RTL Quality for TLM Models

Author: Preeti Sharma

Presenter: Preeti Sharma

Company/Organization: Synopsys

accellera SYSTEMS INITIATIVE™

2014 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION INDIA

Agenda

- Introduction
- TLM Verification challenges
- A solution: Verification IPs
  - UVM VIP: An overview
- What we propose
  - UVM based VIP re-usage
  - Integration with TLM models
  - Case Study
- Conclusion
- Acknowledgements/References

Introduction

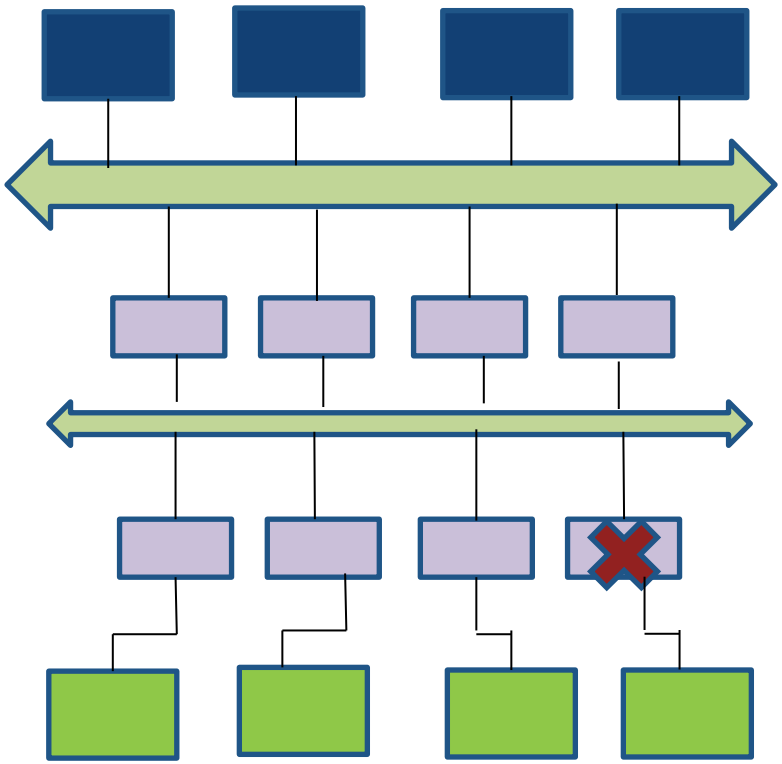
Faster simulation speeds

Architecture exploration

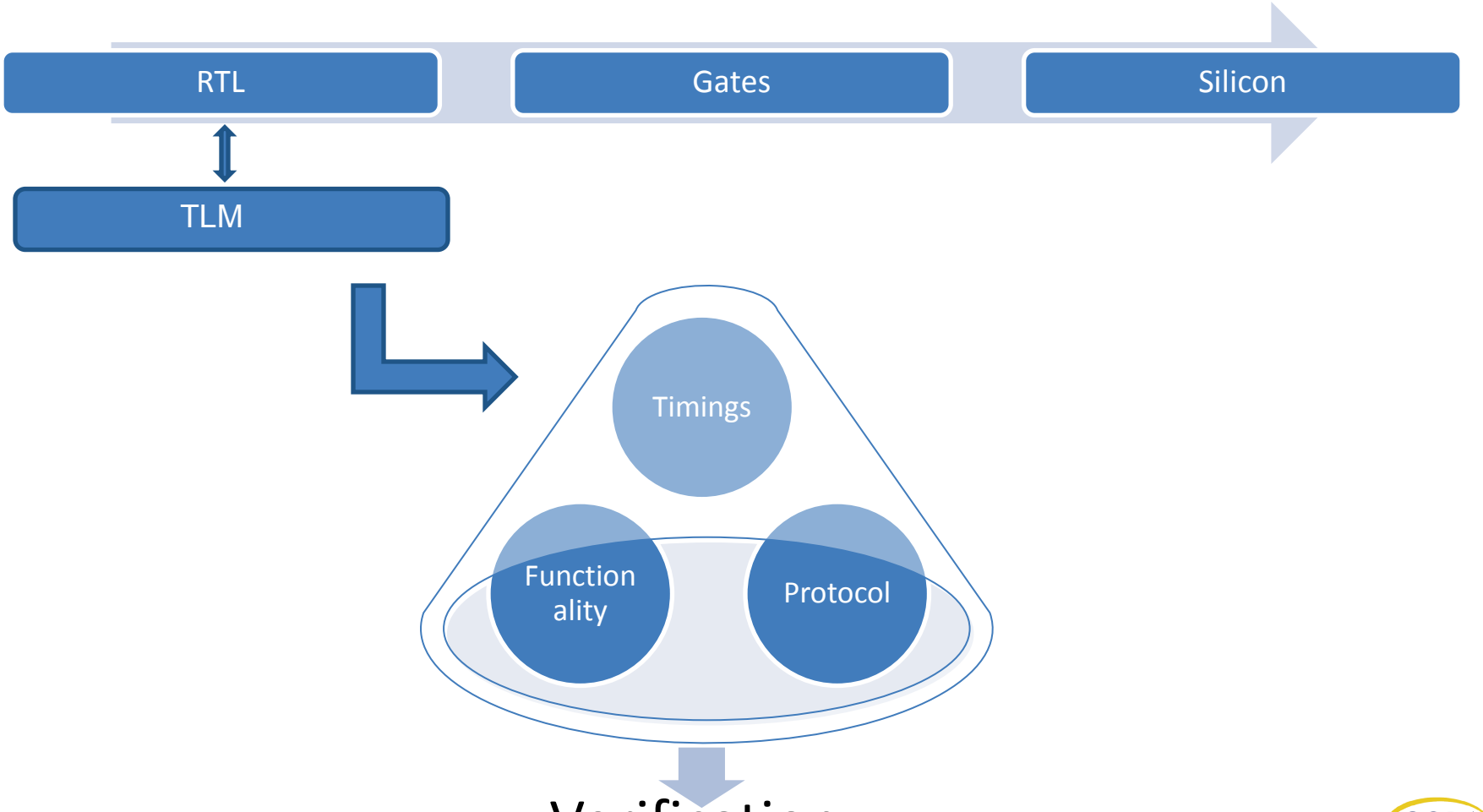
Early software development

TLM Models

Virtual Platform

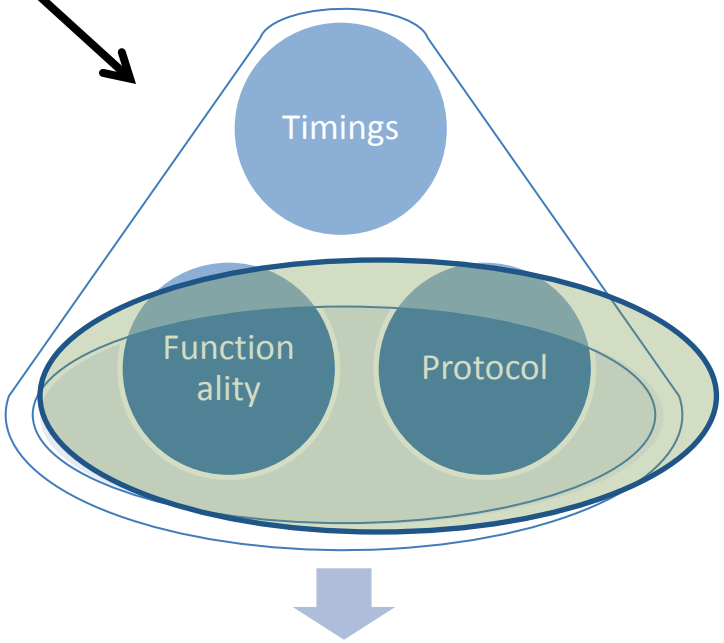


TLM Verification challenges



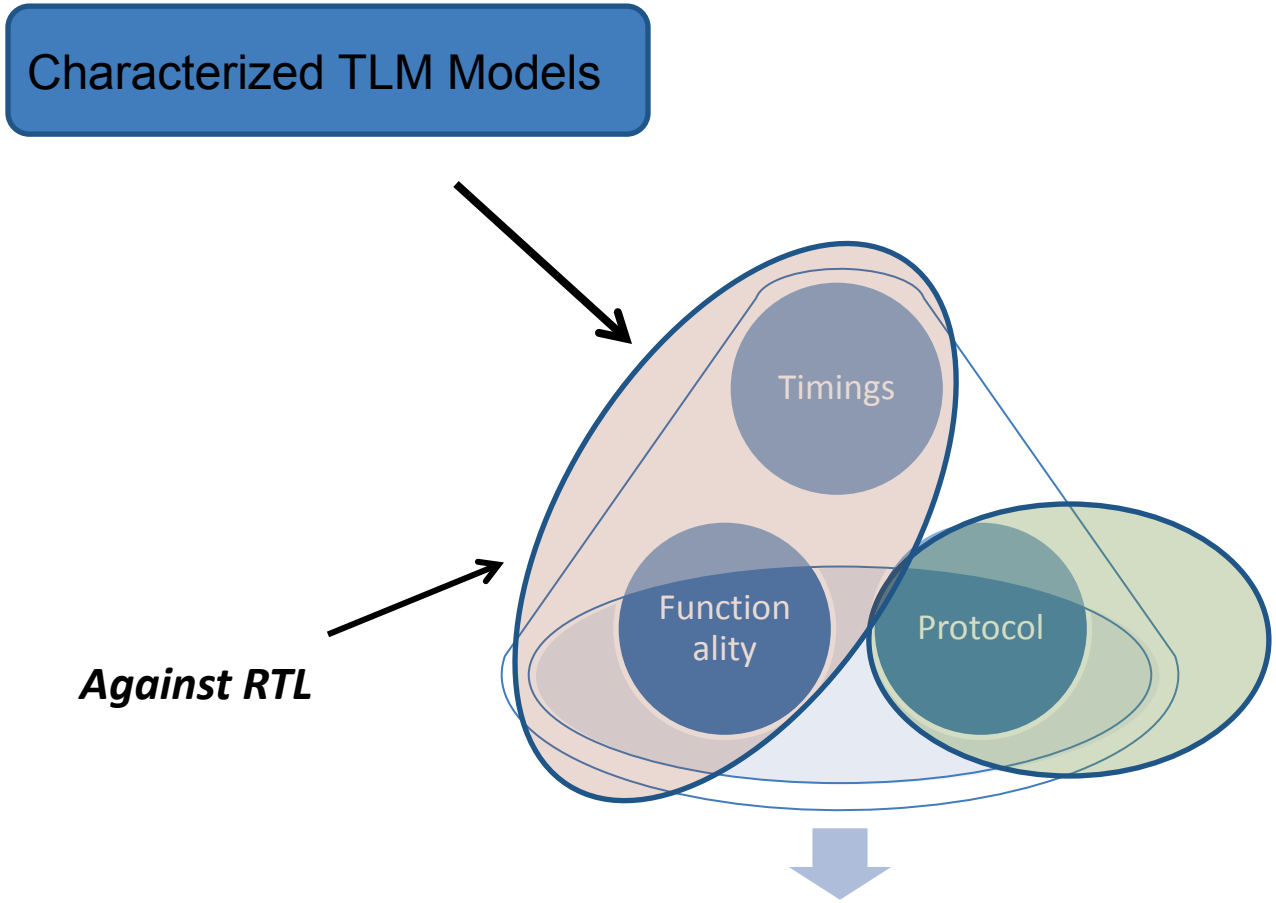
TLM Verification challenges

Generic TLM Models



Verification

TLM Verification challenges



TLM Verification challenges

- Large and complex models
- Generic versus Characterized TLM models
- Creation of robust verification test-bench environment
  - Robust
  - Modular and reusable
  - Comprehensive protocol compliance testing
- No established standard TLM-driven verification flow

TLM Verification challenges

- Large and complex models
- Generic versus Characterized TLM models
- Creation of robust verification test-bench environment
  - Robust
  - Modular and reusable
  - Comprehensive protocol compliance testing
- No established standard TLM-driven verification flow

TLM Verification challenges

- Large and complex models
- Generic versus Characterized TLM models
- Creation of robust verification test-bench environment
  - Robust
  - Modular and reusable
  - Comprehensive protocol compliance testing
- No established standard TLM-driven verification flow

TLM Verification challenges

- Large and complex models
- Generic versus Characterized TLM models
- Creation of robust verification test-bench environment
  - Robust
  - Modular and reusable
  - Comprehensive protocol compliance testing
- No established standard TLM-driven verification flow

TLM Verification challenges

- Large and complex models
- Generic versus Characterized TLM models
- Creation of robust verification test-bench environment
  - Robust
  - Modular and reusable
  - Comprehensive protocol compliance testing
- No established standard TLM-driven verification flow

