COSIDE® SYNOPSYS® BRIDGE

ADVANCING ARCHITECTURE EXPLORATION FOR RF-CMOS

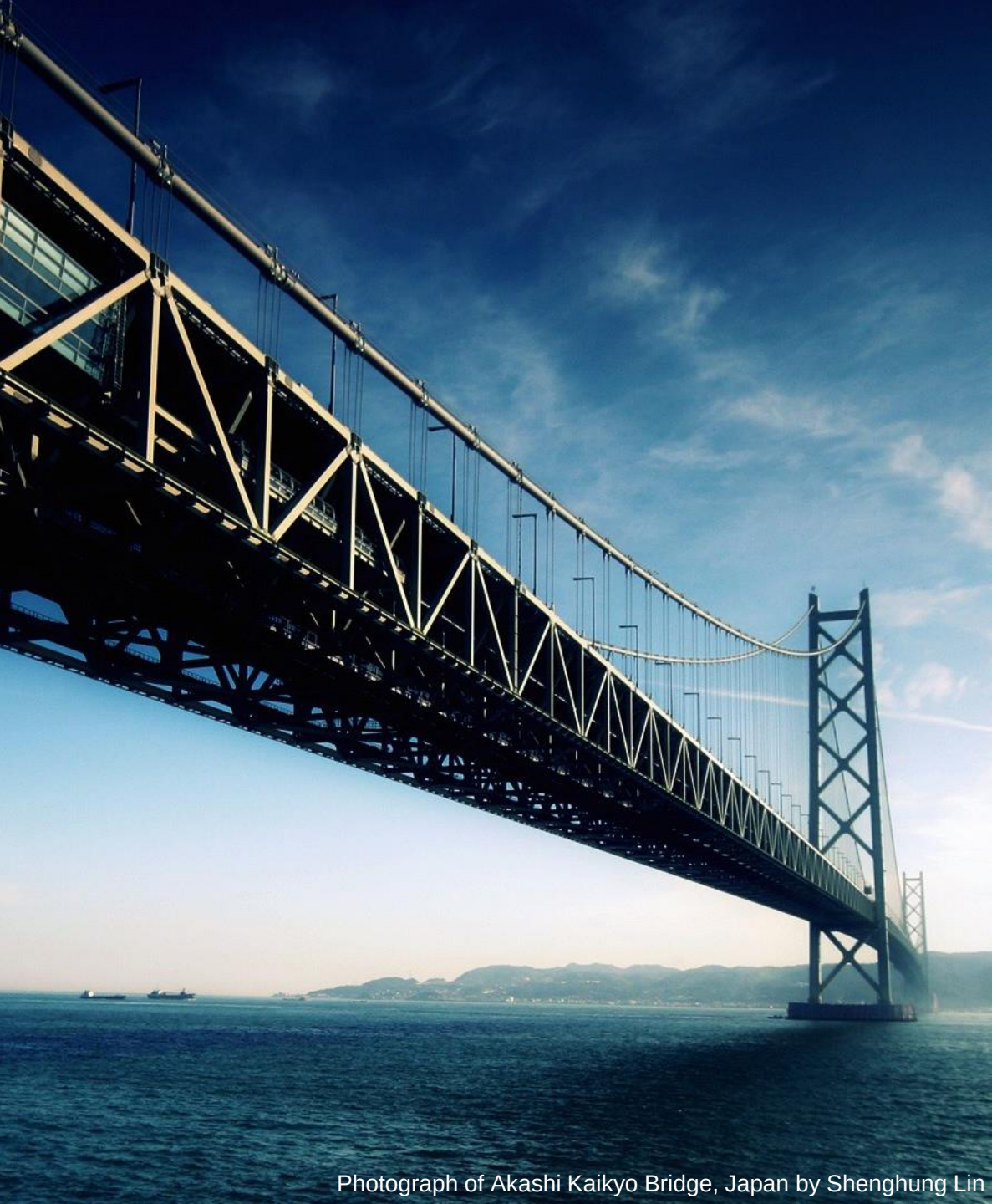
JAN KASTNING

SYSTEMC AMS & COSIDE® UGM 2017

19 OCTOBER 2017

NXP

SECURE CONNECTIONS FOR A SMARTER WORLD



Photograph of Akashi Kaikyo Bridge, Japan by Shenghung Lin

Agenda

1. System design process
2. Architecture exploration process
3. Why advance?
4. COSIDE® Synopsys ® Bridge
5. What‘s been done so far?
6. Simple proof of concept
7. What‘s next?
8. Accomplishments

System design process

- Top-down design structure
  - System decomposition
  - Verify implementation and integration
  - Validate system and product
- Achieve cost-efficient control over system design process
  - System optimization
  - Early identification of technical challenges

