
CLOSING THE GAP BETWEEN SYSTEMC/AMS SIMULATION AND LAB BASED VALIDATION

Thomas Arndt, Karsten Einwich, Thilo Vörtler

CATRENE DTC 2013 Dresden



OVERVIEW

- Introduction
- UVM for SystemC/AMS
- Challenges in System Level Testing
- Validation of hardware using SystemC/AMS
- Summary

Fraunhofer-Institut for Integrated Circuits IIS



- **Founded** 1985
- **Locations** Erlangen, Nuremberg, Fuerth, **Dresden (1992)**
- **Employees** 730
- **Budget** 90 Mio. €
 - 20% basic governmental funding
 - 80% project financed
- **Management** Prof. Albert Heuberger

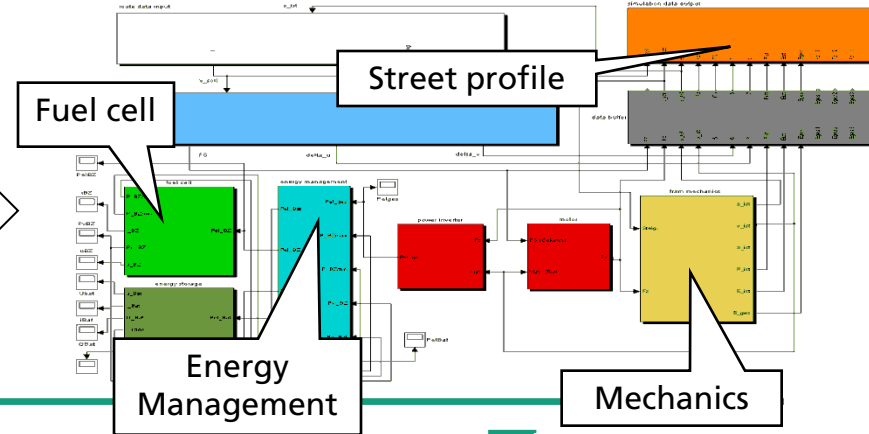
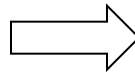
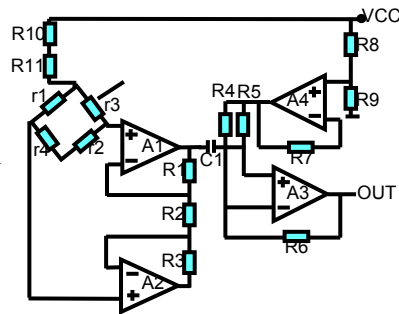
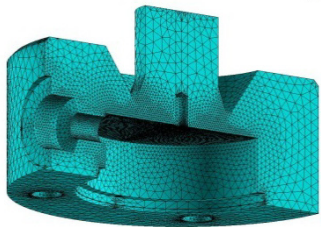
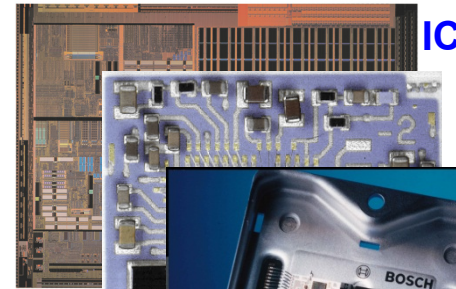
Design Automation Division EAS, Dresden

Founded in 1992

Approx. 90 staff

Budget: approx. € 7 million

- Development of methods and tools for computer-aided design of electronic circuits and systems for the complete value chain
- Main areas of work Modeling, simulation, synthesis, optimization, verification and testing



Introduction into SystemC

- SystemC is a C++ based system description language for higher abstraction levels → Used for virtual prototyping
- SystemC/AMS extends SystemC for abstract modeling of analog/mixed-systems
- Hosted and standardized by the Accellera Systems Initiative
- One of the big challenges is the verification of the designed system and its reference implementation
- EU funded FP7 VERDI works on developing a verification methodology for SystemC/AMS

