

COSIDE® SYNOPSYS® BRIDGE

JAN KASTNING
SYSTEMC AMS & COSIDE® UGM 2017
19 OCTOBER 2017

ADVANCING
ARCHITECTURE
EXPLORATION
FOR RF-CMOS



PUBLIC



SECURE CONNECTIONS
FOR A SMARTER WORLD



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

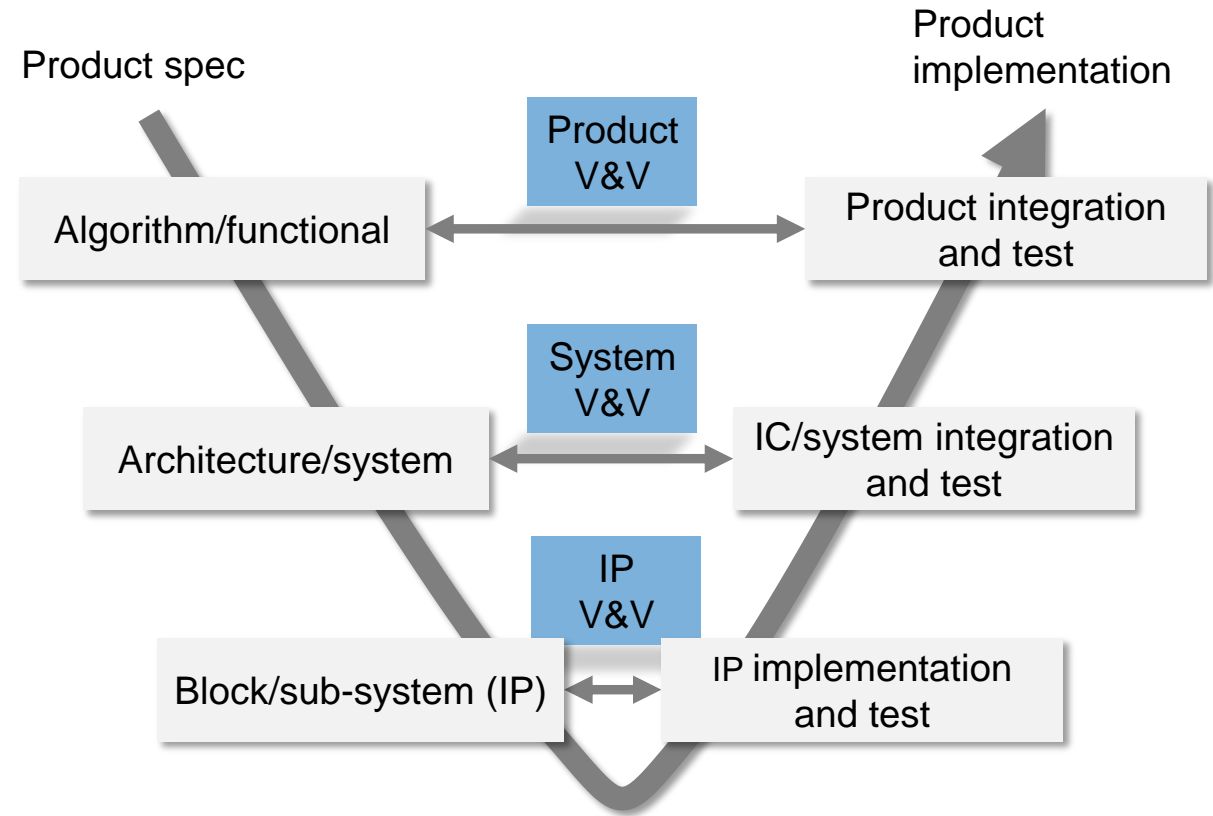
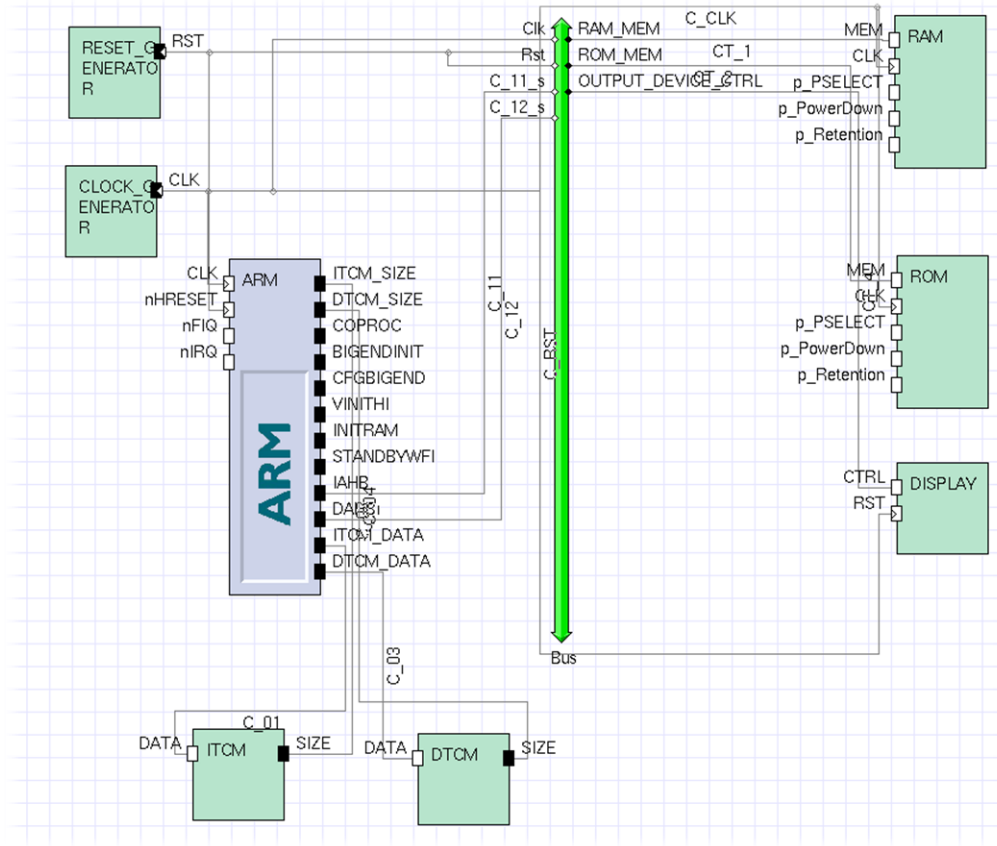