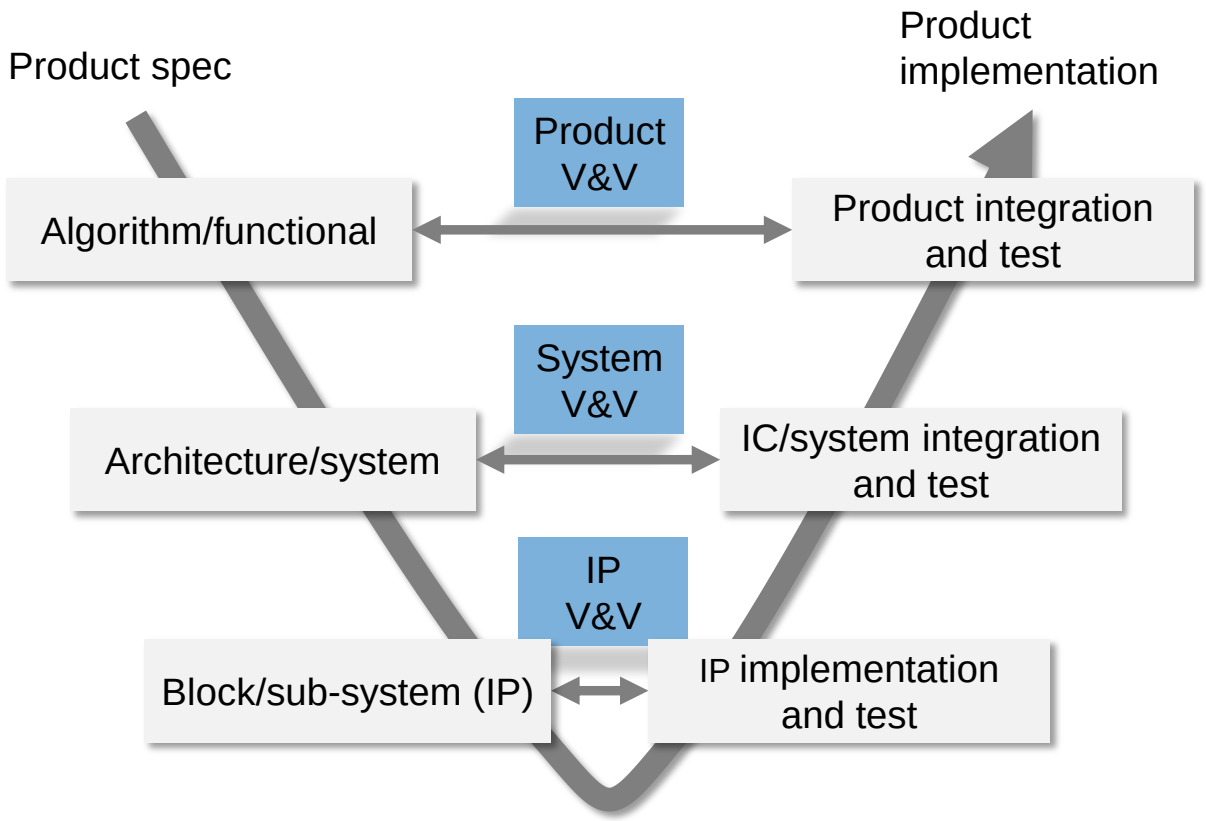
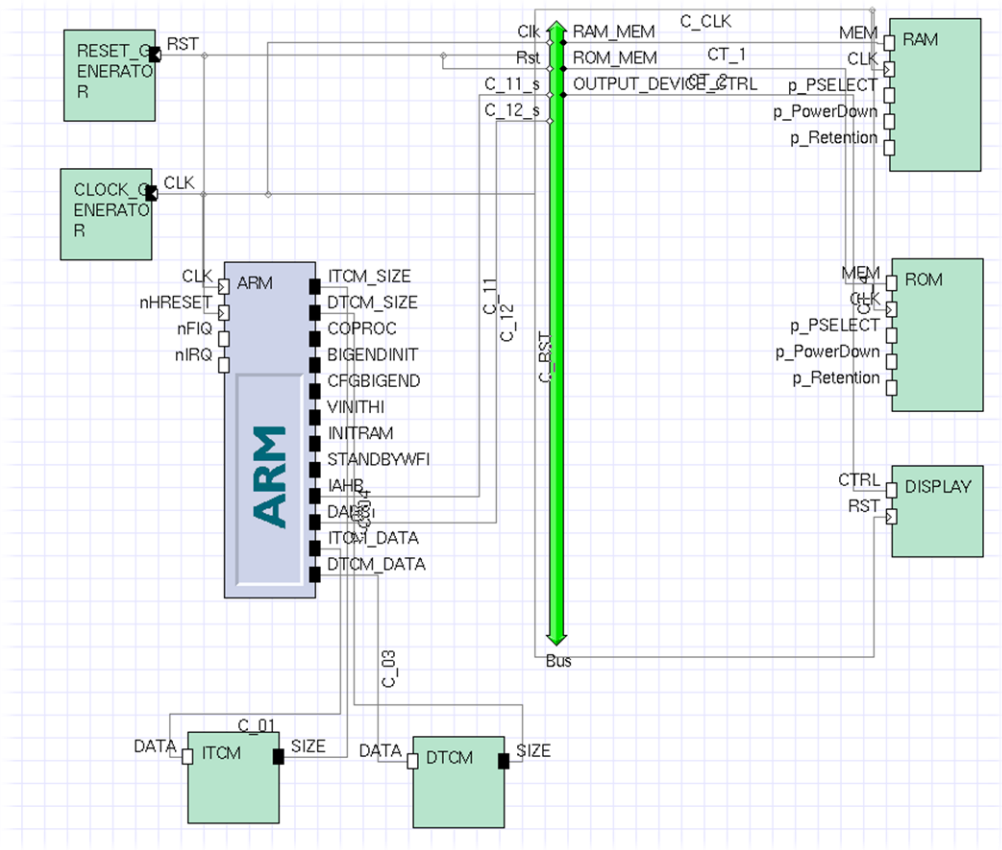


Illustration by Erwin de Kock, Martin Barnasconi

Architecture exploration process

Methodology



- Approach for system decomposition
- Estimate and optimize system resources
- Software: Platform Architect

Architecture exploration process

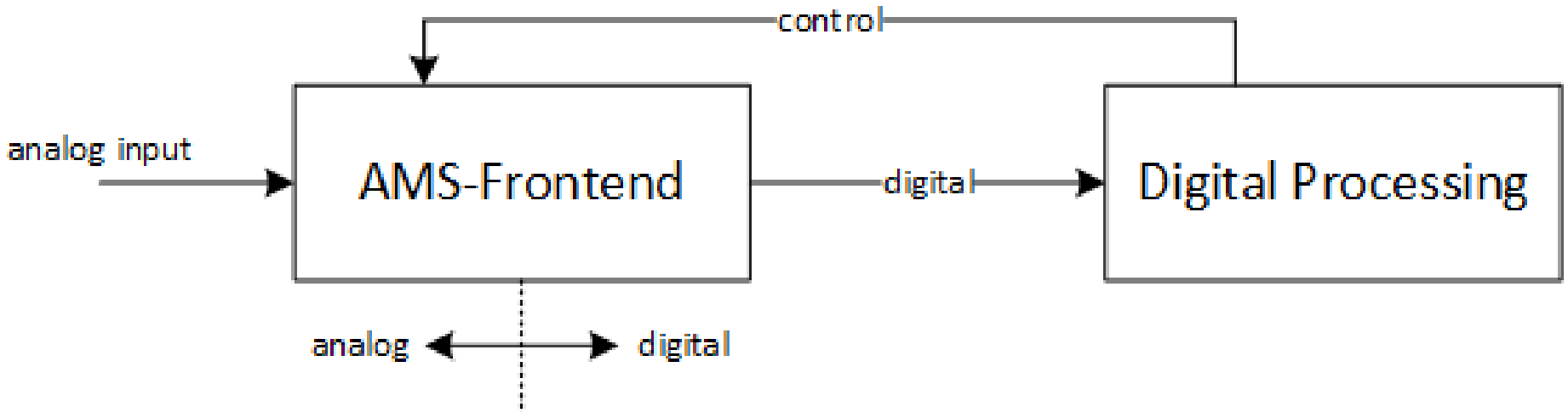
Model approaches

	High level	Low level
Design process	Architecture design / exploration	Block / sub-system design / exploration
Model base	Virtual processing / memory units → **Virtual platform representation**	IP models → **Accurate platform representation**
Simulation base	Task graph based on: • Resources (time & memory access) • Task dependency → **Basic behaviour**	• Input data • Application images (source code) → **Functional behaviour**
Advantages	• Fast & early architecture exploration • Sufficient for major design decisions	• Cycle accurate decisions and task dependecies • Allows fine tuning and design optimization
Outcome	**Virtual platform which represents basic system behaviour**	**Highly accurate platform and functional behaviour representation**
Use-case	Design decisions & system partitioning	System verification & optimization

Why advance?

Motivation

- Controlling AMS-IP behavior
  - Affects digital processing behavior
- Combine AMS-Frontend & digital processing



Why advance?