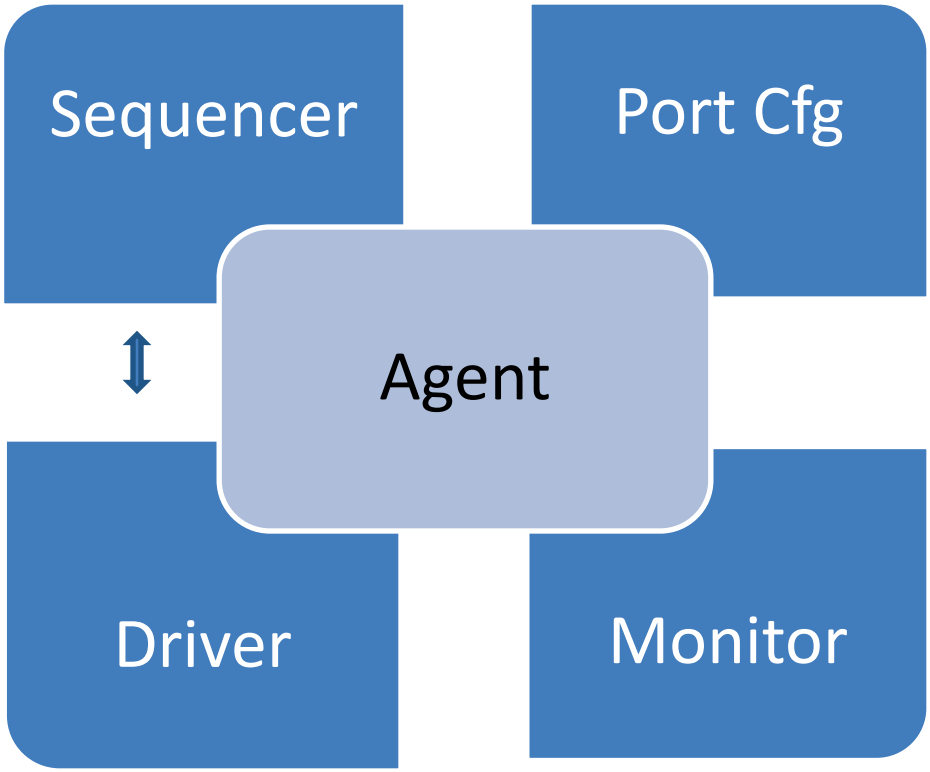
TLM Verification challenges

- Large and complex models
- Generic versus Characterized TLM models
- Creation of robust verification test-bench environment
  - Robust
  - Modular and reusable
  - Comprehensive protocol compliance testing
- **No established standard TLM-driven verification flow**

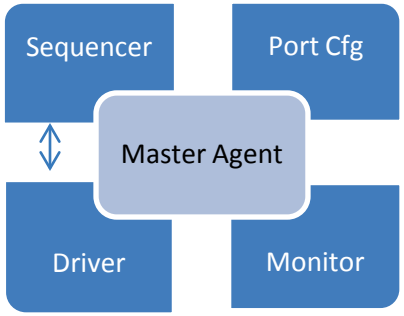
A solution: Verification IPs

- Standard methodologies for RTL design verification
- System Verilog based test-benches
  - built-in methodology support (UVM, VMM, OVM)
- Modular test-bench architecture
  - Programmable number of masters, slaves etc.
  - Features to simplify test-bench development
  - Protocol and test coverage
- Rapid creation of complex tests
  - sequence library
- Constrained random verification
- Error injection

