Verdi

Verification for heterogeneous Reliable Design and Integration

System Level Verification/Validation for heterogeneous Systems

Thilo Vörtler

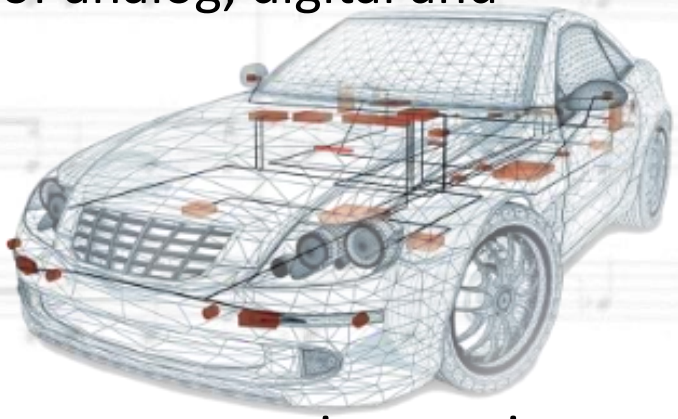
FDL 26.09.2013

Overview

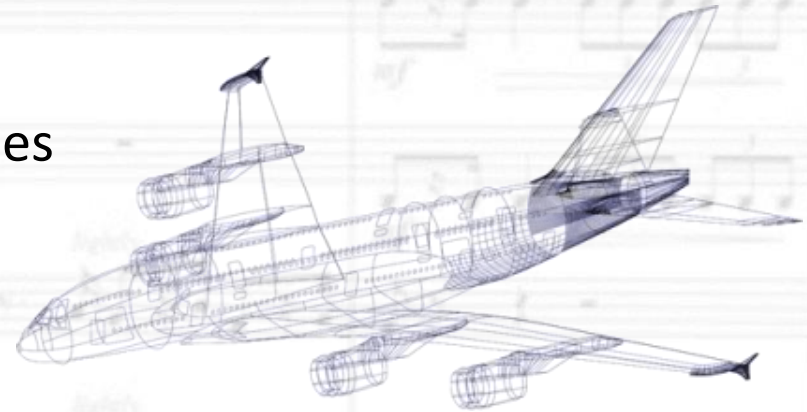
- Motivation
- State of the art in system verification
- Verdi Methodology Design Flow
- From Verification to Validation
- Conclusions

Motivation

- Heterogeneous systems are a combination of analog, digital and mechanical elements



- The SystemC/AMS language allows it to model such systems at a high level of abstraction → Used for Virtual Prototyping
- In heterogeneous systems time and values are seen continuous, in contrast to purely digital systems



- Different kind of verification approaches exist:
  - Simulation based
  - Formal Methods
  - Lab based measurements

Verdi

State of the Art in System Verification

- For purely digital Systems:
  - Component/Module level: Formal methods + constrained random tests
  - System level: Verification methodologies like UVM

    → Allow reusability of test components (Verification IP) and easy testbench setup

- For heterogeneous Systems:
  - Mostly directed testing available

    →formal methods only at research level (e.g. hybrid automata)

    →automatic evaluation of analog signals is hard
  - No standardized verification methodology available

Challenges in System Verification

- System Level or *High level tests* are derived from the requirements and/or specification → No reference model for comparison

- *High level tests* check the integration of the whole system
  →Timing of test sequences is very critical
  →Tests are also the foundation for lab based *Validation*

- Especially tests description for heterogeneous systems often contain timing uncertainties
  →e.g. “between 0 ms and 5 ms, the output should be set to 1 after…”