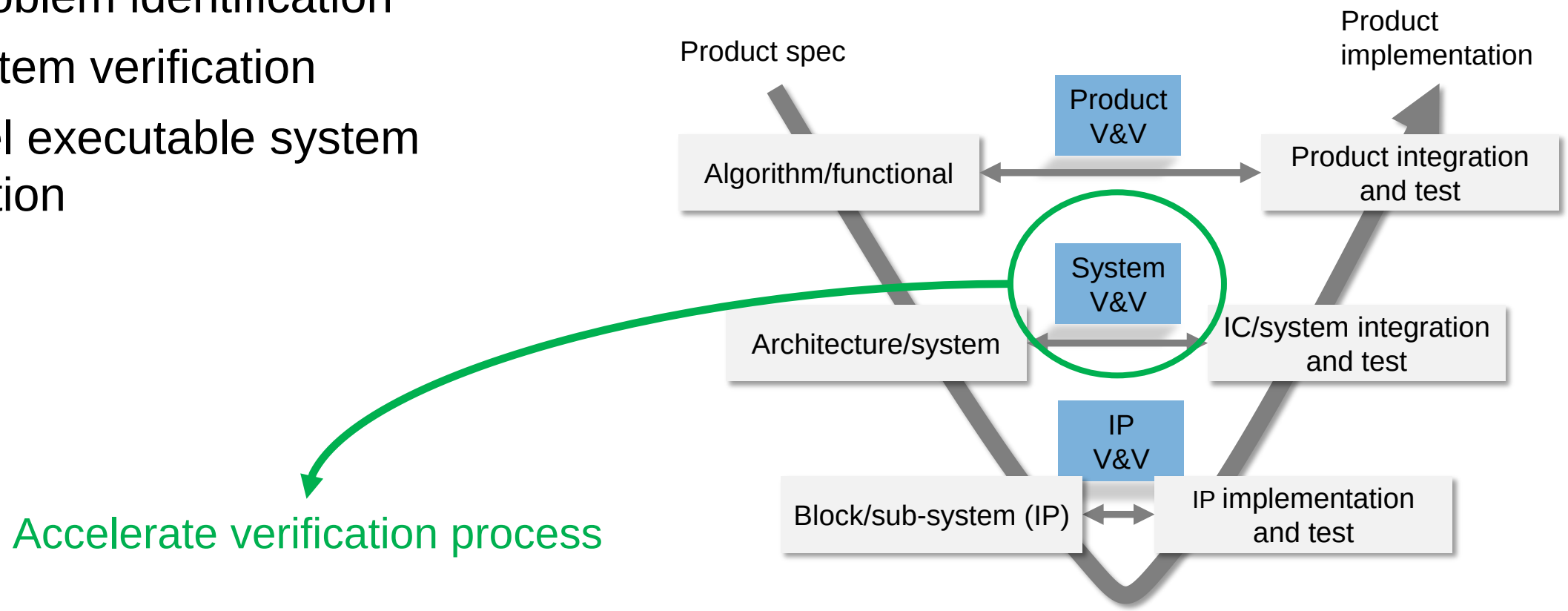
System complexity needs

- IP level
  - Input dependency and history
- System level
  - IP interconnections and relations
- AMS subsystems highly complex
  - Constraining or specify

Why advance?

Expectations

- Better problem identification
- Early system verification
- High level executable system specification



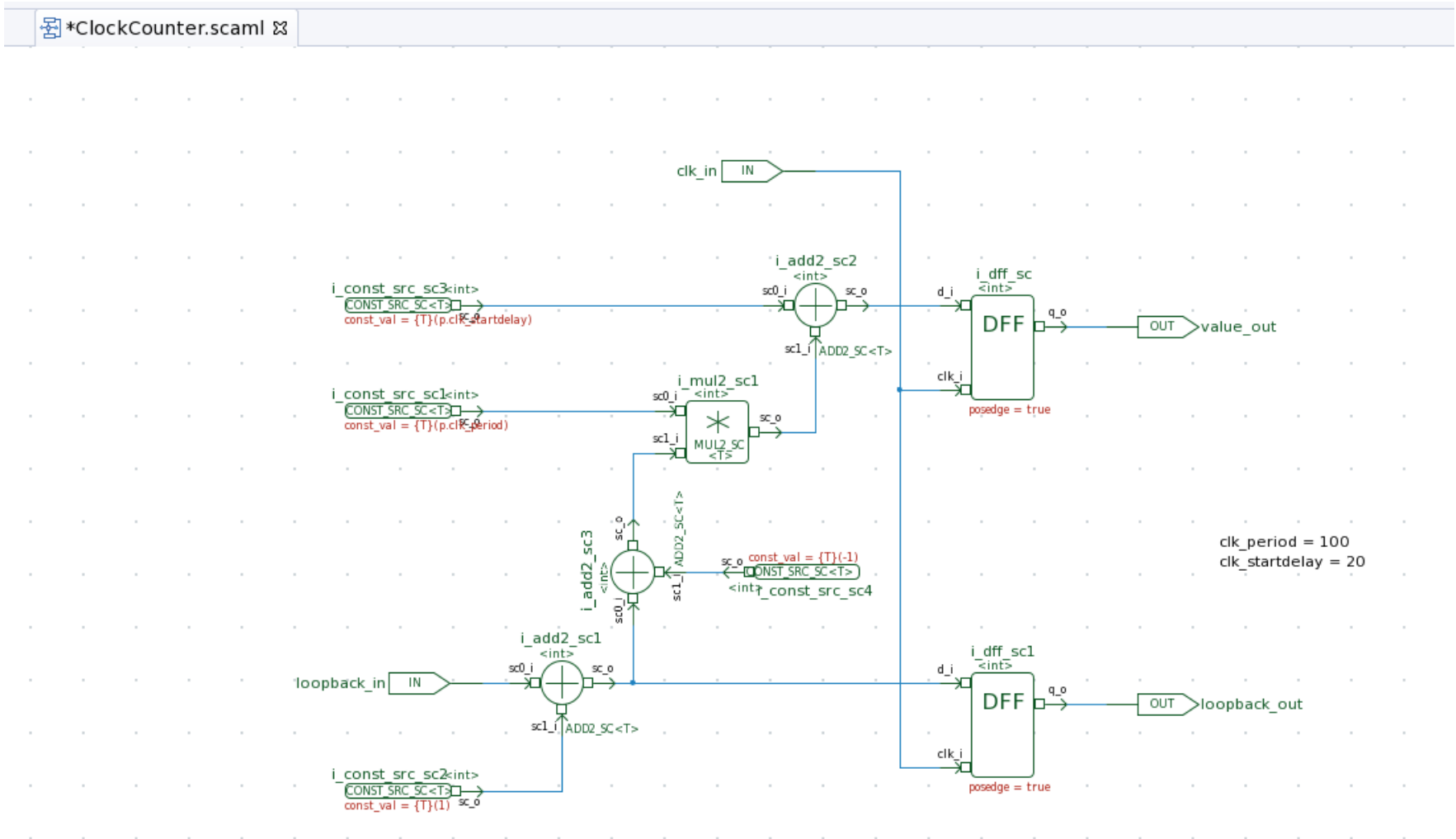
COSIDE® Synopsys® Bridge

- Import any AMS model from COSIDE ®
  - AMS model as standalone subsystem
  - Analog input controlled and defined in AMS model
    - Signal generator from COSIDE® library
    - Stimuli file import
- No tool configuration needed
  - Simple model import into Platform Architect
  - Basic SystemC port type support
- Full COSIDE® library support