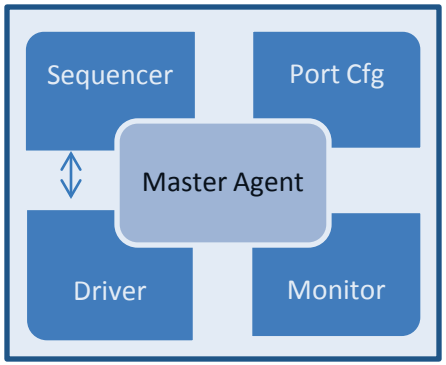
UVM VIP: An overview



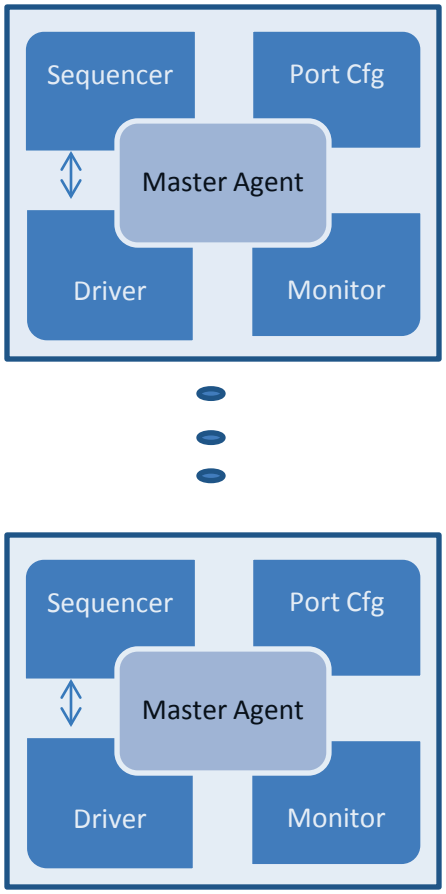
UVM VIP: An overview



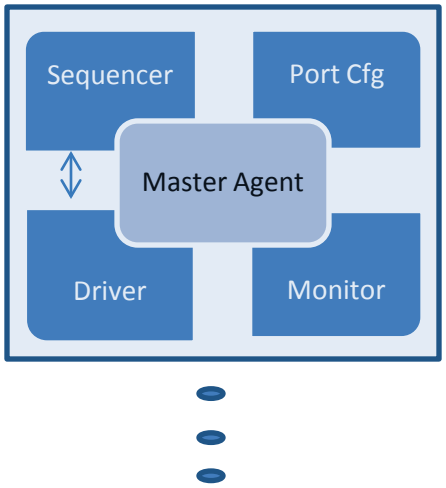
UVM VIP: An overview

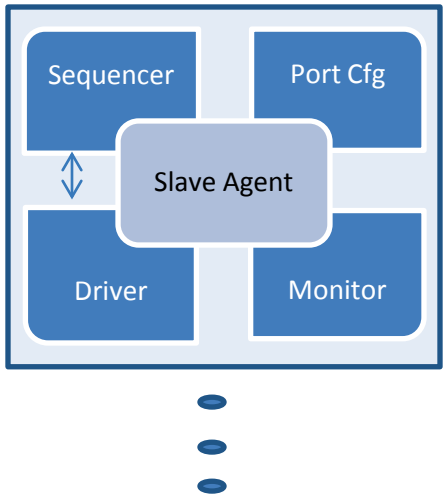


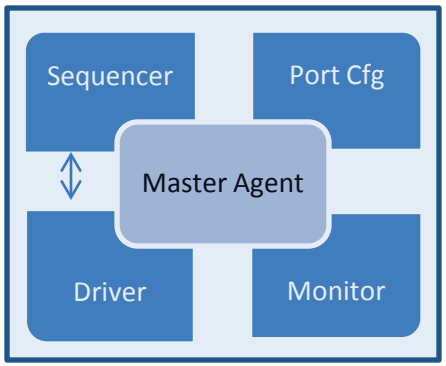
UVM VIP: An overview

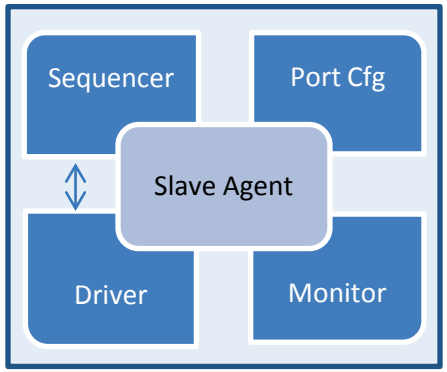


UVM VIP: An overview

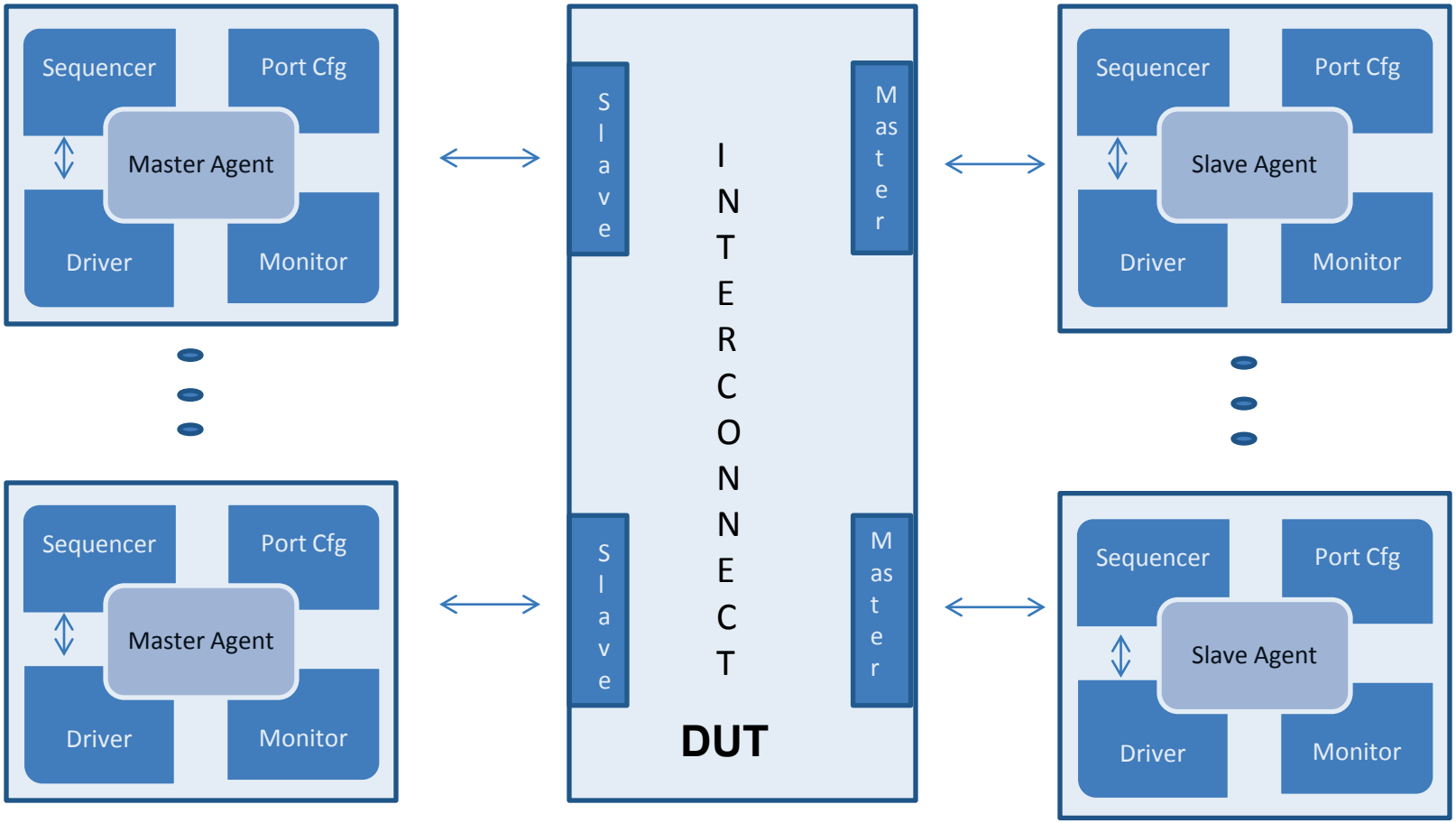




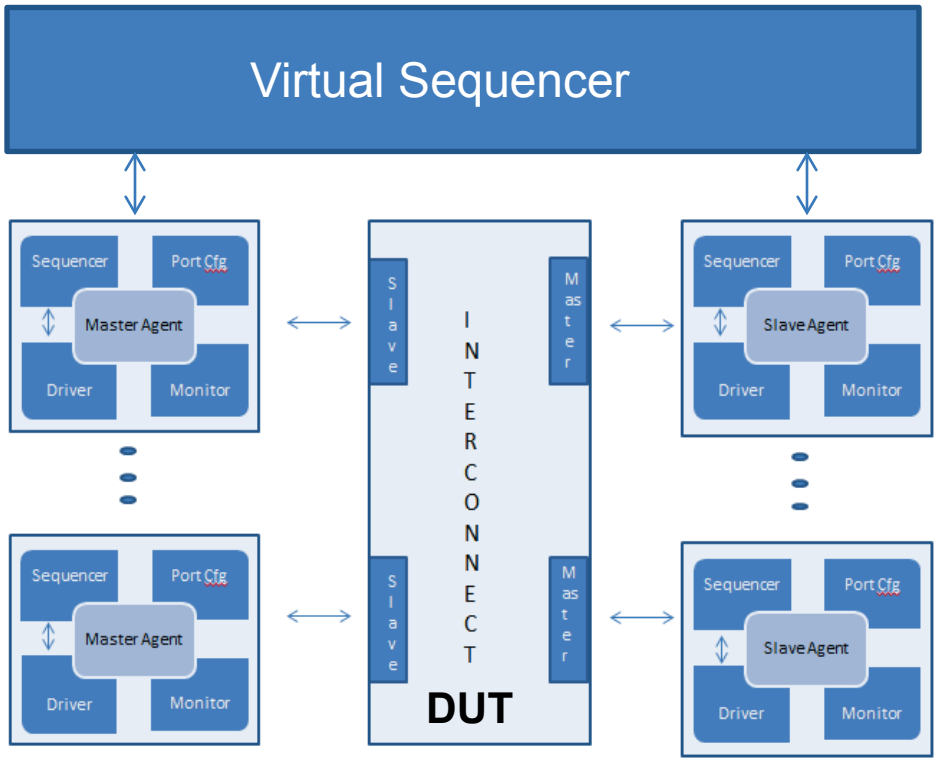




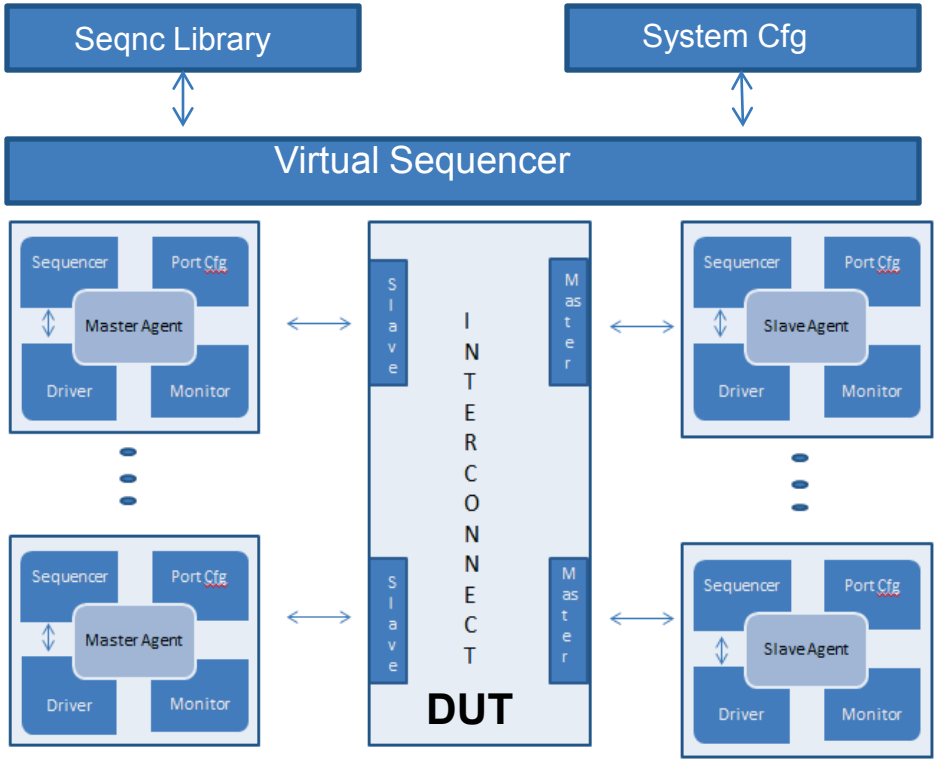
UVM VIP: An overview