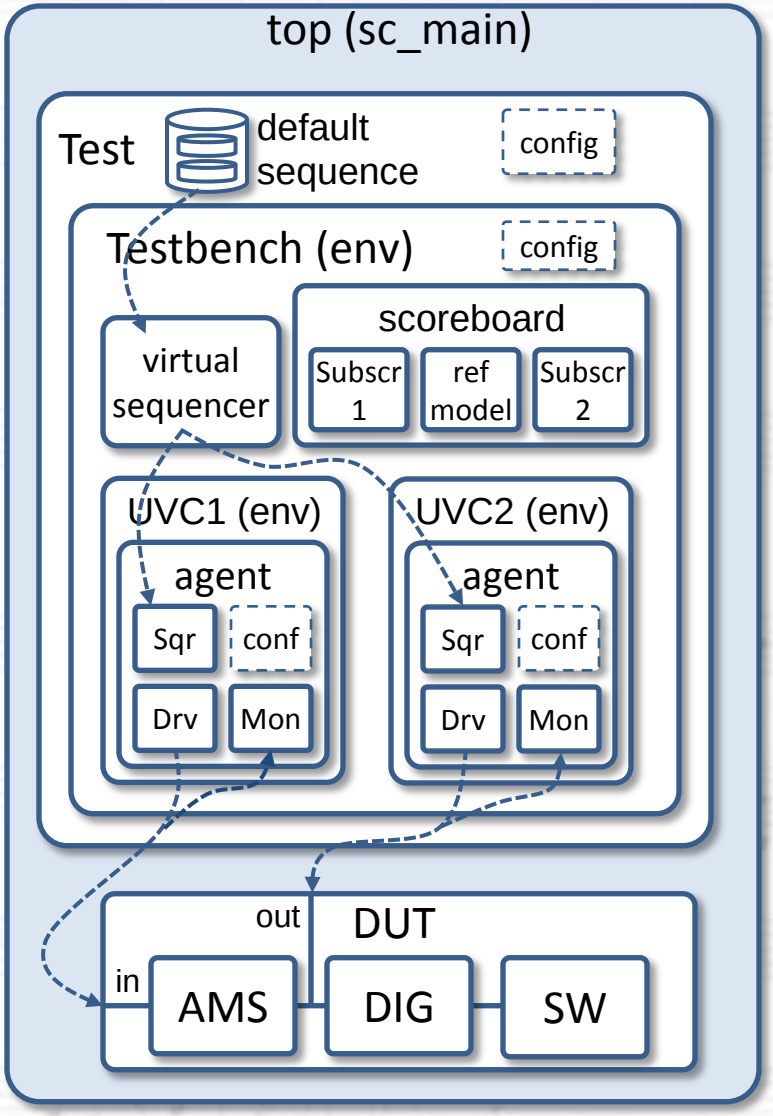
→Verdi project aims to develop a Verification Methodology based on UVM in a SystemC/AMS flavor for Verification and Validation

Verdi

UVM for SystemC

Universal Verification Methodology UVM

- Industry standard in digital verification
- Facilitates the development of reusable verification components
- Library implementation only available in SystemVerilog → Currently being ported to SystemC
- Relies heavily on use of SystemVerilog language features