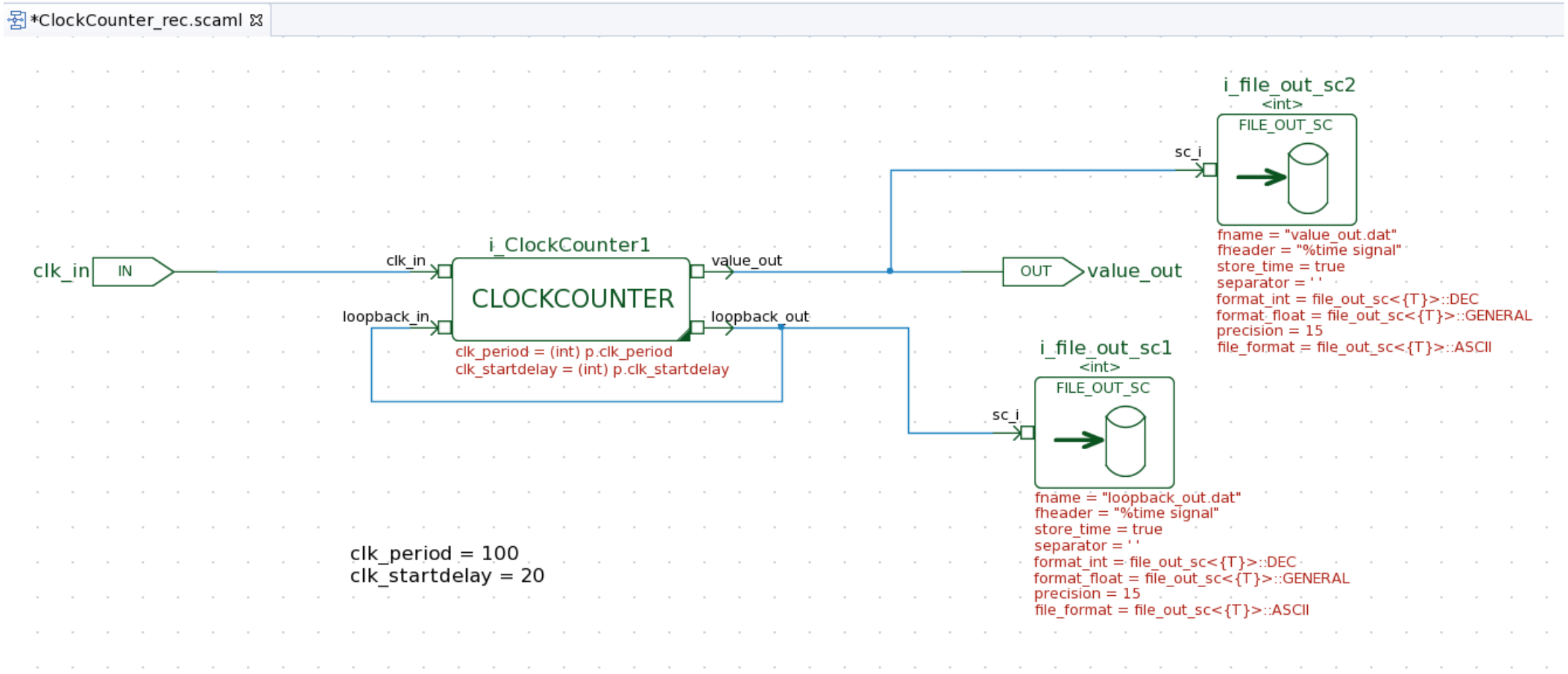
What‘s been done so far?

- Analysing need of dynamic tool connection
- Determining use-cases
- Brainstorming possible solutions
  - Getting Coseda & Synopsys onboard
- Starting tool development
- Testing and improving
- Simple proof of concept

