Simulator Independent Coverage for RTL Hardware Languages

Kevin Laeufer, Vignesh Iyer, David Biancolin, Jonathan Bachrach, Borivoje Nikolić, Koushik Sen

4University of California, Berkeley, 5SiFive, 6JITX

Coverage: Who tests the testers?

All tests pass. Are we done?

Not Tests are empty.

Problem 1: Most open-source simulators lack coverage
- Verilator only supports line and toggle coverage
- Treadle (Chisel interpreter), ESSENT and FireSim have no coverage support

Problem 2: Verilog coverage is an inadequate replacement for native Chisel coverage
- In this example, the translation to structural Verilog replaces a branch with a conditional assignment. Therefore, 100% line coverage on the generated Verilog does not necessarily imply complete line coverage of the Chisel source.

Our Solution: Simulator Independent Coverage

We show how a single new cover statement allows us to move all coverage instrumentation into the FIRRTL compiler. Adding support for the cover statement to a simulator is fairly simple.

Line Coverage

Verilator: cover Statement Support

Verilator: Coverage Instrumentation Overhead

Coverage instrumentation overhead on Verilator v4.034. For TLRAM, the measured overhead of our FIRRTL line coverage is close to zero.

FireSim: cover Statement Support on FPGA

We generate saturating counters and a scan chain for all cover statements for FPGA-accelerated simulation with FireSim.

Finite State Machine Coverage

The cover statement

Clock Predicate Unique Name

More Features
- Applicable to other hardware languages.
- Feedback-Directed Fuzz Testing
- Coverage merging
- Formal cover trace generation

Paper PDF, Code & Slides

https://kevinlaeufer.com/papers/simulator_independent_coverage_asplos2023.html [laeufer, vighnesh.iyer, biancolin, bora, ksen]@eecs.berkeley.edu, jrb@pobox.com

ASPLOS 2023