JULY 9–13, 2023

MOSCONE WEST CENTER
SAN FRANCISCO, CA, USA

Design for Verification

IP-level Prospectives

Pouya Taatizadeh
Google

Challenges

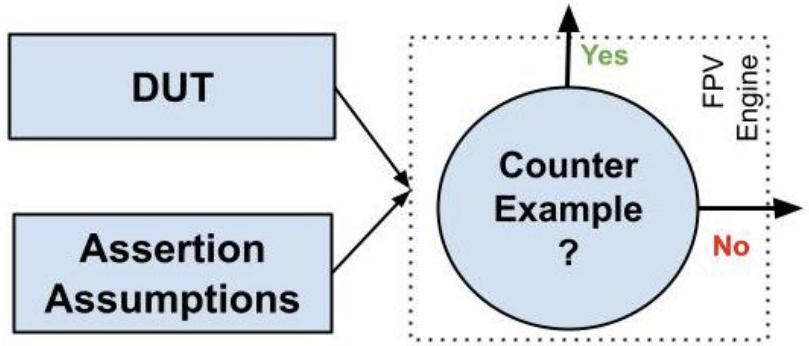
- Rising complexity of SoCs
  - Multi-Die, Chiplet-based architectures
    - Extra long Full-Chip simulation times
- Vertical reuse in different steps
  - Pre-Si assertions in Emulation
  - UVM sequences in Post-Si
  - SV constraints in Emulation and/or Post-Si
- Scarce supply of motivated DV engineers ☺️
  - Insufficient focus on DV in college curriculum
  - College grads opt-in for design vs. DV

Current Practices

- **U**niversal **V**erification **M**ethodology (UVM)
  - Re-usable agents across projects
  - Fits well with SV constrained-random stimulus generation
  - Provide means of TLM
  - Traditionally relies on dynamic simulation
    - Slower for larger designs
  - Development turn-around time for new IPs

Current Practices

- **F**ormal **P**roperty **V**erification (FPV)
  - Gaining momentum in the last few years ✅
  - High confidence ✅
  - No need for stimulus ✅
  - Not suitable for SoC DV ❌
  - Staffing ❌



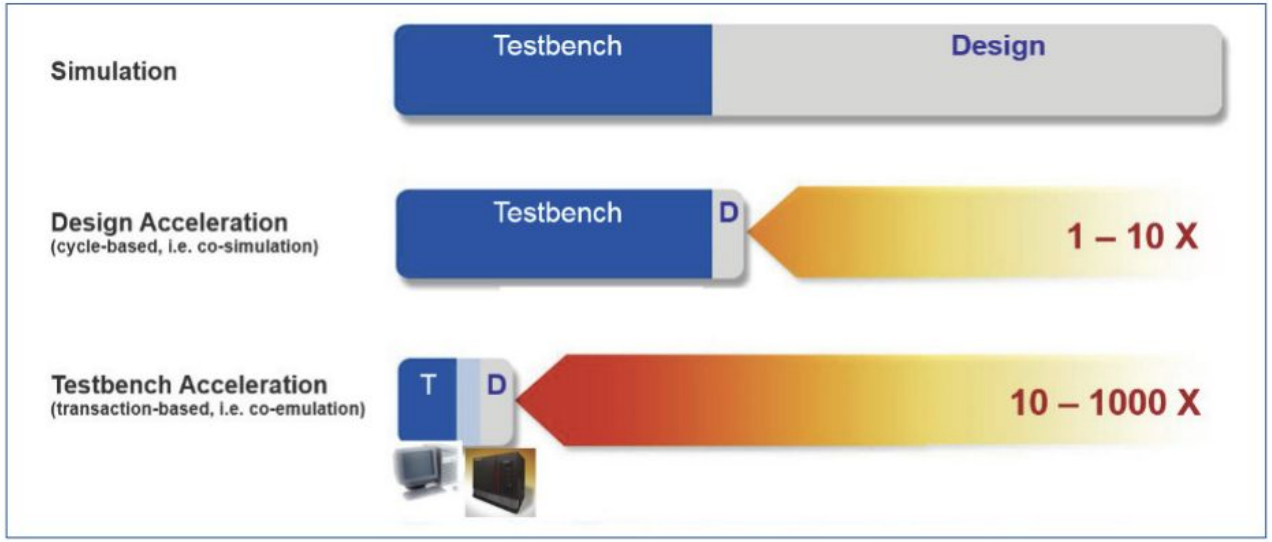
Layered verification

- Level 1 (Block-level)
  - UVM, FPV, ABV
- Level 2 (Sub-system level)
  - UVM, ABV
- Level 3 (Full-chip level)
  - SV DPI-C calls to C-based tests
  - FPV for connectivity