Illustration by Erwin de Kock, Martin Barnasconi

Architecture exploration process

Methodology



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

Architecture exploration process

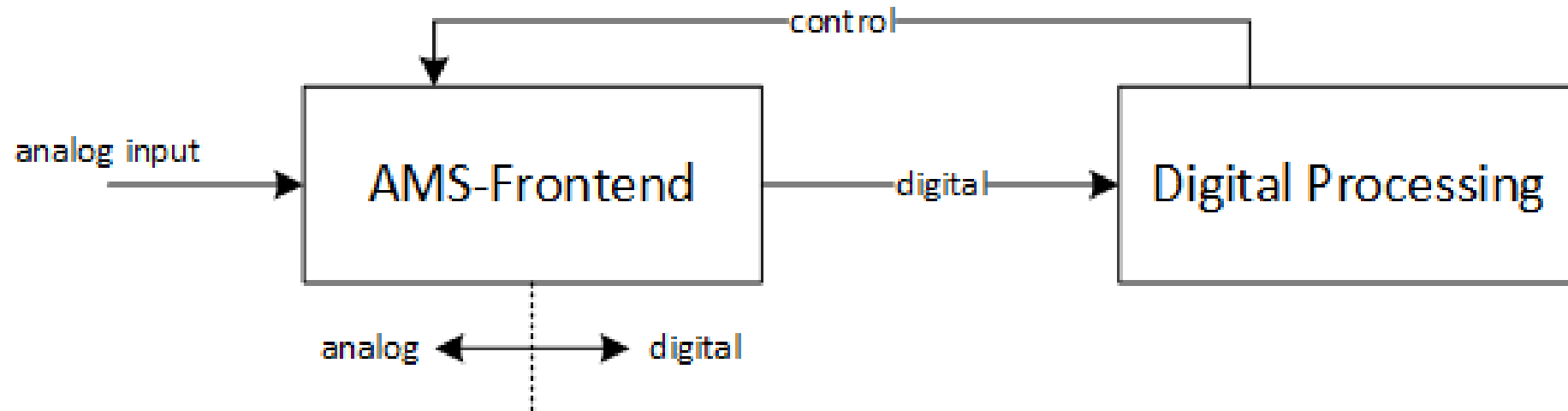
Model approaches

	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: <ul style="list-style-type: none">• Resources (time & memory access)• Task dependency → Basic behaviour	<ul style="list-style-type: none">• Input data• Application images (source code) → Functional behaviour
Advantages	<ul style="list-style-type: none">• Fast & early architecture exploration• Sufficient for major design decisions	<ul style="list-style-type: none">• Cycle accurate decisions and task dependencies• 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?