UVM SystemC implementation challenges

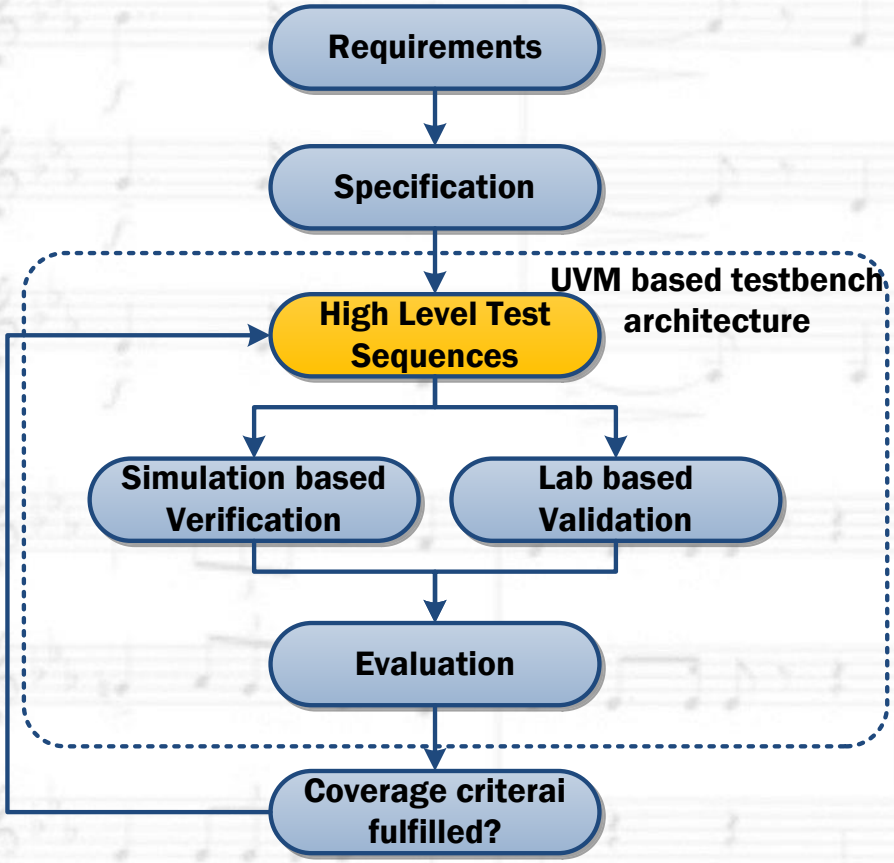
SystemC language doesn't provide all language features of SystemVerilog needed for efficient verification:

- Constraint randomization available using SCV, CRAVE Library
  → Miss support for distribution functions and analog values
- Coverage classes not available
- Assertion language needed, which supports continuous time and value domains

Description of high level tests:

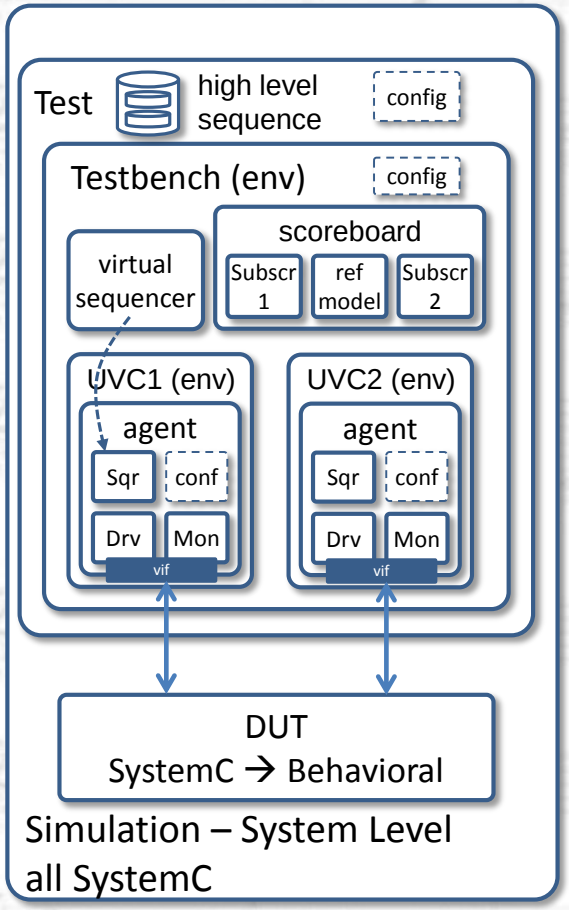
- UVM sequences are mostly untimed → timing handled mostly in the lower level
- No reference model exists to compare results with