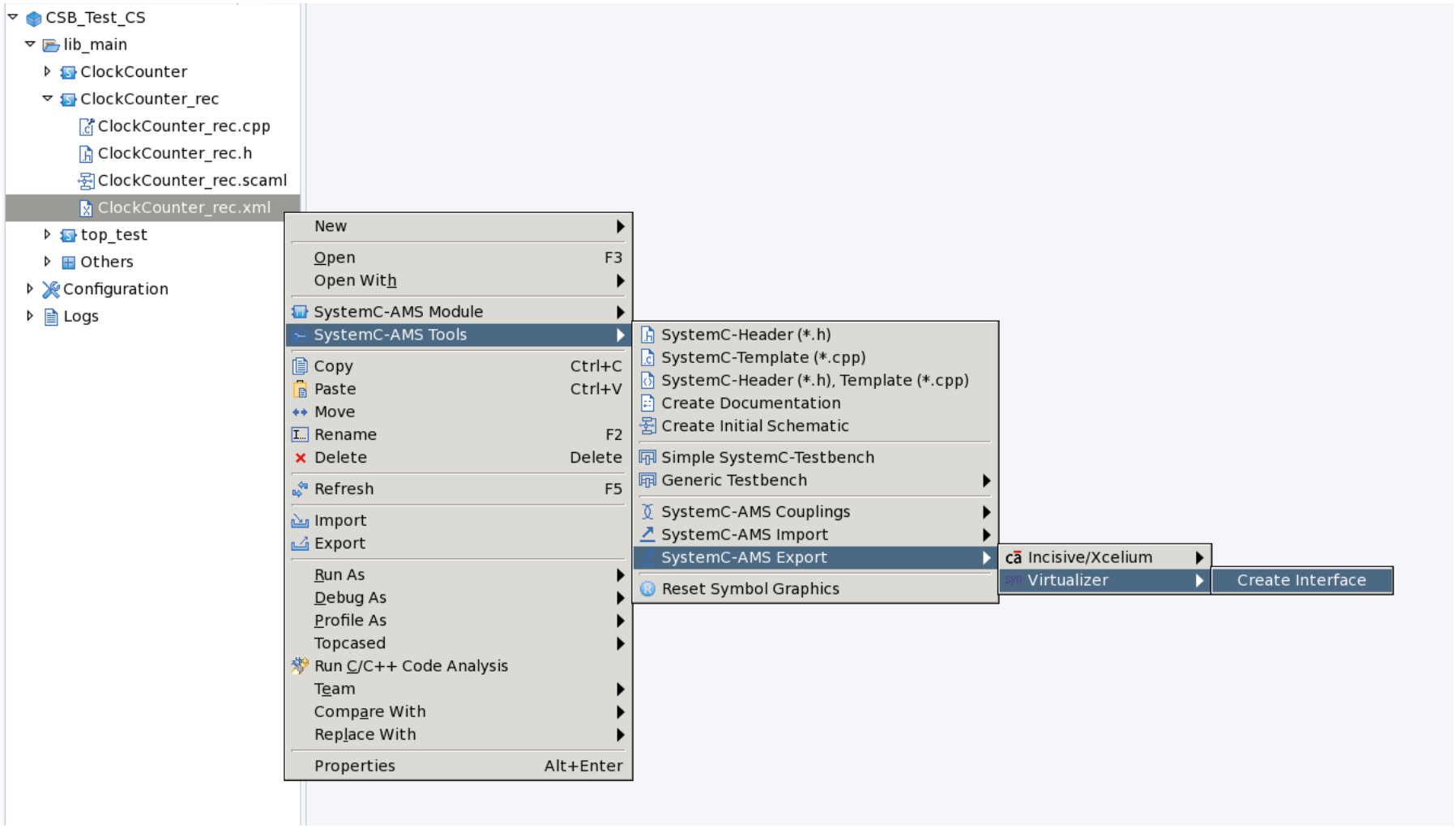
Simple proof of concept



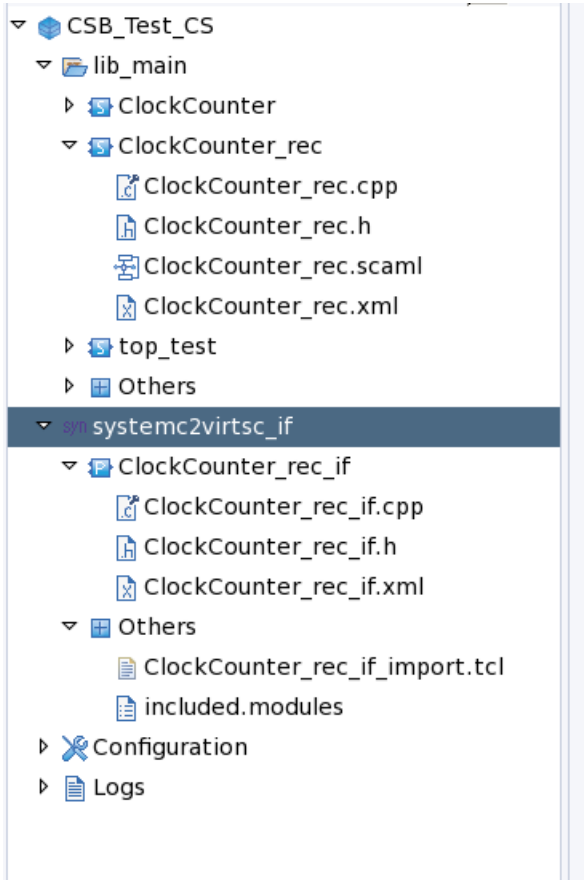
Simple proof of concept

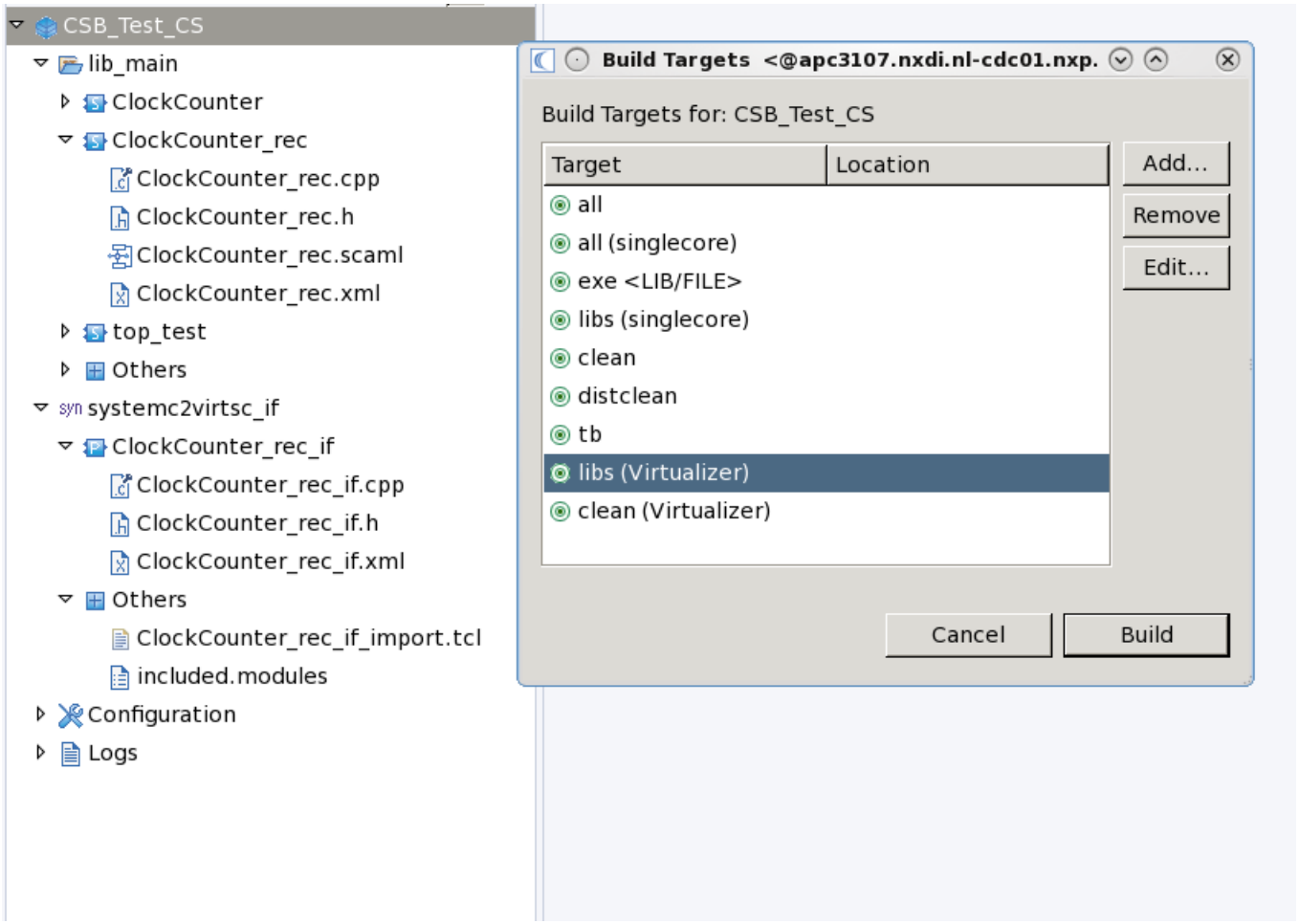


Simple proof of concept

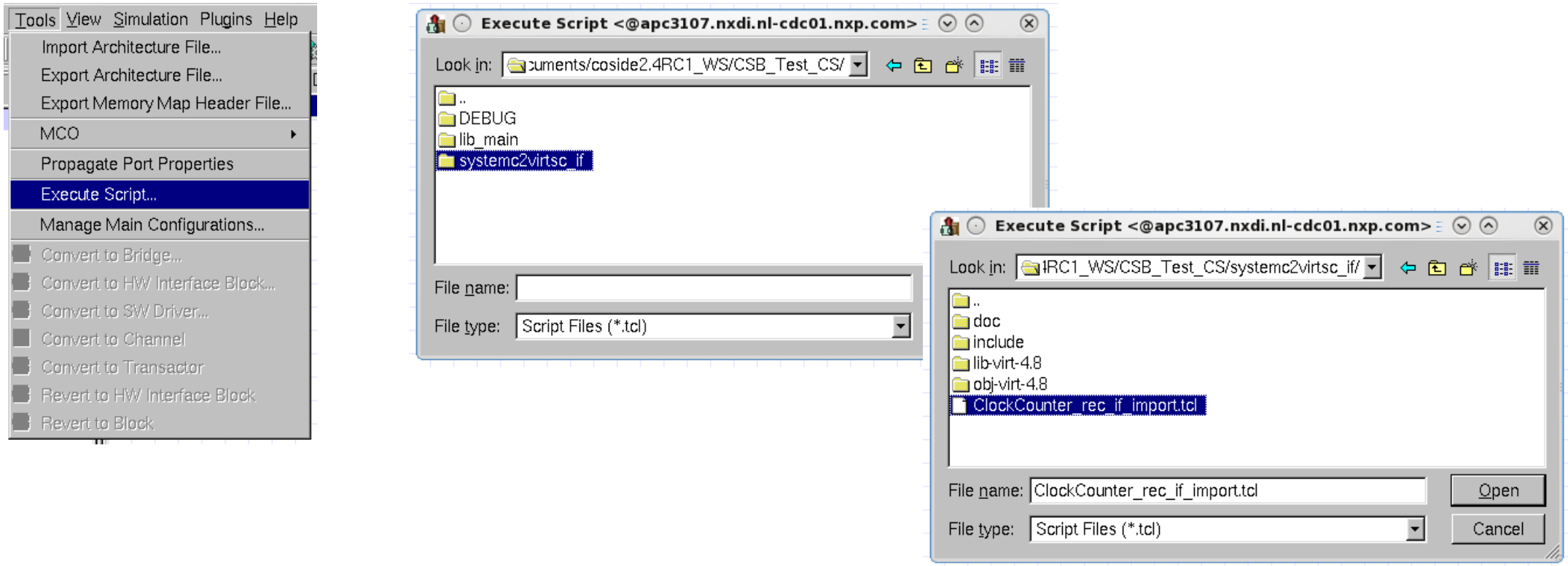


Simple proof of concept

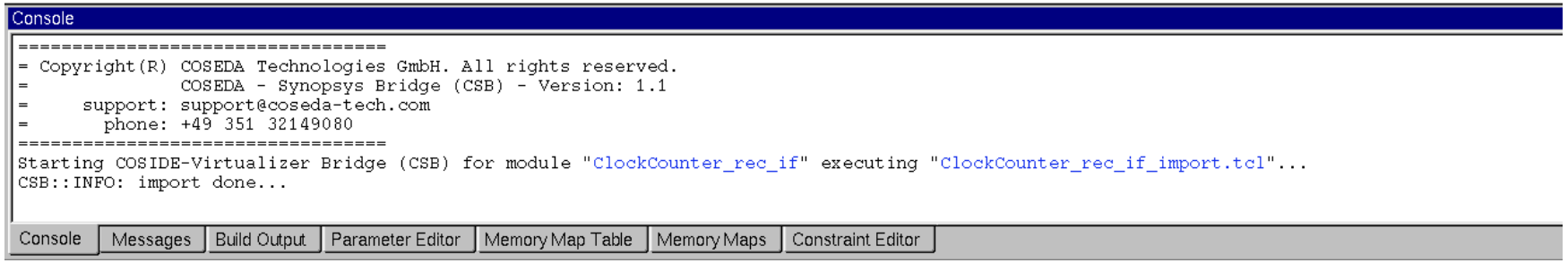