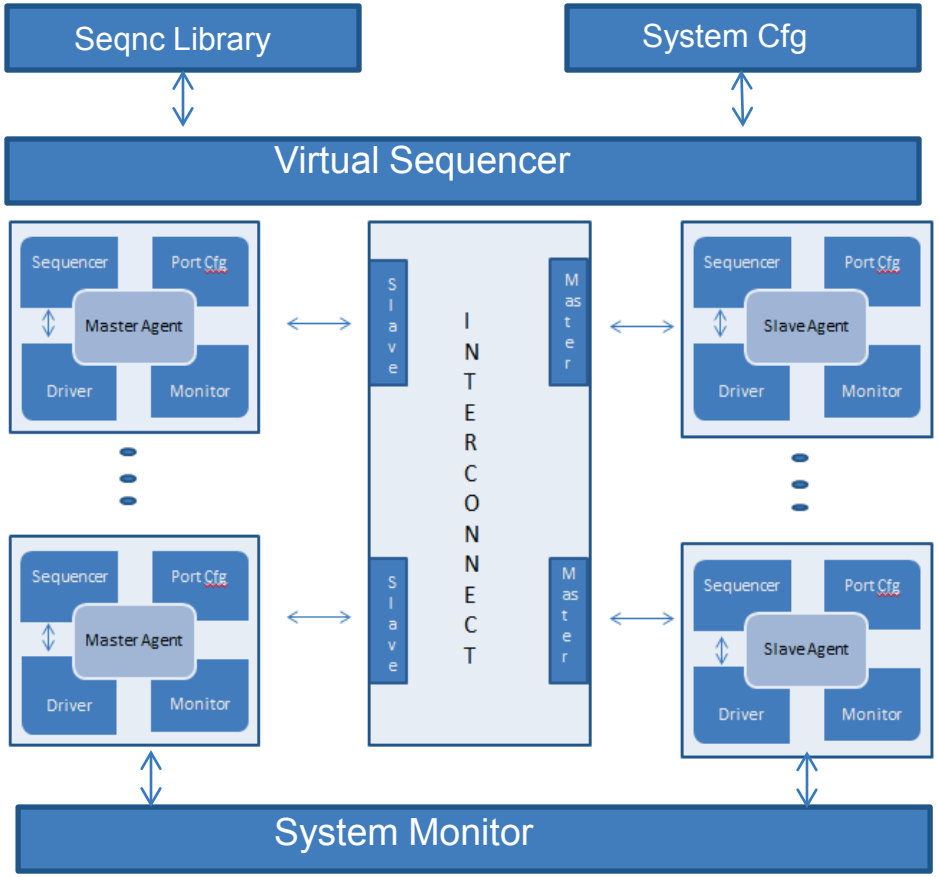
UVM VIP: An overview



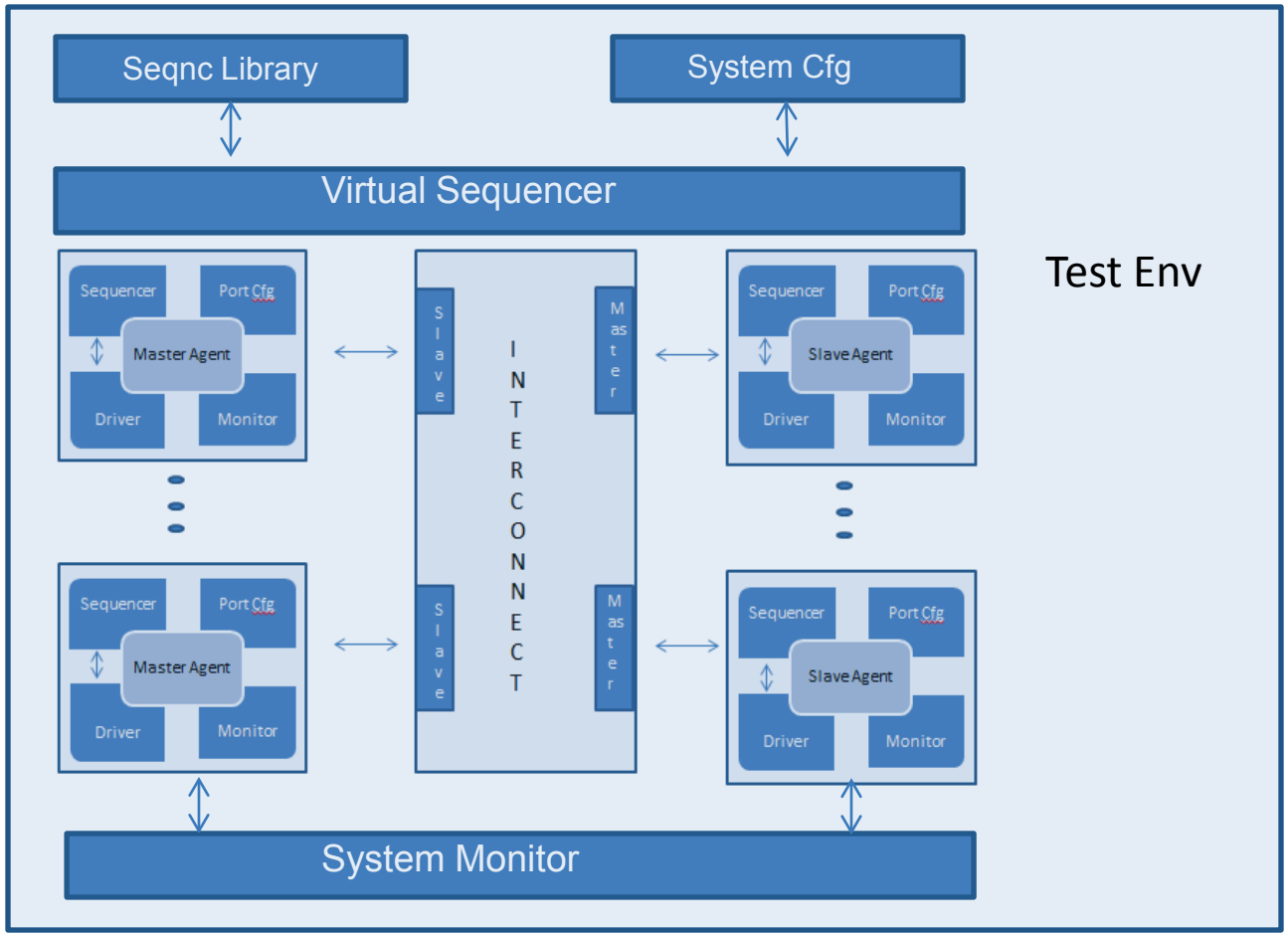
UVM VIP: An overview



UVM VIP: An overview



UVM VIP: An overview



The proposal

- UVM based VIP re-usage
  - Rework/refine UVM components (master, slave and interconnect agents)
  - Create wrappers: combination of VIP agents
- Integration with TLM models
  - Using PA-MCO, create a hybrid platform (modified SystemVerilog-UVM VIPs with SystemC-TLM IPs)
  - Co-simulate
  - Generate objective data with pre-defined/modified VIP sequences