Verdi Methodology Design Flow



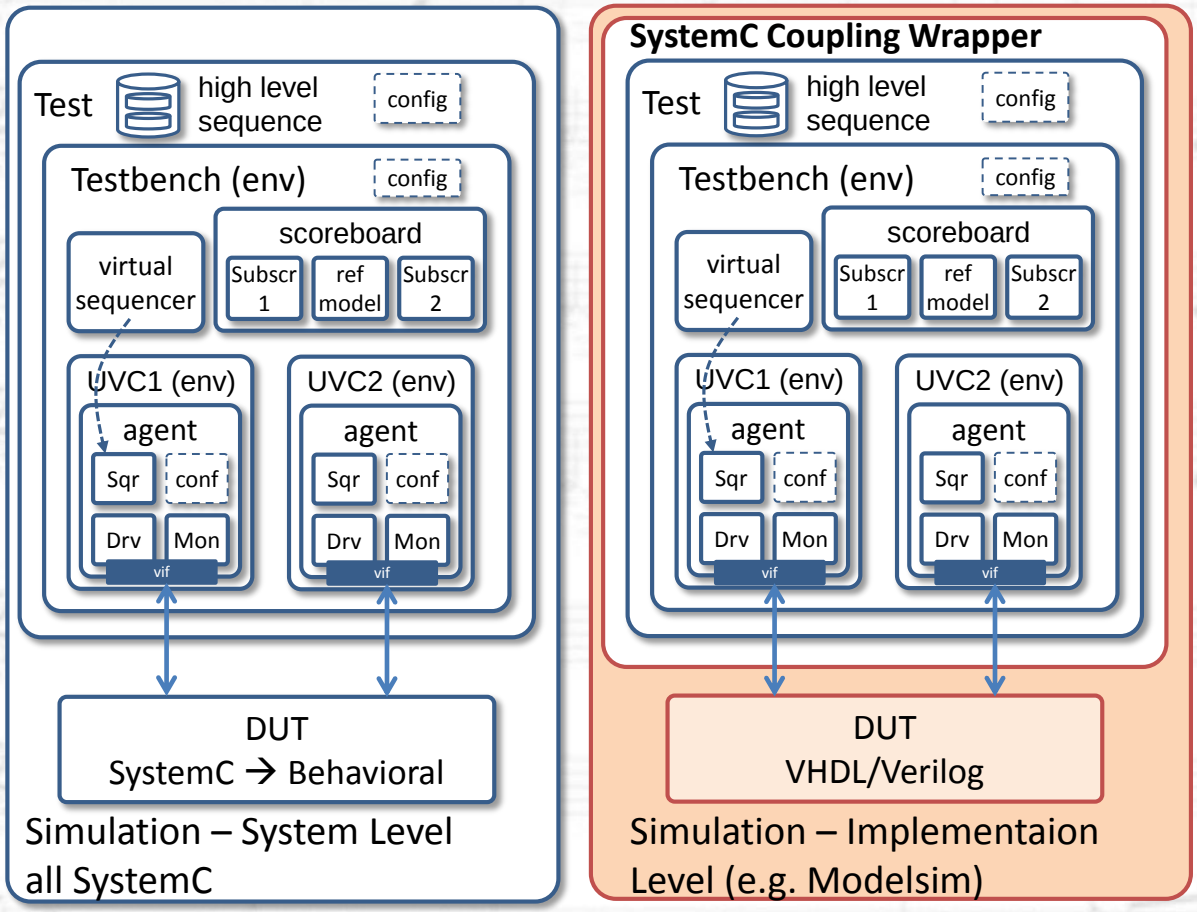
- High level tests shall be reused without modification between verification and validation
- UVM Testbench shall be generated from IP-XACT specification of UVM components
- Traceability of requirements to sequences shall be possible