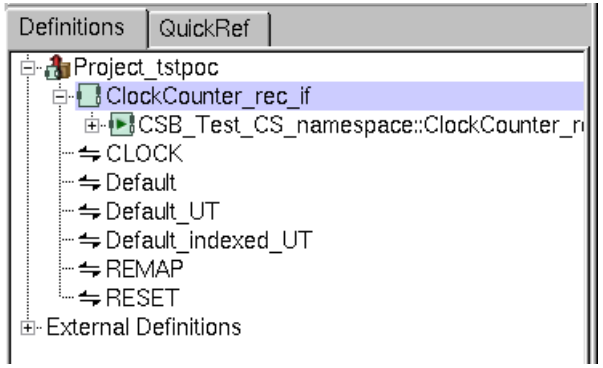


Simple proof of concept

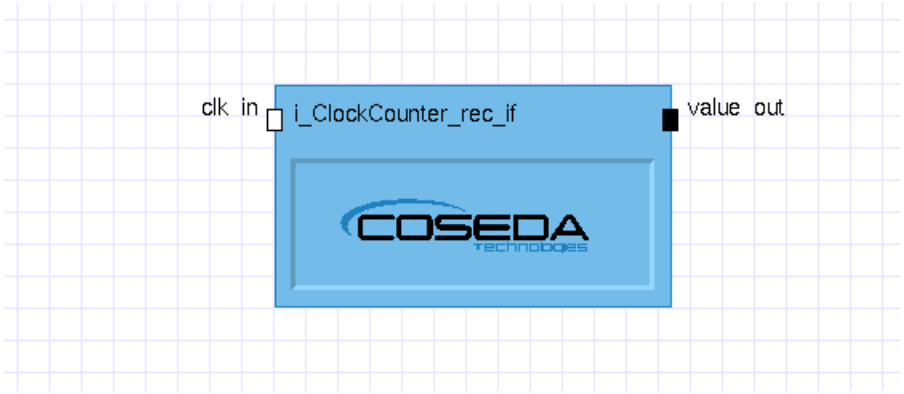


Simple proof of concept





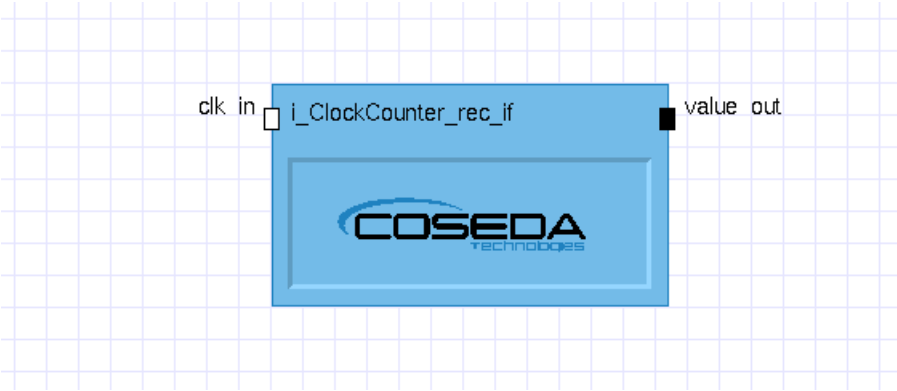
Simple proof of concept



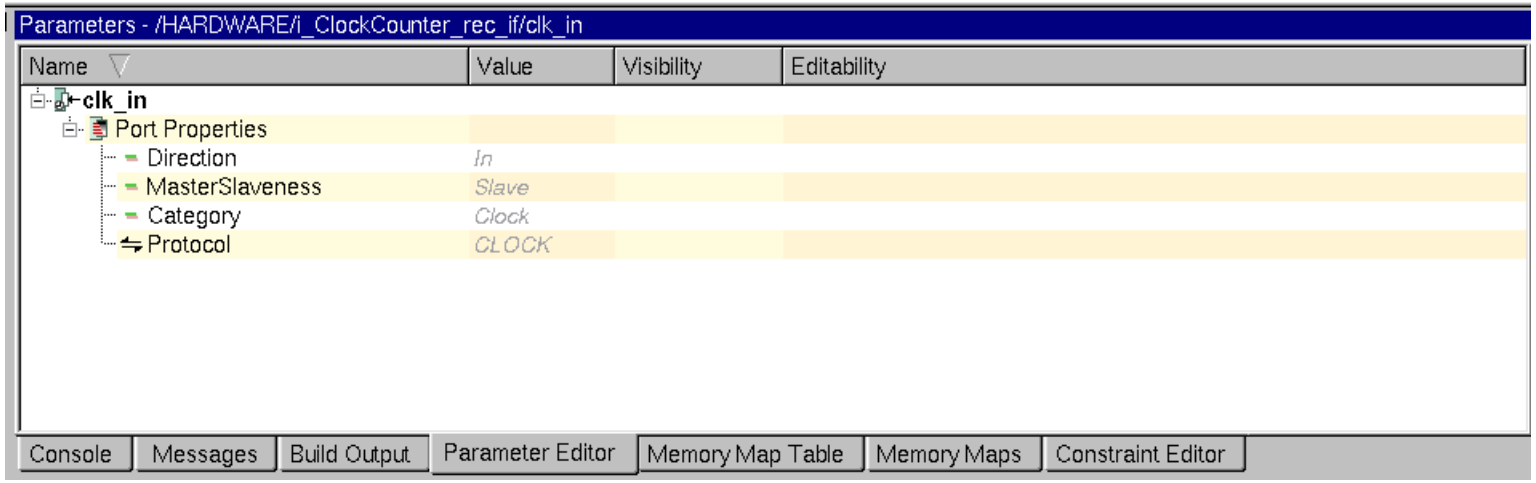
Name	Value	Configuration	Visibility	Editability
Block properties				
Name	i_ClockCounter_rec_if			
Scml Properties				
clk_period	100	Default	Visible	Until Simulation Start
clk_startdelay	20	Default	Visible	Until Simulation Start

Console | Messages | Build Output | Parameter Editor | Memory Map Table | Memory Maps | Constraint Editor

