Fast Forward your Software Development with Advanced Hybrid Technologies

Xiaowei Pan, AE Manager
Synopsys China
Agenda

• Introduction

• Synopsys Hybrid Technologies

• Use-cases and Successes

• Synopsys Hybrid Innovation

• Conclusion

• Q&A
SoC Design Cycles are Getting Longer

Driven by Increasing Software Content and SoC Complexity
1. **Verification and Software Shift Left** Verification Continuum platform
2. **Validation Shift Left** pre-silicon ← post-silicon
3. **Enablement Shift Left** pre-silicon ← end-customer SW bring-up
How to Verify HW/SW Systems Faster?

Virtual Prototyping
Have we exercised real-RTL?

Emulation
Is the speed good enough for SW Development?

FPGA Prototyping
Do I have Prototype-ready RTL and Capacity?

How about a Hybrid?
Combining best of all the three for Faster to SW
SoC Verification Flow

**Performance & Power** – *Architecture Analysis, Optimization & Verification*

- Platform Architect
- ZeBu
- HAPS
- Post-Silicon

**Software Enablement** – *Bring-up, Validation & Optimization*

- Virtualizer
- Virtual Host
- ZeBu
- HAPS

**Function** – *Block & SoC Verification, HW/SW Verification & System Validation*

- VC Formal DPV
- VCS
- ZeBu
- HAPS
- Post-Silicon
Synopsys Hybrid Solution

• Hybrid Architecture Analysis
  – Efficient architecture analysis and smart performance monitoring
  – Offline and online analysis of performance data collected on ZeBu
  – Faster PPA optimization, helping shift left verification cycle

• Hybrid Emulation
  – Early Driver/Firmware/Application development
  – Complete SoC model using VDK and synthesizable RTL IPs
  – Power and performance validation over billions of application cycles

• Hybrid Prototyping
  – Software driven system validation with real-world IO
  – Early Driver/Firmware/Application development
  – Modular and scalable validation from IP to system level
Software Driven Verification Project Flow

Production Software Stack → SW Signoff before Tapeout

Continuous SW Driven Development/Verification Platforms

Hybrid Emulation

- CPU Model + IP/SoC RTL
  - VDK
  - Hybrid
  - ZeBu
  - 10+MHz

- CPU Model + IP/SubSys RTL
  - VDK
  - Hybrid
  - HAPS
  - 20+MHz

Hybrid Prototyping

- SoC RTL
  - ZeBu
  - 1+MHz

- SoC RTL
  - HAPS
  - 10~200MHz

SW Driven
- Full System Validation
- Development Platform
- IP/Subsystem Validation
Agenda

• Introduction
• Synopsys Hybrid Technologies

• Use-cases and Successes

• Synopsys Hybrid Innovation
• Conclusion
• Q&A
Use Case: Virtual I/O

Using VDK to Bring up AI SoC Firmware

- PCIe Virtual I/O for AI HW/SW Integration & Optimization
- AI Host OS and SW Tool chain SW running on VirtualBox
- VDK connected to VirtualBox via PCIe Virtual I/O

Leading AI SoC Company brings-up SoC FW 3~5 months before RTL

Optimize SW Tool chain
Use Case: Virtual Host

Reuse VirtualBox setup for ZeBu HW/SW validation

- Same VirtualBox setup
- Speed up HW/SW validation and turning performance
- Re-use mature host side AI SW tool chain, driver, and SoC Firmware developed based on VDK environment

Smooth Transition to Emulation for Validation
Use Case: Hybrid Emulation

5G Modem Full HW/SW Verification Platform

- Start full-stack 5G HW-SW co-verification and testing 6~9 months earlier
- OS boot within 10 minutes - run one 5G case in 15 min
- Users identified more than 100 HW/SW issues on hybrid platform before tape out
- More productive HW/SW debugging
Use Case: Hybrid Prototyping

Accelerating SW Development and RTL Validation

- Easy Setup
  - VDK with custom OS
  - Different IPs in HAPS
  - Smaller prototype
  - Faster prototype compile

- Debug Productivity
  - Boots OS in <1 minute
  - OS trace/profiling helps HW/SW debug and SW performance analysis
  - Full HW debug visibility for registers/pins

- Fast Speed
  - 4K 100 Frames Encoding took 3mins to complete in hybrid, while in pure FPGA it took 2+ minutes

Design Type: Drone SoC

Leading Asia Drone Company boots Linux in <1 minute
Customer Successes

Spreadtrum Adopts Hybrid for AP & Modem SoC

Successful AI Edge Design Case: A Leading AI System Company
VDK → Hybrid → Zebu Smooth Design/Verification Flow

Socionext Hybrid Prototyping

FPGA Prototyping
- Difficult to implement large SoC to 1 chip FPGA
- Implement too huge circuit effect to performance
- Divide to multiple FPGA has more load
- Smooth connection to SW design env

Hybrid Prototyping
- Minimize circuit implementing to FPGA
- Reduce risk to slow performance
- Use VDK for CPU sub system
- Flexible to connect SW design flow
- Easy to change memory map
- Reuse by changing target IPs

Use as internal tool
Use as sales tool for external customer

Hybrid Emulation for faster SW development
Presented by NXP and Synopsys at SNUG World 2021

- Increasing demand for software bring-up on emulator for complex embedded multi-core SoCs
- Hybrid emulation technology provides shift-left approach and helps to reduce risk and cost
- Fast setup with large library of models and transactors in Virtualizer and Zebu
- Hybrid FastMen technology enables high simulation speed for shared memory access
- Planning to setup automated regression tests and explore power and performance use-cases
Customer Successes