From Verification to Validation



- SystemC (C++) based UVM allows easy adaption to different hardware platforms

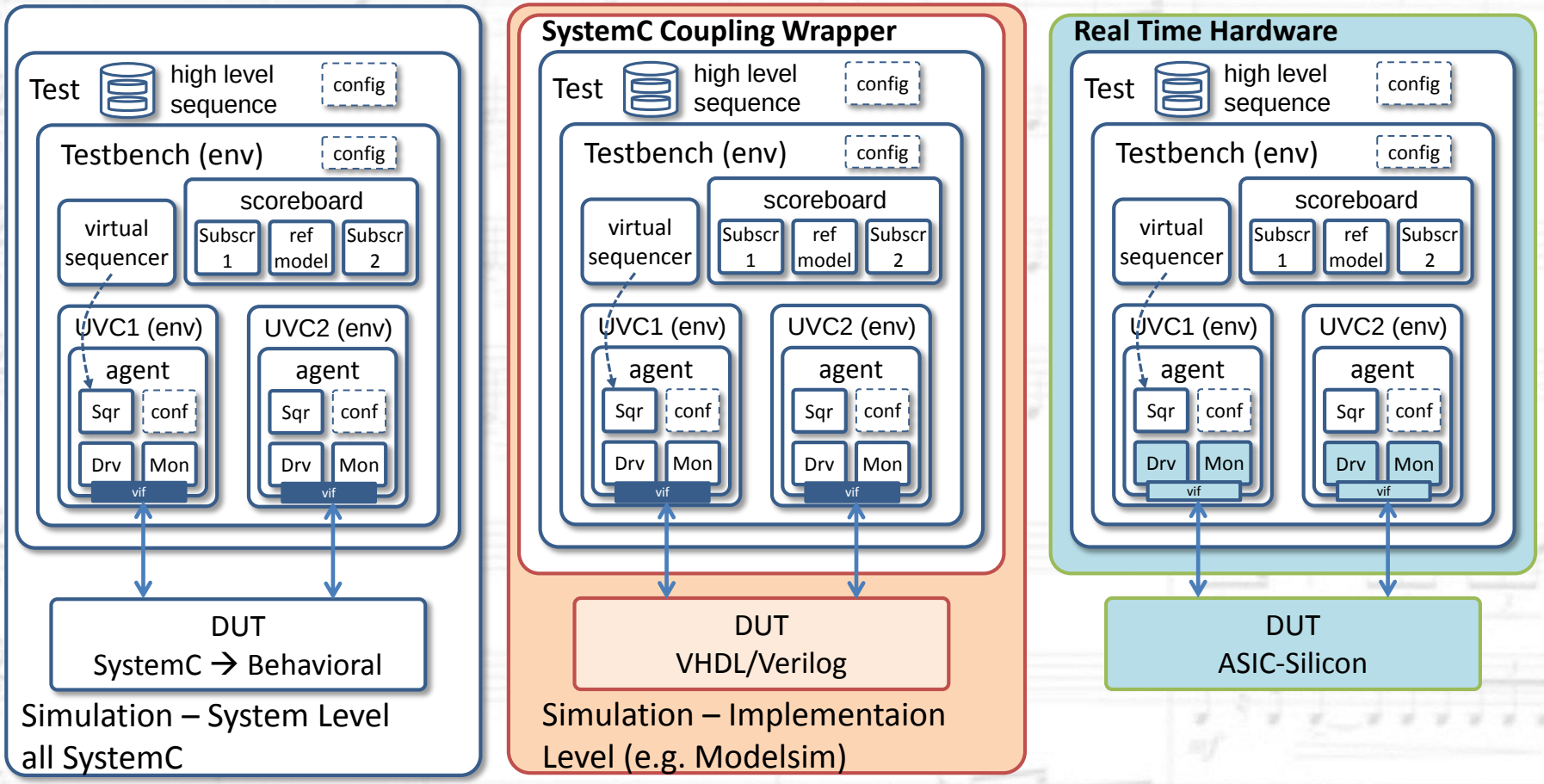
From Verification to Validation



- SystemC (C++) based UVM allows easy adaption to different hardware platforms

Verdi

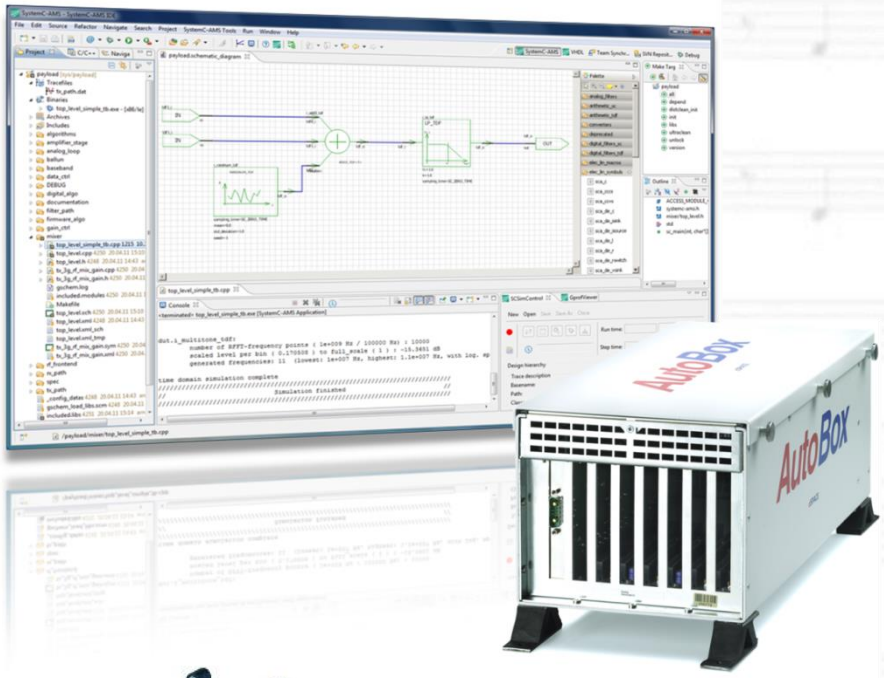
From Verification to Validation



- SystemC (C++) based UVM allows easy adaption to different hardware platforms

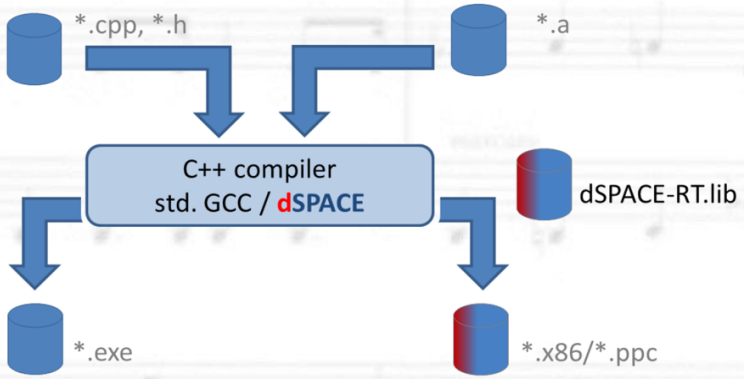
Verdi

Validation Using Real Time Hardware



SystemC UVM **testbench**

SystemC/AMS **simulator**



SystemC UVM **simulation testbench**

SystemC UVM **dSPACE real-time testbench**

DUT

SystemC based UVM testbench

Verdi

Conclusions

- The VERDI project develops a Verification and Validation methodology aimed at the system level using SystemC/AMS and UVM
  - Key features of the UVM SystemVerilog implementation are made available for SystemC
  - The Library will include randomization and functional coverage capabilities for AMS and system-level verification not currently available for SystemC/AMS
  - Proof-of-concept library will be submitted to Accellera Systems Initiative
- Sequences described for system level test shall also be used for lab validation
  - These sequences can also be run on real time hardware like dSpace

→ The same testcase can be executed in simulation and in hardware