UVM for SystemC/AMS

- The Verdi project has the goal to :
 - Define a unified system-level verification (simulation) and validation (lab) methodology
 - Specify a reuse strategy for verification IP between and within companies and for different product generations
 - Define a path from verification IP to validation IP, to bridge the gap between simulation based verification and laboratory prototype validation



Project partners:



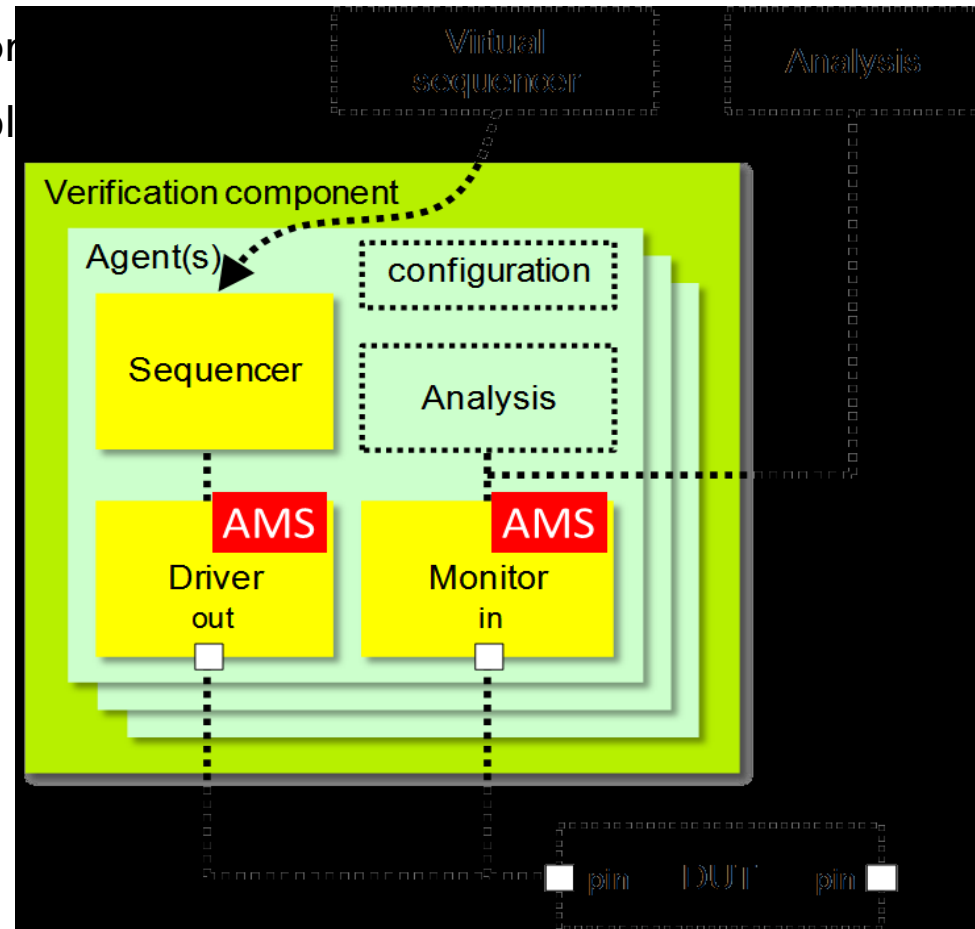
UVM for SystemC/AMS

■ Universal Verification Methodology UVM

- Industry standard in digital verification
- Facilitates the development of reusable verification components
- Implementation only available in SystemVerilog → **Currently being ported to SystemC**

