Using Mission Logic for Embedded Testing

Rainer Dorsch Hans-Joachim Wunderlich
Computer Architecture Lab, University of Stuttgart, D-70565 Stuttgart, Germany
{rainer.dorsch,wu}@informatik.uni-stuttgart.de

Abstract

Testing logic cores of a system-on-a-chip causes a high test data volume which has to be stored on the external automatic test equipement (ATE), a high bandwidth requirement between ATE and the chip under test implying the need for high-speed ATE. This paper reduces these requirements by reusing embedded cores during test mode as embedded testers. Hard, firm, and soft cores may be reused, since only the functionality of the core in system mode is used.

The ITRS roadmap [3] predicts an increasing need for embedded testing since external testers become prohibitive expensive and the resolution of external testers relative to the chip frequency decreases. This may be avoided using embedded testers which move part of the test functionality from the ATE to the chip under test. To limit the costs for embedded testers (contained in each chip) the overhead for embedded test hardware should be minimized.

The presented methodology uses **MI**ssion **L**ogic for **E**mbedded **T**esting (MILET). Fig. 1 illustrates the test architecture of MILET. Instead of using a medium sized test access mechanism (TAM) from the ATE directly to the core under test (CPU in this example) [1], we suggest to transport *compressed test data* on a *narrow TAM* from the source to an *embedded tester* core (MPEG core in the example). The embedded tester core implements system functionality, part of it is reused for pattern decompression in test mode. At the same time, the core under test (CUT) is stimulated by the *decompressed test data* propagated on a *wide TAM* from an embedded tester core. The wide TAM propagates test responses to a compactor, which serves as pattern sink. This may be a small LFSR or another core implementing system functionality (DSP in the example) [2].

To be tested using MILET each core must come with a set of test patterns (hard cores) or it must be generated (soft cores) by the core integrator. We assume in this paper that the core under test is purely combinational or a pipelined structure, so the patterns may be reordered arbitrarily, and additional patterns may be generated and applied to the core under test.

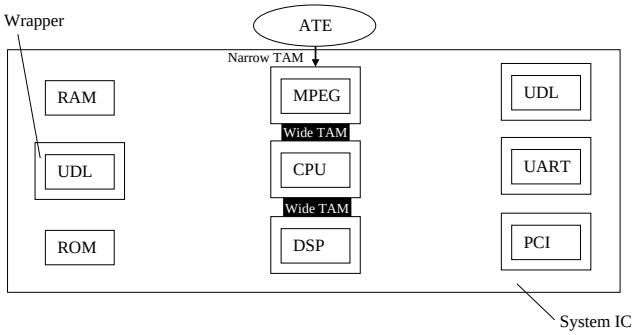

Figure 1. Test architecture of MILET.

MILET is able to reduce the test data volume compared to a compact test set, because deterministic test patterns contain unspecified bits, deterministic test patterns are correlated and deterministic test patterns may be reordered.

The algorithms used to encode the test data are based on symbolic traversal of finite state machines. Thus MILET benefits from the ongoing work in formal verification, but avoids the complexity problems since it solves an optimization and not a decision problem.

The experimental results show that the volume of the encoded test set is between 1% and 30% of the original test sets for the ISCAS85 and the combinational part of the ISCAS89 benchmarks. The compression quality is similar for all examined structures and compares well to dedicated hardware decompressors.

References

[1] E. J. Marinissen, Y. Zorian, R. Kapur, T. Taylor, and L. Whetsel. Towards a Standard for Embedded Core Test: An Example. In *Proceedings of the IEEE International Test Conference (ITC)*, pages 616–627. IEEE, 1999.

[2] J. Rajski and J. Tyszer. *Arithmetic Built-In Self Test For Embedded Systems*. Prentice-Hall, 1998.

[3] Semiconductor Industry Association. *The International Technology Roadmap for Semiconductors (ITRS)*. International SEMATECH, Austin, TX, 1999.