Spreadtrum Adopts Hybrid for AP & Modem SoC

Socionext Hybrid Prototyping

Enabling Aggressive Mobile SoC Schedules

Samsung Mobile Devices drives rapidly changing requirements: Samsung SLSI delivers on time

Left-Shifting Paradigm for SW Development/Verification

Results: OS Boot Time @ Mobile SOC

- DAC 2018

**Benefits:**
-ピン&Subsystem validation testcases run in minutes in hybrid; in pure EMU run hours to boot Linux and do testcase on top of it
- SW img. load at runtime after Linux boot
- Save design size in Zebu as rest of design in VDK side
- Using pure emulation to do HW/SW final validation
- Leak SW debug in pure emulation as SW tested/debugged in VDK or hybrid platforms
Agenda

• Introduction
• Synopsys Hybrid Technologies
• Use-cases and Successes

• Synopsys Hybrid Innovation
• Conclusion
• Q&A
Typical Hybrid Block Diagram Overview

Virtualizer

- Virtualizer Debugger Infrastructure
- GIC
- Cortex Clusters
- Hybrid Register Server
- DDR
- RAM
- Storage Controller
- UART
- Timer
- Card/Disk
- File System .bin

ZeBu/HAPS

- SoC Blocks (Multimedia, …)
- CPU Sub-System & … Black Boxed

- SW / Linux Images
- FastMem
- Memory
- Memory
- XTOR
- Adapt
- resets

- ZeBu
- HAPS
10x Faster Hybrid Creation

Configurable Hybrid Building Block

- 10x faster Hybrid setup
- Fully Automated
- Correct by construction
- Pre-configured ZeBu Testbenches for UART/CSI/DSI
- Automatically generate templates for Custom monitors and testbenches
Higher Hybrid Debug Productivity

Unified ZeBu-Virtualizer views for Single-window operation

- **RTL browser**
  - Access to all DUT signals

- **Runtime monitor**
  - System Clock Status
  - XTORs Readiness Status

- **Runtime control**
  - Waveform dump (FWC, QiWC, Dynamic Probe)
  - Triggers
  - Forces
Hybrid Checkpoint Restore

Enables fast turn around time for boot & run, debugging

Standard Sequence

1. **Start VDK**
2. **Load SW**
3. **Initialize ZeBu & Load DB**
4. **Boot, Apps, Tests**
5. **X min X Seconds**
6. **Y min**

Runs through the OS Boot, load & run applications

Checkpoint-Restore Sequence

1. **Start VDK**
2. **Load SW**
3. **Initialize ZeBu & Load DB**
4. **Boot OS, Checkpoint, Run Apps etc**
5. **Next time: Restore, Run Apps**
6. **X min X Seconds**
7. **Y min**

Rerun boot

- **Bypass boot sequence for consecutive tests**
- **Save Hybrid Setup**
  - HW DUT in ZeBu
  - Virtualizer Platform including files and configurations
- **Restore Hybrid State**
Hybrid Clock Regulator

Time alignment between Software and Hardware

• **Purpose**
  - Resolve timeout issues between Fast (Software) and Slow (DUT in Hardware) side Hybrid components
  - Maintain speed

• **Technology**
  - Clock regulator to align timer in Virtualizer with execution in ZeBu
  - No need to “hack” Software
  - No trial and error to figure out timer & ZeBu clock settings
Faster Debug with Hot Reset

Accelerate the debug TAT by faster Hybrid emulation restart

Standard Sequence

- Start VDK
- Load SW
- Initialize ZeBu & Load DB
- X min X Seconds

Found a SW Bug, potentially fixed it, Need to restart to debug again

Hybrid Emulation

Wait the Zebu to reload the DB

X Seconds

Hot-Reset Sequence

- Start VDK
- Load SW
- Initialize ZeBu & Load DB
- X min X Seconds

Found a SW Bug, potentially fixed it, Need to restart to debug again

Hybrid Emulation

Almost no wait time to restart the Hybrid simulation

• Reduce the restart time to only a few seconds

• No Design Database reload

• Load and test multiple SW Images at Runtime
Validate Performance at High Speed

Smart Monitors for Performance and Transactional Analysis  
ZeBu

- Performance monitoring at SoC Interconnects
  - Throughput, Latencies Command/Data/Transactions
  - Assertion-based max command latency violation checks

- Transactional Monitoring for Memories and Bus Interfaces
  - < 5% emulation speed impact

DUT instrumented with SPM

- Automatic launch & configuration
- Fast & efficient analysis database
- Stream performance analysis
  - Batch / Post-run
  - Dynamic / On-line

- SV insertion in DUT
- Fast pre-processing
- Fast DPI-C interface

Platform Architect

- CPU
- NPU
- GPU
- Monitor
- AXI Interconnect
- DDR Controller
- ISP
- MIPI

ZeBu Platform Architect Monitor

- Monitor
- Monitor
- Monitor

accellera SYSTEMS INITIATIVE
Conclusion

- Software and System Validation drives more than 50% of design cost and timelines
- Early HW/SW system verification is key for time to market
- Synopsys Hybrid is widely adopted in the industry to address early HW/SW system verification
- Synopsys has the largest investment in virtual prototyping, emulation and continues to innovate in new hybrid technologies
Thank You

Q & A