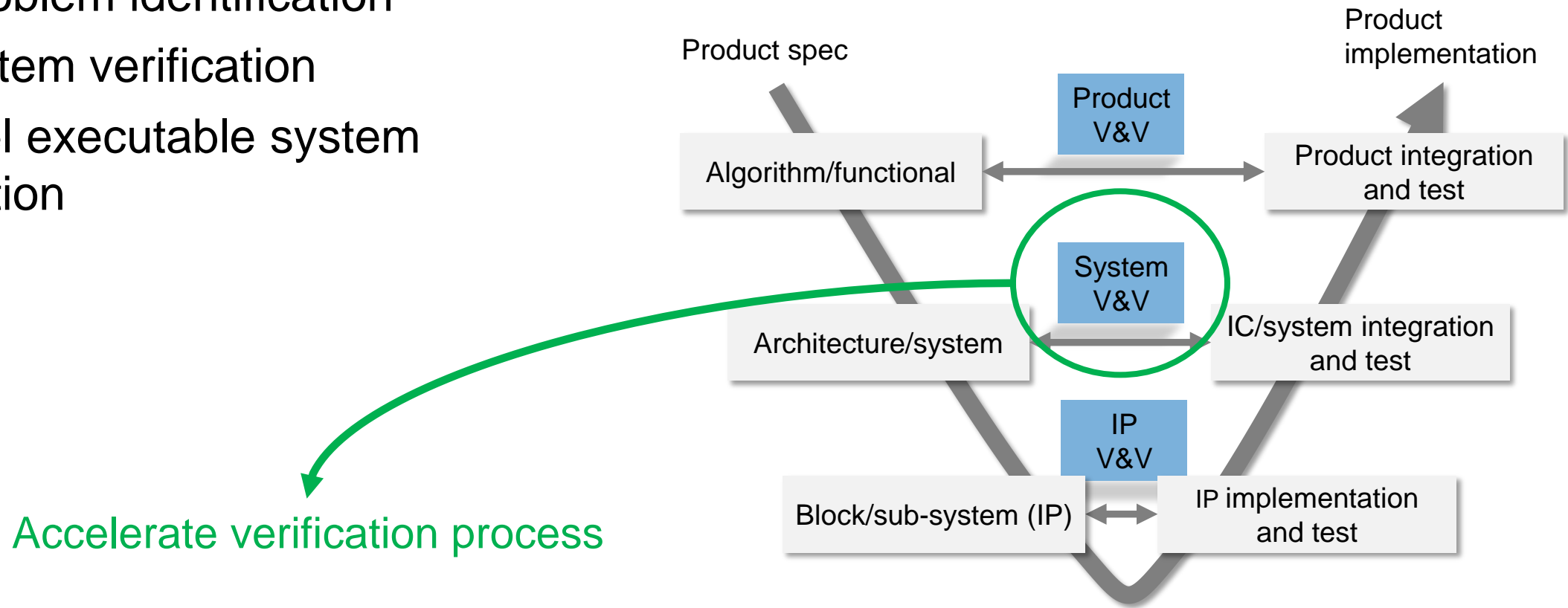
System complexity needs

- 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



