The Role of Verification at the Architecture Definition Stage

Debashis R Chowdhury

DesignCon East – Panel Discussion

September 20, 2005
Conceptual System-to-RTL DV flow

- Algorithm design
- Data-flow
- System architecture
- TLM, PV/T
- HW verification
- RTL
- SW development
- PV

Architecture Verification
RTL Verification
Key Components of Verification

- **Stimulus generation**
  - Testbench technology
- **Check correctness** - golden results or self-checking
  - Assertions
- **Coverage**
  - Functional coverage

Observer / Scoreboard to compute Functional Coverage

Stimulus Generation ➔ Design Under Test ➔ Self-Checking with Assertions
Potential Trend

• Testbench development is one of the most time-consuming tasks
  ▪ Early start will reduce overall design time
• Reuse of verification between architecture and RTL phases will shorten design time and ensure consistency
• Reuse means -
  ▪ Unified testbench technology between architecture, RTL phases
  ▪ Unified assertions
  ▪ Notion of functional coverage at TLM level
Unified Verification from Architecture-to-RTL

Scenario Layer
Function Layer
Command Layer

Unified Testbench Methodology

Tests

Assertions

Simulate Architecture model of the SoC

Simulate RTL model of the SoC

Ensure Consistent Architecture and RTL Models
Unified Verification – Proposed Approach

• One language for Testbench at TLM and RTL
  ▪ needs to be standard
  ▪ needs to be optimal at RTL, where detailed test features and simulation effort will be spent
  ▪ SystemVerilog can be the choice!
• Must have Testbench reuse for both top-down and bottom-up design styles
  ▪ Alternative requires 2 designs (TLM and RTL) and 2 testbenches
  ▪ Hard to verify that two verification environments are equivalent (bad idea!)
• Testbench reuse requires methodology
Summary

• Increasing trend that Verification between architecture and RTL phases to be reused
• Extending RTL testbench, assertions, coverage technologies is essential to bridge the gap
• Methodology for reuse is key