  - Verify simulation results through VIPs

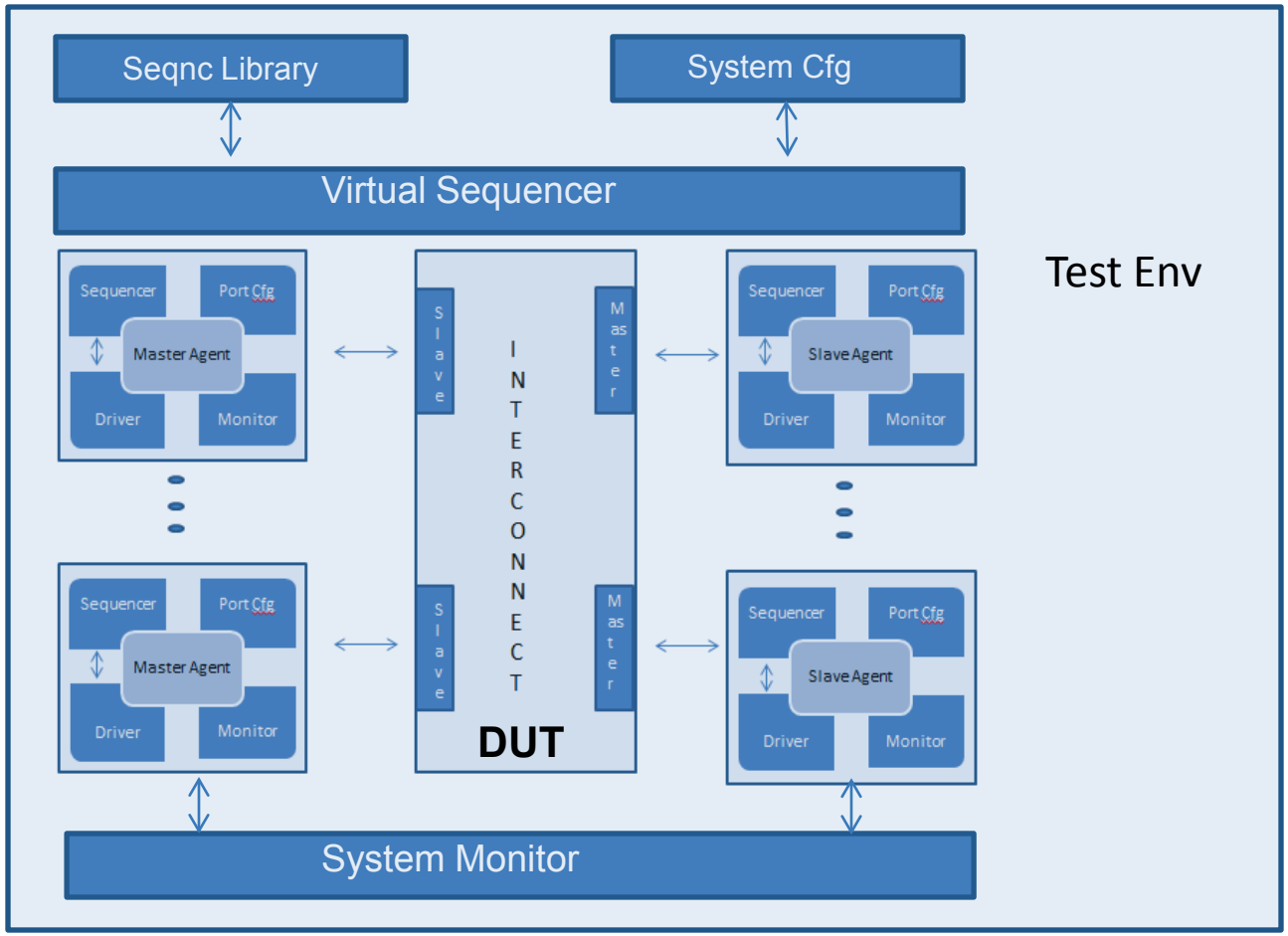
The proposal

- UVM based VIP re-usage
  - Rework/refine UVM components (master, slave and interconnect agents)
  - Create wrappers: combination of VIP agents
- Integration with TLM models
  - Using PA-MCO, create a hybrid platform (modified SystemVerilog-UVM VIPs with SystemC-TLM IPs)
  - Co-simulate
  - Generate objective data with pre-defined/modified VIP sequences
  - Verify simulation results through VIPs