- Need significant compute and human resource
- Hard to execute in parallel
- Emulation ?
- **D**esign **f**or **V**erification (DfV)?
- Emulation + DfV ?

Emulation

- Speed-up simulation speeds
  - DUT running in hardware emulator
  - TB running in Simulator
- Only 2-5x speed-up
  - If tightly coupled with tb running ir a workstation
- 10-1000x speed-up
  - TLM level testbench interaction

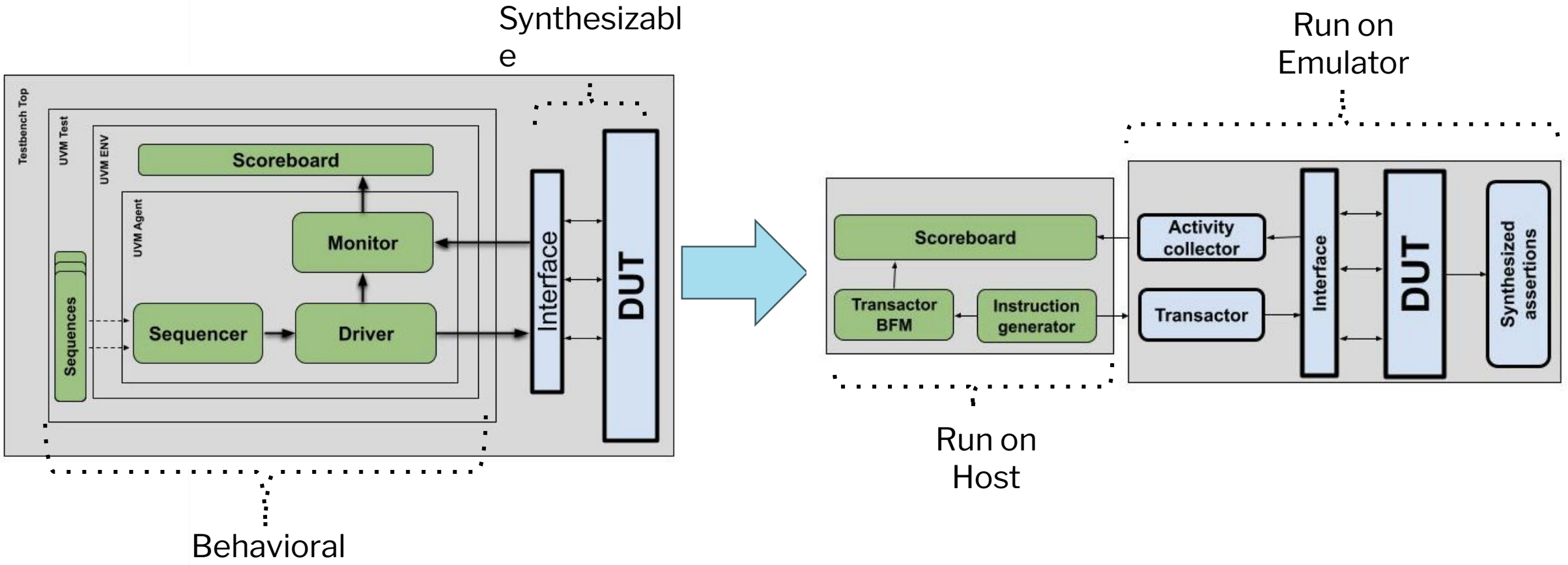


Source : Reducing Design Risk with Testbench Acceleration, Mentor Graphics, 2016

Design for Verification

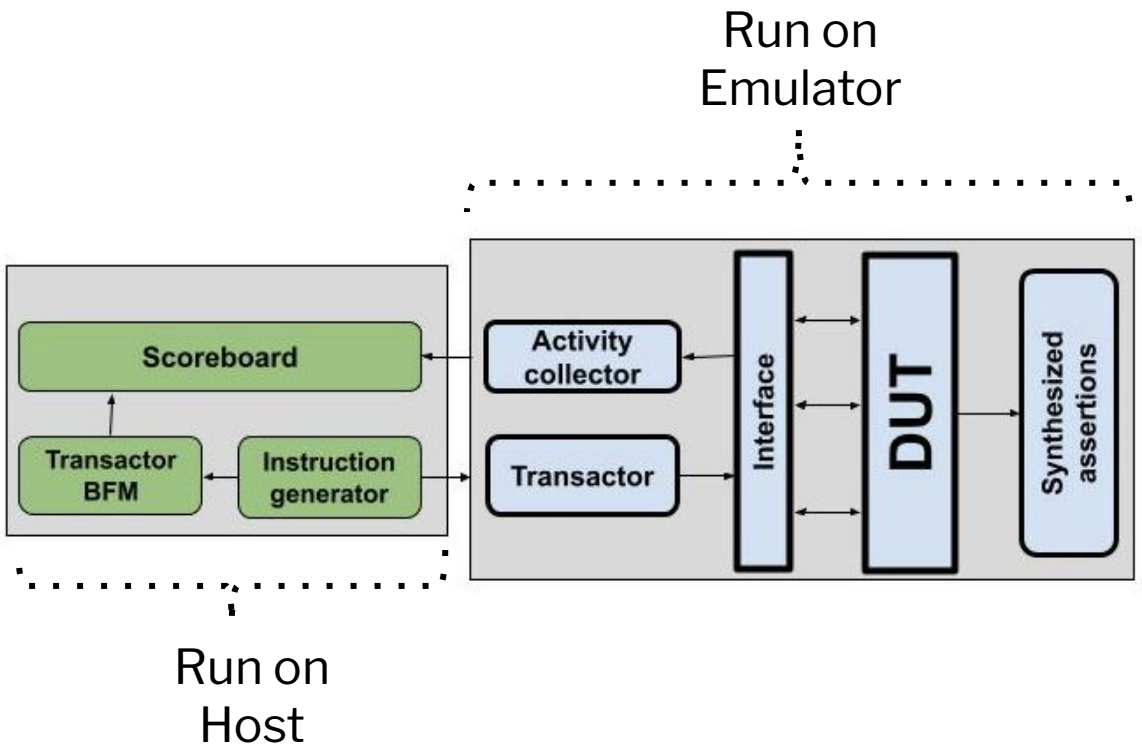
- Introduce synthesizable blocks/IPs that are added to the DUT
  - Specialized hardware blocks
    - Monitors
    - Transactors
    - Checkers
- Can be part of final netlist --> used in Post-Si
- Can be automatically generated to be used on the emulator

DFV Contd.


Synthesizable
Testbench Top
UVM Test
UVM ENV
UVM Agent
Scoreboard
Monitor
Sequencer
Driver
Sequences
Interface
DUT
Behavioral
Run on Emulator
Scoreboard
Transactor BFM
Instruction generator
Activity collector
Transactor
Interface
DUT
Synthesized assertions
Run on Host

DFV Contd.

- Checkers
  - Synthesized assertions
- Transactor
  - Constrained-based stimulus generation
- Activity Collector
  - Generates transaction-level, not cycle-based packets
  - Interact with host for coverage, scoreboarding, etc.



Summary

- SoCs are becoming more complex
  - Multi-die chiplet
- Pre-Si simulation cannot close the huge state-space gap
- Emulation can help if not slowed down
  - Interaction with the host
- Automated solutions to generate synthesizable DFV blocks
  - Instructions and masks for on-chip constrained random generators
  - Synthesized assertions
  - On-chip signal-level -> txn level convertors

Thank You !

Title Here

- Edit Master text styles
  - Second level
    - Third level
      - Fourth level
        - dfgdfFifth level