

Universal Verification Methodology: Fundamentals and Migration

Move a UVM 1.2 testbench toward IEEE 1800.2-based libraries

FA-0709

Embedded Systems and Hardware

Introduction

Review reusable UVM verification components and plan a controlled migration from UVM 1.2 to an IEEE 1800.2-based library. The course combines compatibility analysis with a guided migration of a small sample testbench.

Distinguish the IEEE methodology standard from Accellera reference-library release numbers and simulator support. The lab uses an explicitly selected library and supported simulator; migration effort and runtime benefits must be assessed rather than assumed.

Learning outcomes

- Explain the relationship between UVM 1.2, IEEE 1800.2 and the UVM 2020 reference-library releases.
- Identify relevant deprecated APIs, compatibility mechanisms and documented implementation differences.
- Review sequence, configuration, factory and reporting behaviour in an existing testbench.
- Migrate and regress a sample environment while preserving its verification intent.
- Use release notes, simulator documentation and measurements to assess migration risks and performance.

Prerequisites

- Basic knowledge of Verilog/SystemVerilog for digital design and verification.
- Experience with UVM 1.2, including developing testbenches and verification components.
- Familiarity with object-oriented programming concepts.
- Understanding of digital electronics and system-on-chip (SoC) architecture.

Access to a supported SystemVerilog simulator with the selected UVM library and a small reproducible UVM 1.2 testbench.

Training outline

1. Module 1 — Standards and library versions

- Review the UVM component model and reusable verification architecture.
- Distinguish UVM 1.2, IEEE 1800.2-2020 and the Accellera UVM 2020 reference implementation; do not use an ambiguous standalone “UVM 2.0” label.
- Choose the target library and simulator combination using vendor support, release notes and known deviations.

2. Module 2 — Compatibility and behaviour review

- Compare the selected releases and identify removed, deprecated, compatibility-only or implementation-specific APIs.
- Review sequences and sequence items, configuration, factory overrides, reporting and utility classes in the sample environment.
- Separate testbench functional coverage from simulator-specific collection and analysis features; do not assume the new library adds a complete coverage tool.
- Inspect documented fixes and changes rather than attributing generic new mechanisms or guaranteed improvements to every release.

3. Module 3 — A controlled migration plan

- Capture the working baseline, compile options, seeds, regression expectations and dependencies.
- Update build configuration and relevant APIs incrementally; decide where compatibility support is appropriate and where code should be revised.
- Check third-party verification components and simulator behaviour; retain a rollback path and review representative migration examples.

4. Module 4 — Guided migration lab

- Migrate the supplied UVM 1.2 testbench to the selected IEEE 1800.2-based reference library.
- Resolve compile errors, deprecation warnings and changed runtime behaviour.
- Run comparable tests and inspect sequences, reports and coverage expectations to identify regressions.
- Measure selected runtime or memory characteristics under comparable conditions; do not infer a universal performance gain.

5. Module 5 — Documentation and maintenance

- Navigate the standard, class reference, release notes, compatibility documentation and simulator resources.
- Document the validated library/tool combination, exceptions, unresolved issues and regression evidence.
- Use the Accellera community and vendor support channels for reproducible verification problems.