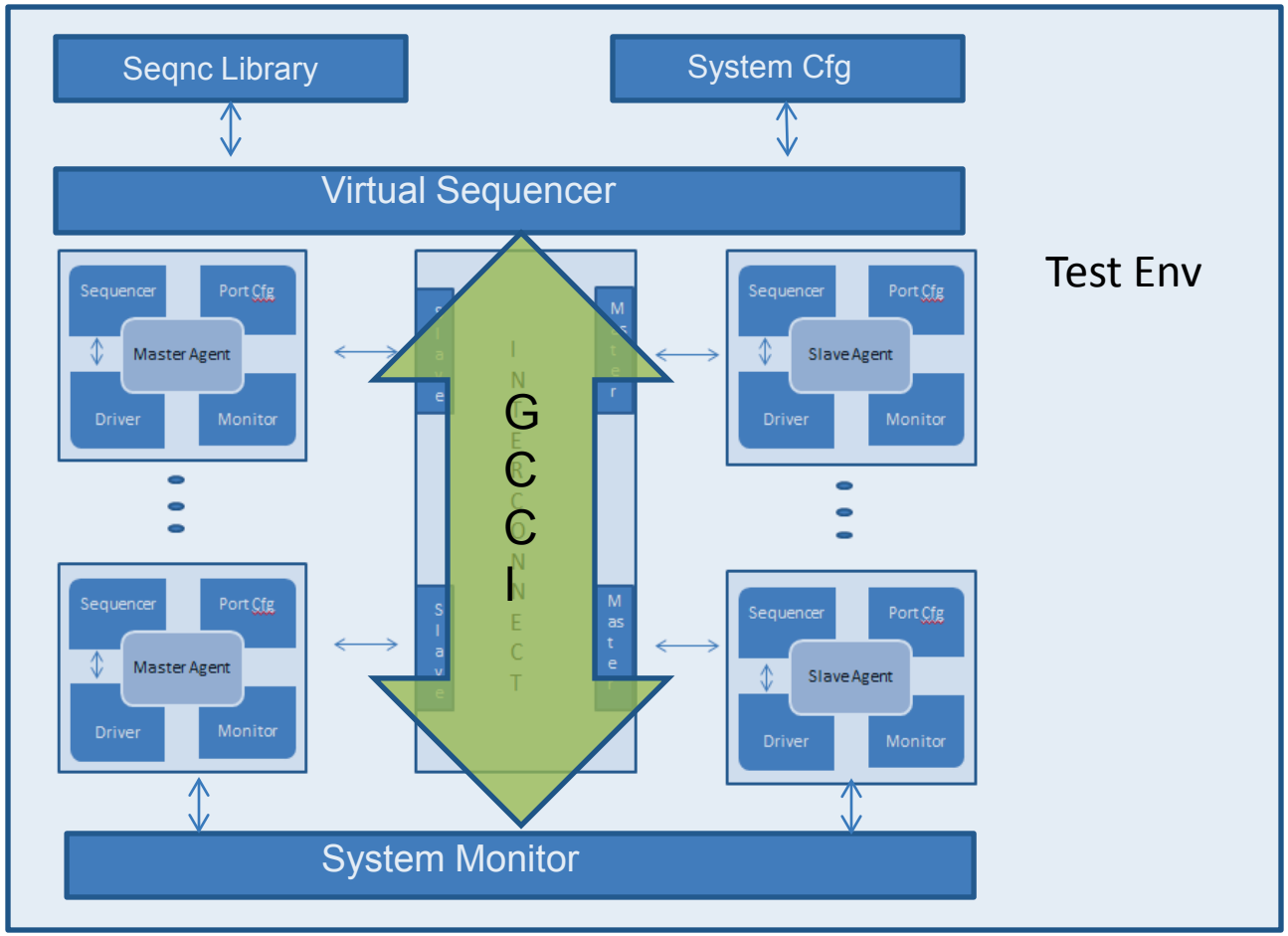
The proposal

- UVM based VIP re-usage
  - Rework/refine UVM components (master, slave and interconnect agents)
  - Create wrappers: combination of VIP agents
- Integration with TLM models
  - Using PA-MCO, create a hybrid platform (modified SystemVerilog-UVM VIPs with SystemC-TLM IPs)
  - Co-simulate
  - Generate objective data with pre-defined/modified VIP sequences
  - Verify simulation results through VIPs