■ Challenges for AMS implementation

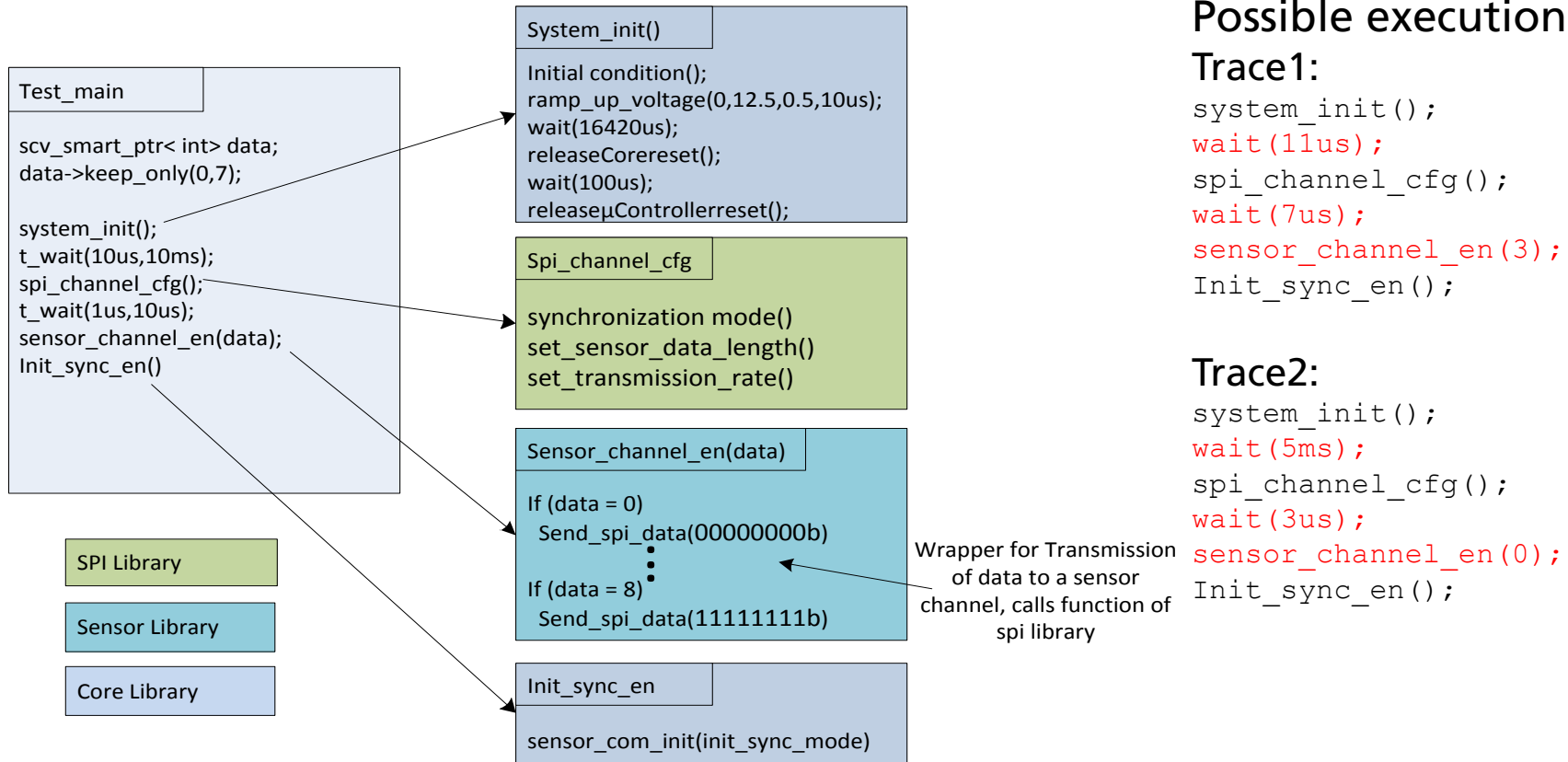
- Randomization & coverage of analog signals
- Assertions supporting analog events
- Description of system level test cases



