to verify and analyze the software model in the virtual environment can be used to verify and analyze the hardware circuit in the hardware environment. This results in good test coverage for the integrated circuit design and reduced expense for developing the diagnostic tests.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 illustrates a block schematic diagram of the present invention.

FIG. 2 contrasts test code utilized for verifying a software model with test code utilized for verifying a hardware circuit.

FIG. 3 illustrates a block schematic diagram of an exemplary hardware environment utilized to verify a design for an integrated circuit processor TR3100.

DETAILED DESCRIPTION

FIG. 1 illustrates a block schematic diagram of the present invention. A software model 100 is illustrated in a virtual environment 200. The software model 100 is a software