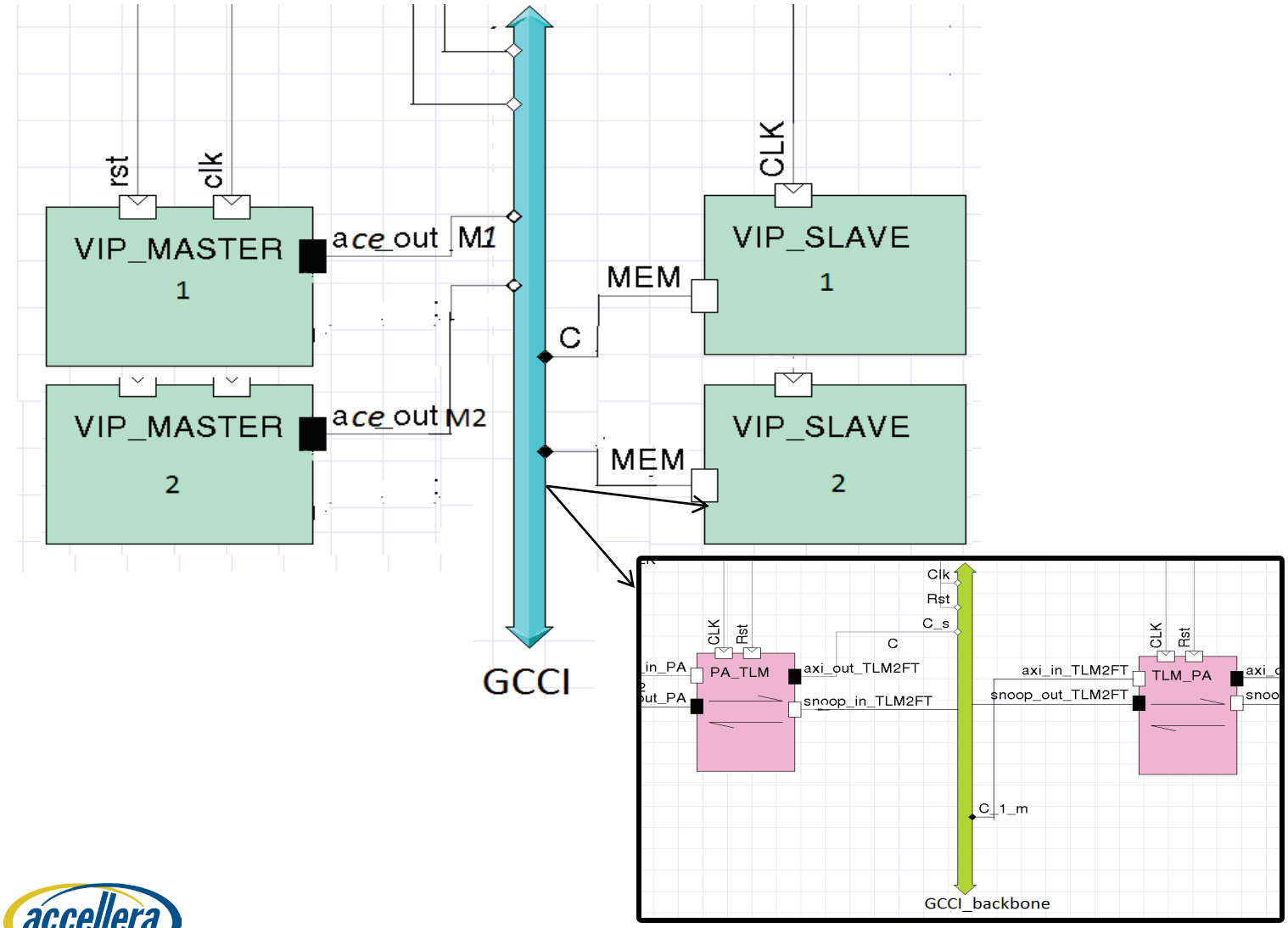
Case Study


Seqnc Library
System Cfg
Virtual Sequencer
Sequencer
Port Cfg
Master Agent
Driver
Monitor
Slave
INTERCONNECT
Master
DUT
Slave Agent
Test Env
System Monitor

Case Study


Seqnc Library
System Cfg
Virtual Sequencer
Sequencer
Port Cfg
Master Agent
Driver
Monitor
Slave Agent
GCCI
INTERCONNECT
System Monitor
Test Env

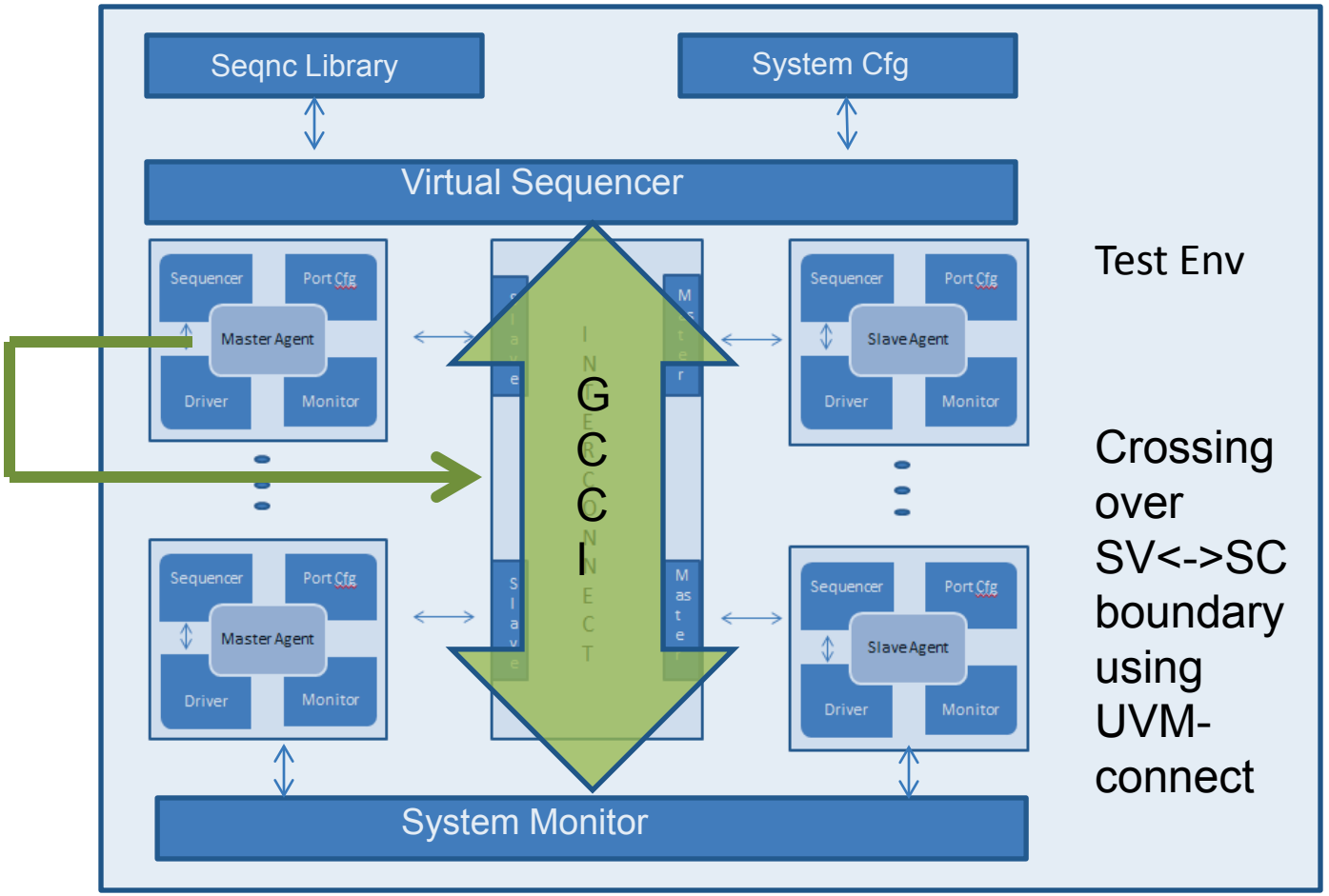
Case Study


rst
clk
VIP_MASTER
1
ace_out
M1
VIP_MASTER
2
ace_out
M2
C
MEM
MEM
CLK
VIP_SLAVE
1
VIP_SLAVE
2
GCCI
Clk
Rst
C_s
C
CLK
Rst
in_PA
PA_TLM
axi_out_TLM2FT
snoop_in_TLM2FT
axi_in_TLM2FT
snoop_out_TLM2FT
TLM_PA
C_1_m
GCCI_backbone

Case Study: Results

- The experiment helped in quality verification
  - identifying and fixing a number of hard-to-find bugs in TLM models that could not be caught using non-standardized TLM SystemC based initiators/targets
    - Untested
    - unanticipated
- Different combinations of above VIP flow possible
- Using PA-MCO, necessary test-bench modifications are nonintrusive

Alternative approach (WIP)



Conclusion

- ✓ Equivalence
- ✓ Productivity
- ✓ Efficiency

Conclusion

- ✓ **Equivalence**
- ✓ **Productivity**
- ✓ **Efficiency**

Conclusion

- ✓ **Equivalence**
- ✓ **Productivity**
- ✓ **Efficiency**

Conclusion

- ✓ **Equivalence**
- ✓ **Productivity**
- ✓ **Efficiency**

Acknowledgements/References

- Sandeep Kumar, Synopsys
- www.testbench.in
- www.vmmcentral.org

Q&A