Challenges in System Level Testing

- System level tests are derived from the requirements and/or specification
- Tests are the foundation for simulation and lab checks
 - Tests check the integration of the whole system
 - Different from block level tests for which UVM is best suited
 - Timing is of test sequences is very critical
- Especially in AMS, test often contain timing uncertainties
 - e.g. "between 0 ms and 5 ms, the output should be set to 1"
- UVM for AMS is extended to high level sequences, that can be run on UVM (virtual) sequencers

Challenges in System Level Testing



Possible execution traces

Trace1:

```

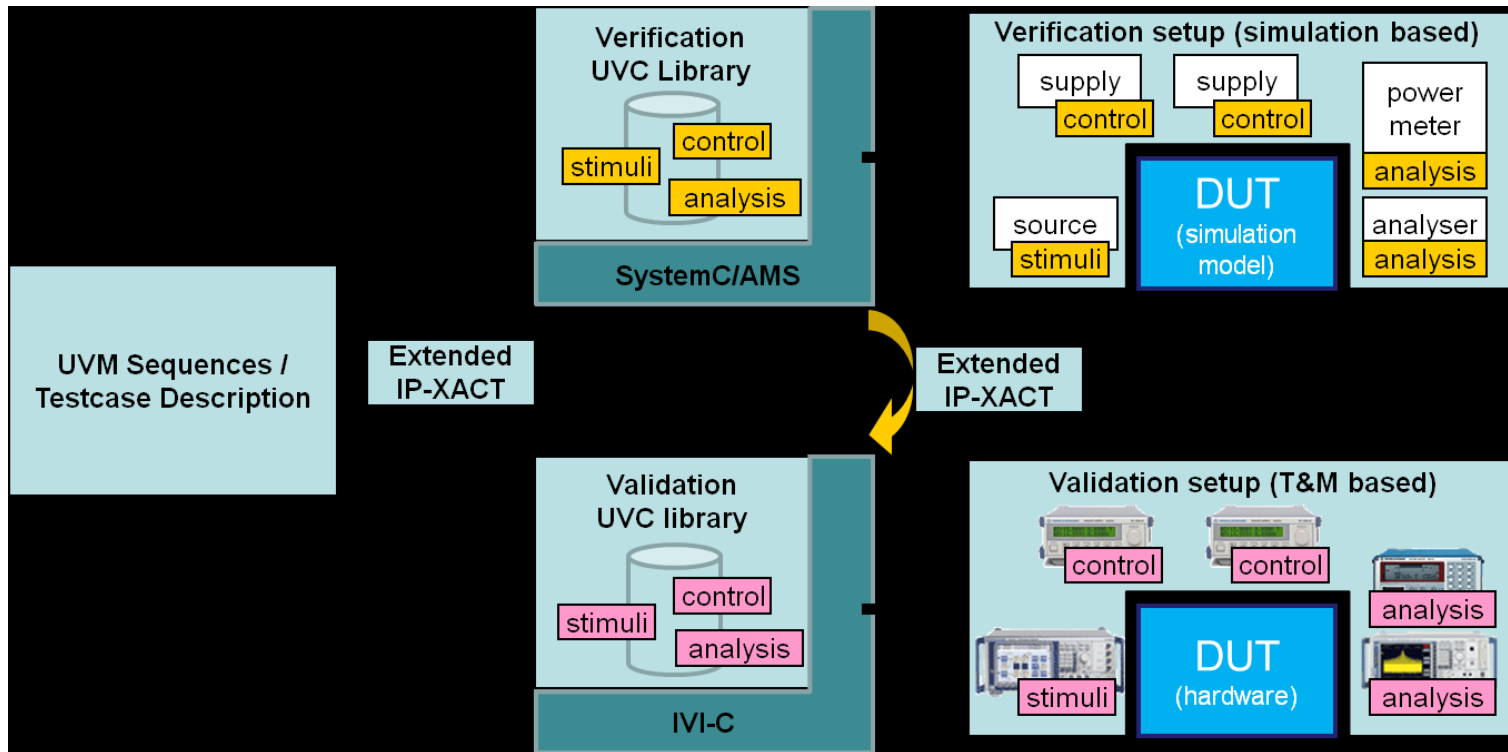
system_init();
wait(11us);
spi_channel_cfg();
wait(7us);
sensor_channel_en(3);
Init_sync_en();
  
