Hybrid Simulation
A SystemC – HDL Co-simulation

T&VS - Verification Futures
13 May 2014
Introduction / Agenda

• Setting the context for need for Hybrid Simulation

• Case study: RTL IP integrated into SystemC based VP, in TI’s CCS
Authors

- Vishal Goel - Texas Instruments (TI)
- Asiful Mondal, Amit Nene – TI
- Other key contributors
  - Vignesh R - RVCE, Bangalore
  - Pratish Kumar KT - TI
  - Bharat - T&VS
Platform availability linked to Silicon milestones for new SOC Designs

For IP (RTL) available earlier than SOC, we can use Hybrid Sims (as early as Compile 0) for enabling DV, AVV as well as SW validation for an IP in the new SOC context.
Why Hybrid Simulation?

- Increasingly complex SOCs
  - Time consuming to create complete RTL Test bench as per the SOC

- IP reuse in different SOCs
  - Test benches based on different SOCs not acceptable
  - Lack of reuse of validation code across SOCs

- Very slow simulation
  - Entire SOC running in KIPS (RTL)
  - Impractical to qualify all scenarios/System level tests

- Challenges in debugging SW running on RTL
  - Need debuggers like TI-CCS & Models with greater speed, visibility and advanced debug hooks for SW level debugging
Why Hybrid Simulation?

- IP design validation
- Pre-silicon SW validation
- Overall flow improvements

Co-simulation can help improve all these phases of chip-design
**IP – Design Verification**

- Easy validation of IP RTL in the SOC context
  - Can write test cases in C, instead of Specman or System Verilog etc
  - Easy to write and maintain test cases
  - Reuse across different platform

- Reuse of C test cases across various development platforms
  - The same TCs can be reused in AVV, Silicon Validation, as well as Software drivers at SOC level

- **SW/low level drivers becomes nearly production quality before Si**
Pre-silicon SW validation

- SW Validation on actual RTL

- Greater flexibility in VP/platform integration
  - Use RTL from earlier SOC, if C-model is not available/possible

- Use RTL models if more timing accuracy is required
Overall Flow Improvements

• Improved chip development life cycle
  – By skipping RTL TB creation around IP, instead using SystemC VP akin actual SOC

• RTL fixes can be delivered much faster to SW teams rather than waiting for re-spin of entire SOC

• Faster Simulation (than RTL-only environment)

• Quick debug cycle
  – RTL – SW – co simulation and co-debugging are possible.
  – Faster Simulations help run regressions faster
Not Targeting

- Improvement of IP DV (target is SOC DV)
- Replacement of RTL environments, Specman or System Verilog, etc
- Platform architecture evaluation
  - Not accurate in terms of timing (functional Virtual Platform)
- For Complex IPs
  - Speed degradation and challenges in synchronization
Co – simulation - Techniques

• Direct Signal Interface (DSI)
  – A pin/signal level interfacing technique between HDL and SystemC models

• Direct Programming Interface (DPI)
  – Allows direct inter – language function calls between the languages on either side of the interface

Work presented in this paper is based on Direct Signal Interface
Targeted Use case

• IP Validation of “DSS” (Display Sub System) available as RTL, with OCP interface

• Rest of the SOC implemented using SystemC/TLM2 – TL4
Proposed Solution

- Direct Signal I/f
- Integrate RTL (TL0) and SystemC (TL4)
- Develop adaptors: Generic TL4 $\leftrightarrow$ OCP TL0
- Connect SystemC and RTL simulation environments using TCP/IP

Environment/Tools
- Third Party EDA tool for RTL simulation
- SystemC based simulator for Cortex – M4
- TI Code Composer Studio as Debugger/IDE
- Demonstrated on OCP protocol based IPs.
Proposed Solution – Top level view

- Cortex – M4
- Interconnect
- DSS Proxy (client)
  - a.b.c.d:xxxx
- IP1
- IPn
- DDR
- TI Debugger
- SystemC
- TCP/IP
- RTL Environment
  - proxy Server
  - TL4->TL0
  - TL4<-TL0
  - RTL
Proposed Solution - Adaptors

- **Reused OCP TLM tool kit**
  - Separate adaptors for TL3->TL1 and TL1->TL0 available from OCP
  - Created: Generic TL4->TL0(ocp) and TL0(ocp)->GenericTL4, encapsulating the above two

- **Adaptors**
  - Clock from RTL environment
  - Provide configuration APIs to help map to OCP configuration
Proposed Solution - IPC

- TCP/IP b/w SystemC/TI Debugger and RTL

- Proxy Client for TI Debugger
  - TLM/memory mapped in place of actual IP
  - Sends/receives payload received from CPU R/W commands to Server

- Server for RTL simulator
  - A separate thread in RTL simulator
  - Polls TCP/IP port for any R/W request
  - Forwards R/W requests to RTL using TL4 to TL0 OCP adaptors
Declaring SystemC module in Verilog. SystemC Class name must match Verilog module name. Signal names should be same on both sides.

```verilog
module testbench10 (Clk, NAddr, MData, NReset_n, NConnID, MBurstLength, MRqInfo, NByteEn, NRespAccept, NRespLast, SData, SCmdAccept, SResp, SRespLast);

// Instantiate SystemC master

instance SystemC master

endmodule
```

Testbench10 tb(
  .Clk
  .NAddr
  .MData
  .NReset_n
  .NConnID
  .MBurstLength
  .MRqInfo
  .NByteEn
  .NRespAccept
  .NRespLast
  .SData
  .SCmdAccept
  .SResp
  .SRespLast
);
Test Environment

- C test cases to configure DSS and trigger DSS DMA transfer.
- C test case is compiled with ARM compiler
- Executable is run on Cortex –M4 core on SystemC Simulator in CCS.
Results

• Able to run basic SOC level test cases involving the DSS, under TI CCS (Windows)

• Adaptors for TL4->TL0 and TL0->TL4 developed, which can be reused with other OCP based IPs

• Fairly fast simulation for simple test cases involving DSS processing and DMA transfers

• Could attach RTL, SystemC, and SW debuggers – all at the same time

• Can use this platform to realize all the benefits targeted from a co-simulation
Conclusion

• DV can benefit from co-sim due to
  – **Early availability of test benches**, which matches target SOC
  – Reuse of C-test cases from hybrid to emulation platforms to final Si, for all SOC variants

• Hybrid Sim can **improve quality of Pre-Si SW validation**
  – Using actual RTL for SW validation
  – Catch specs issues (early engagement)
  – Can help w/ correlation of spec, C-model and RTL

• **Debugging** the RTL/SW issues is **faster**
  – SW and HW team can do jointly debug a scenario in their respective debug environment.
Limitation and Further work

• Issues / Limitations with proposed solution
  – Synchronizations for complex IP’s yet to be addressed

• Scope for further work
  – Adding interrupts and run the complete SW stack in this hybrid environment
  – Experiment with IPs with other bus protocols like CBA
References


- “SystemC Verilog Co-simulation for Virtual Platforms and Architecture” - [ISCUG 2013](http://www.accellera.org/community/ocp/)
Thank you