Hardware Software Co-verification in Hybrid QEMU/HDL Environment

Radosław Nawrot, Aldec Inc.
Krzysztof Szczur, Aldec Inc.
SoC verification challenges

- Contains complex hardware and embedded software
  - both need verification
  - software on critical-path
- Heterogeneous architecture and hierarchical Network-on-Chip (NoC)
  - Corner case hardware problems
  - System level integration problems
- Standard IP reuse: CPU, DMA, MEM ... - side effects:
  - Design size fast growth
  - Slow HDL simulation
- Custom IP-Cores
  - Need cycle accuracy for debugging
  - Cannot be fully verified at block level (out of SoC context)
SoC Designs in FPGA Domain

Xilinx Zynq and the like...

- **Processing System**
  - ARM Cortex CPU
  - Standard I/O peripherals

- **Programmable Logic**
  - Classic FPGA used for:
    - More peripherals
    - Glue logic & Bridge
Virtual Platforms & Virtual Machines Pros & Cons

(+) Available earlier than system-level RTL code
(+) Fast enough for firmware/software development
(+) Integrated with software development tool-chain

(-) Not cycle accurate
(-) Virtual models not available for Custom-IP

Virtual Machine

Versatile – many architectures:
• ARM, RISC-V, SPARC, MIPS
• x86, PowerPC, MIPS

Virtual Platforms
Hybrid co-simulation and co-emulation

- Custom IP development
- Block-level verification
- QEMU – software driven testbench

- SoC Integration & bring up
- Drivers & firmware development
- Complete SoC model for SDK
- QEMU – integral part of SoC model
QEMU co-simulation with Riviera-PRO
QEMU co-simulation platform

- Parts of the flow:
  - Riviera-PRO Advance Verification Platform
  - Aldec AXI BFM
  - Aldec QEMU Bridge
  - QEMU Emulator
QEMU co-simulation highlights

• Full debug capabilities of RTL IP Core in Riviera-PRO simulator:
  – Waveforms
  – Hardware Breakpoints
  – Hardware steps
  – Transaction based verification and debug

• Kernel and driver debug via GDB
  – Software Breakpoints
  – Variable probing

• Zynq Linux OS ready to use on QEMU without modifications
Riviera-PRO Verification Platform

A Verification Platform that grows with your requirements.

- **System on Chip**
  - Internet of Things

- **System Interfaces**
  - MATLAB®, SystemVue®

- **Debugging**
  - Mixed-Language IDE, Code Tracing,
    - Waveform, Dataflow, Image Processing, DSP

- **Metrics**
  - Coverage, Test Rank, Assertions, Code and Functional Coverage, SVA/PSL/OVM

- **Test Automation**
  - TLM, RTL, Gate-Level, SystemVerilog,
    - SystemC, Python, UVM, C/C++

- **Simulation**
  - Mixed-Language, Mixed-Signal, Hardware/Software,
    - Verilog, Verilog-AMS, VHDL

© Accellera Systems Initiative
Riviera-PRO Highlights

• High Performance Simulation
  – Extensive simulation optimization algorithms
  – Support for latest Verification Libraries: UVM, OSVVM, UVVM, CocoTB and more

• Advanced Debugging
  – Transaction Level simulation end debug
  – Multi-language debug environment (Verilog, VHDL, SystemVerilog, SystemC, Verilog-AMS)
  – Support for MATLAB and Simulink
  – C/C++ debug environment
  – Support for external C/C++ compilers (GCC, Visual C++)
  – UVM Toolbox, Graph and Class Viewer
  – Code tracing, Waveform, Dataflow, FSM window, Coverage, assertions, memory visualization
  – Comprehensive Assertions-Based Verification (SVA and PSL)
  – Advanced Code and Functional Coverage
  – User-defined test plan linking with coverage database
  – Plot Viewer and Image Viewer
QEMU Highlights

• Free and open-source
• Fast software emulator
• Supports many machines:
  – Xilinx-zynq-a9
• Supports many ARM processors:
  – ARM926
  – ARM946
  – Cortex-A8
  – Cortex-A9
  – Cortex-A15
  – Cortex-A53
  – Cortex-A57