```

Trace2:

```

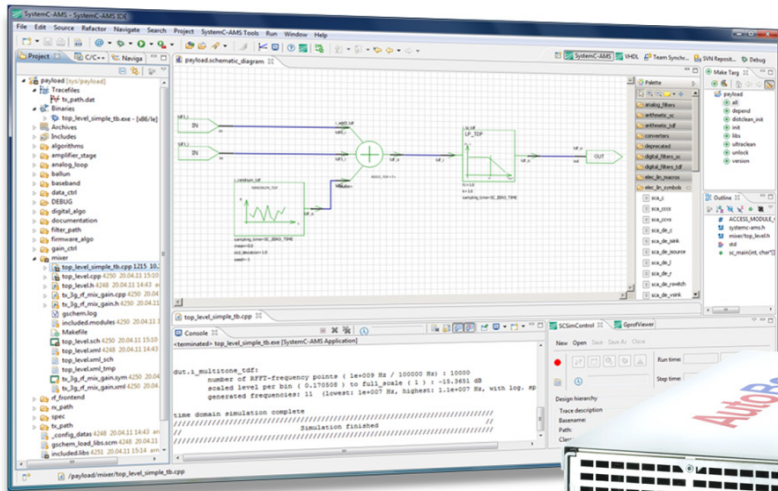
system_init();
wait(5ms);
spi_channel_cfg();
wait(3us);
sensor_channel_en(0);
Init_sync_en();
  
```

Validation of hardware using SystemC/AMS



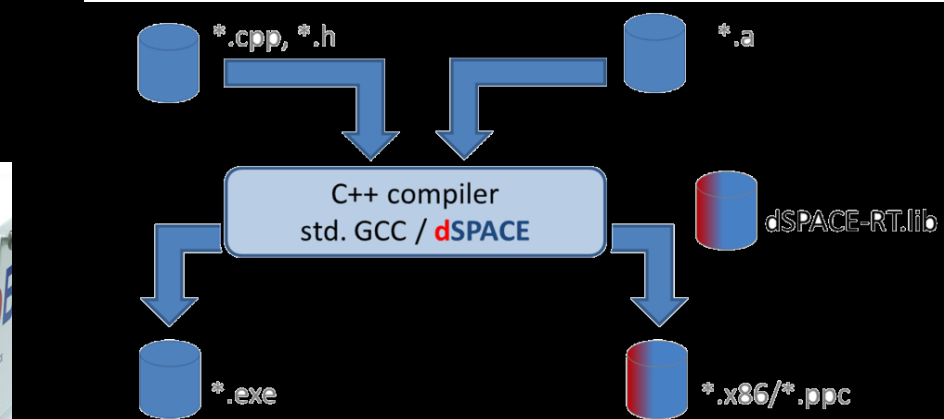
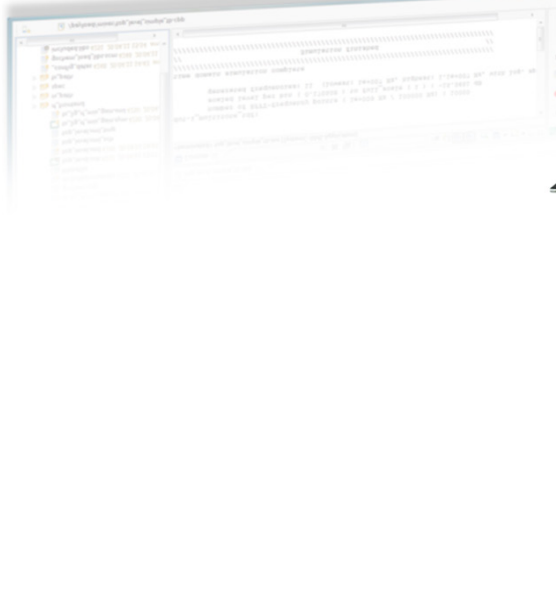
- SystemC UVM test scenarios shall be reused in the lab for validation
- Real time hardware used to run SystemC+UVM