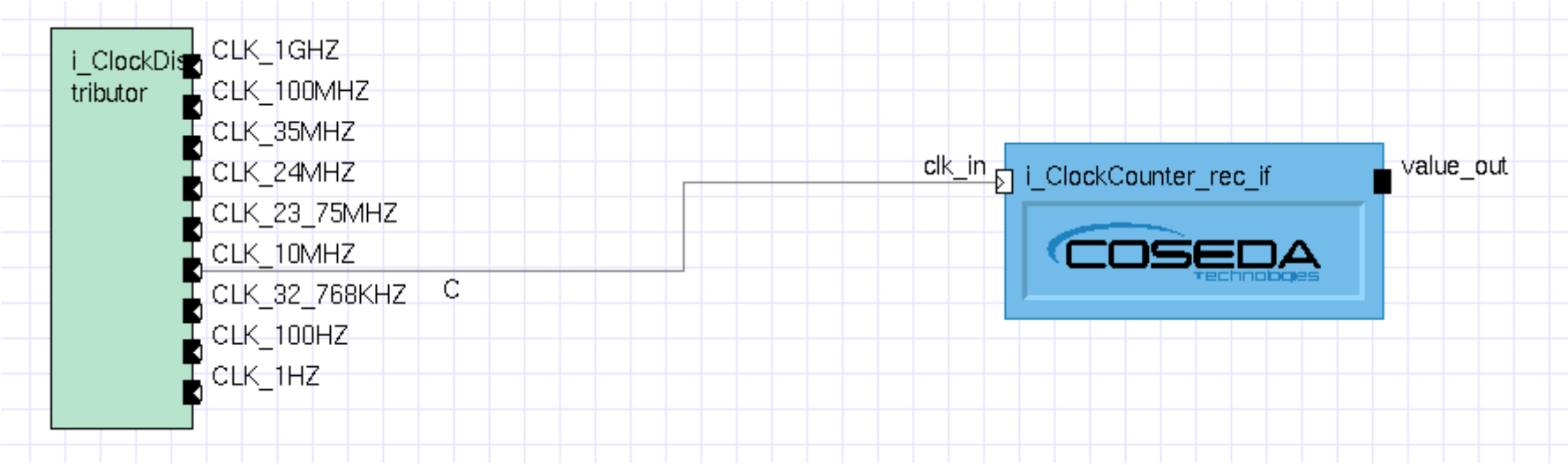
Simple proof of concept



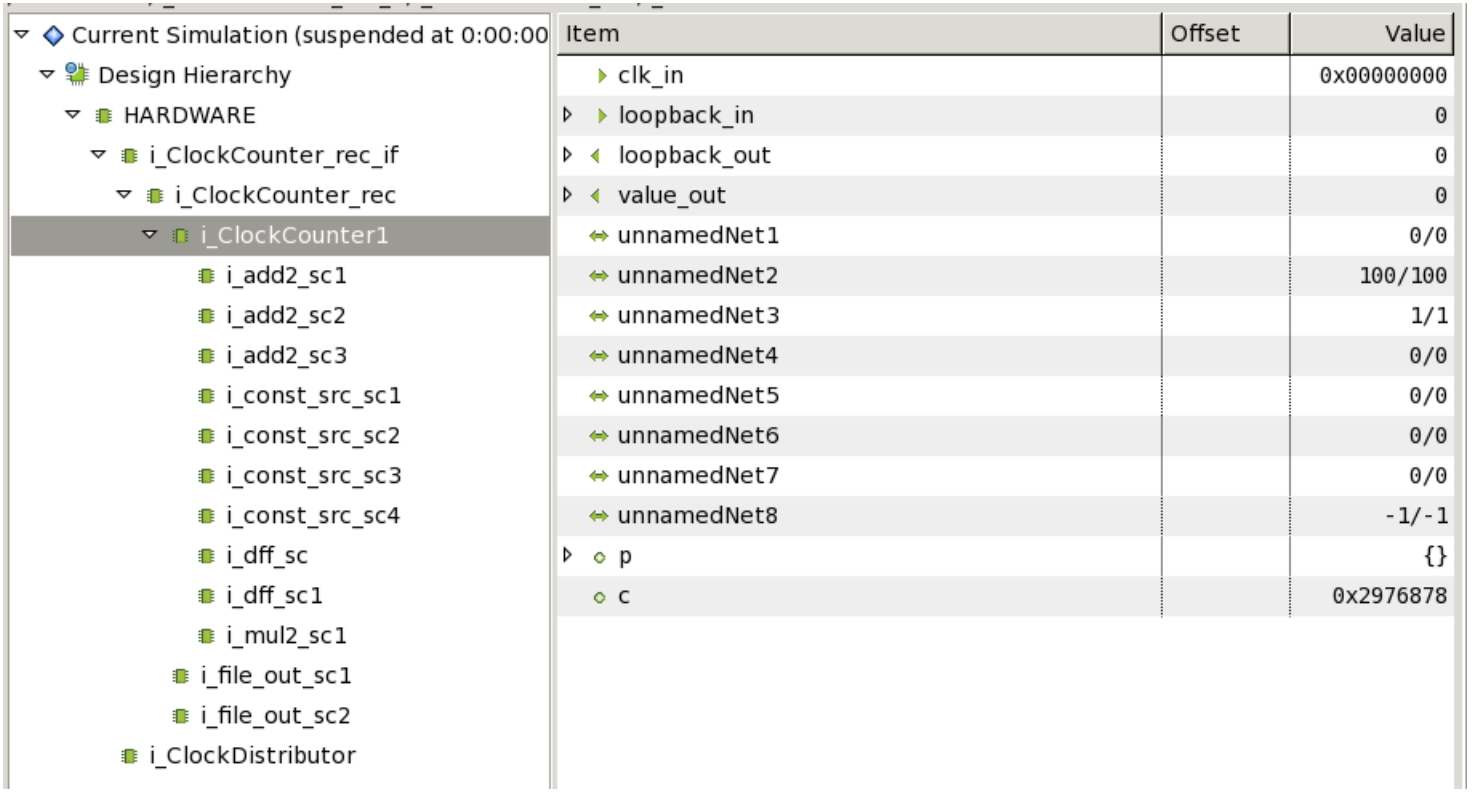
pct.protocol = CLOCK



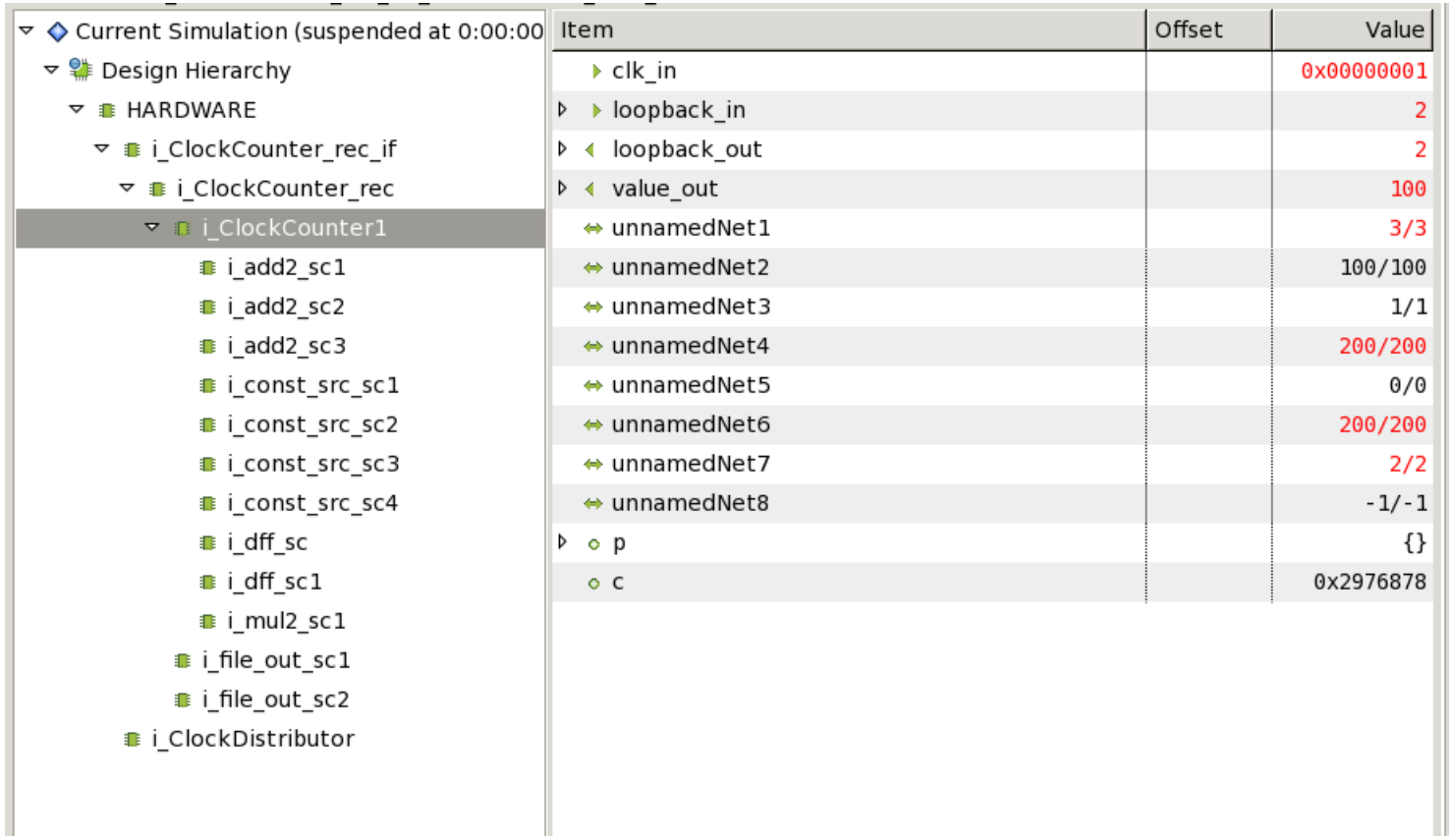
Simple proof of concept



Simple proof of concept



Simple proof of concept



What‘s next?

- Full-blown proof-of-concept
  - AMS functionality
  - Coside library building blocks
- TLM register support

Accomplishments

Thomas Arndt

Martin Barnasconi

Christian Schuster

Karsten Einwich

Martin Klein

Stefan Thiel

SECURE CONNECTIONS
FOR A SMARTER WORLD