© Accellera Systems Initiative
Connecting QEMU with Riviera-PRO

QEMU Emulator
(Linux for TYSOM board) with
Xilinx SystemC SoC Library
ARM Cortex-A9 model

ARM application
running on Linux

Aldec Riviera-PRO

Aldec QEMU Bridge

Aldec AXI BFM

Custom IP Core

© Accellera Systems Initiative
Aldec QEMU Bridge
AXI BFM – HDL site QEMU interface

- AXI 3 Master
- AXI 3 Slave
- AXI 4 Master
- AXI 4 Slave
- AXI 4 Lite Master
- AXI 4 Lite Slave
- AXI 4 Stream Master
- AXI 4 Stream Slave
Aldec AXI BFM usage

- Distributed with precompiled libraries
- Instantiation in HDL part of the project

- Configuration set by parameters
- Turn on build-in transaction recorder

- Automatic connection with Aldec QEMU Bridge process done at simulation stage
AXI BFM instance in HDL Code
AXI BFM instance in HDL Code
Riviera-PRO QEMU environment
Hardware and Software co-sim
Built-in Transaction Recorder
Transaction based debug
Transaction logs
Summary

• SoC FPGA Verification Platform
• Processor modeling based on QEMU emulator (instead of simplified BFM model)
• Hardware and Software debugging
• Fast RTL IP Core Verification and bugs fixing
• Driver and Firmware Verification
• Built-in Transaction Recorder
• Support for many platforms and processors
• Support for different Linux versions
  – Yocto project
QEMU co-emulation with HES-DVM
Emulation Pros & Cons

(+) Fast – 10+ MHz speed
(+) Cycle Accurate hardware model
(+) Advanced Hardware Debugging
(+) Enables testing with real devices/live data

(-) Emulation available later than RTL Simulation
(-) Usually limited emulation resources
HES-DVM™ Versatile Emulation Platform

In Circuit Emulation

Simulation Acceleration

Hybrid Co-emulation

Virtual Machine

Virtual Platforms

SCE-MI

PLI/VHPI

SCE-MI

TLM

SoC Integration

Software Debugging

Hardware Debugging

TCP-IP

JTAG

• JTAG Port
  • vJTAG Xtor

Break

Capture

Run
Connecting Virtual Platforms

SoC Bus Transactors (BFM with SCE-MI)

- **AMBA** – Advanced Microcontroller Bus Architecture
  - AXI Master & Slave
  - AHB Master & Slave
  - APB Master & Slave
- **OCP** – Open Core Protocol
  - Master & Slave

Virtual Platform Plug & Play

- **HES PCIe host driver** (Linux/Windows)
- **SystemC/TLM Proxy** for each bus transactor
- **SCE-MI** – Accellera Standard Co-Emulation Modeling Interface
Software debugging

Where is the CPU core?

- **In Virtual Platform**
  - VP provides interface
- **In HES Emulator**
  - Aldec provides solution:
    - vJTAG Transactor
    - vJTAG Virtual Peripheral library with TCP-IP

![Diagram of software debugging setup](image-url)
Hardware debugging

- Debug probes – 100% visibility
- Breakpoints and Triggers
- Clock control
- True RTL debugging
- Memory visibility
QEMU Co-Emulation

© Accellera Systems Initiative
QEMU PCI to AXI bridge
QEMU Co-Emulation Highlights

• Synchronized Hardware/Software Debug
  – Force Break – stop hardware clocks on software breakpoint
  – Software debugger integration (GDB), custom commands

• AMBA AXI interface support
  – Aldec AXI Master transactor VIP

• PCI to AXI Bridge

• QEMU virtual machine architectures with PCI device tree (ARM, x86 and others)

• Integrated with Linux Guest OS in QEMU

• Automated design compilation in HES-DVM platform
QEMU Co-Emulation Live Demo

© Accellera Systems Initiative 33
Questions