Validation of hardware using SystemC/AMS



SystemC/AMS model

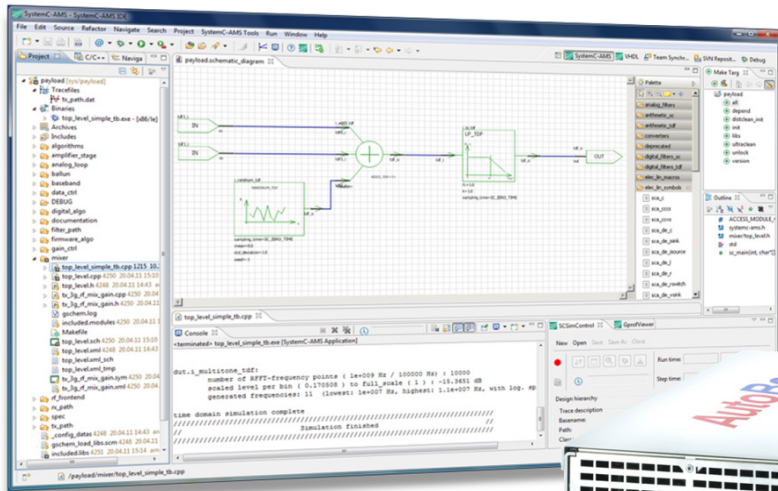
SystemC/AMS simulator



SystemC/AMS stand-alone executable virtual prototype

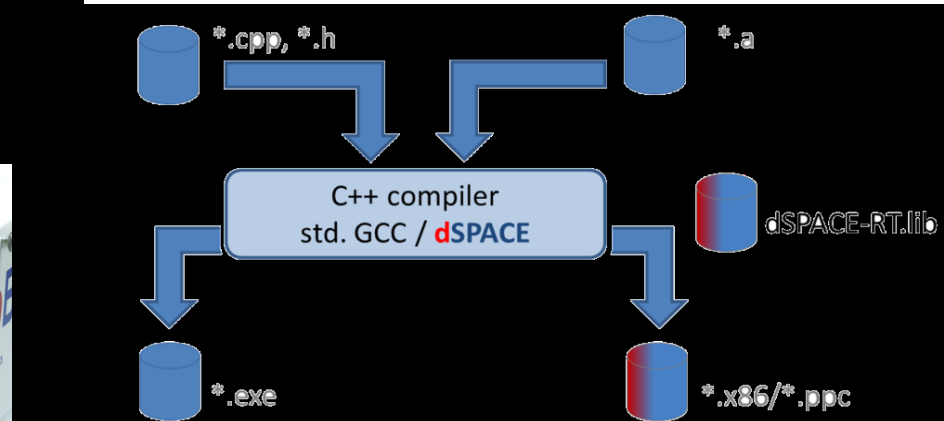
SystemC/AMS stand-alone dSPACE real-time execution

Validation of hardware using SystemC/AMS



SystemC UVM testbench

SystemC/AMS simulator



SystemC UVM simulation testbench

SystemC UVM dSPACE real-time testbench



<http://medcenter.com-online.com>

DUT



SystemC based UVM testbench

Summary

- The VERDI project currently develops a UVM library in SystemC/AMS
 - First focus is making available the features of the UVM SystemVerilog implementation for SystemC
 - The Library will include randomization and functional coverage capabilities for AMS and system-level verification not currently available for SystemC/AMS
 - Aim is to contribute the initial language definition and/or Proof-of-concept to Accellera Systems Initiative later this year
- Sequences described for system level test can 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