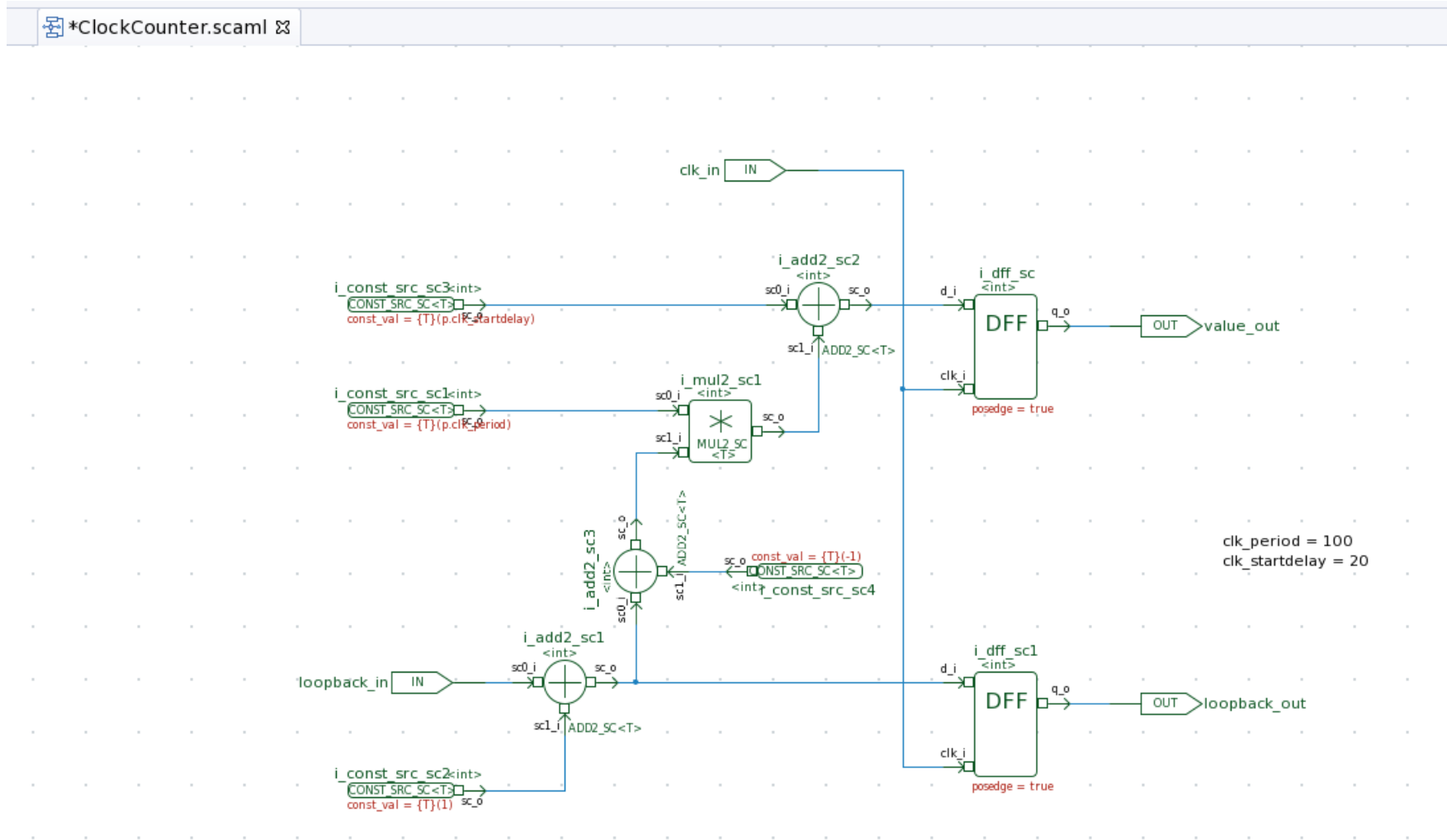
COSIDE[®] Synopsys[®] Bridge

- 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?

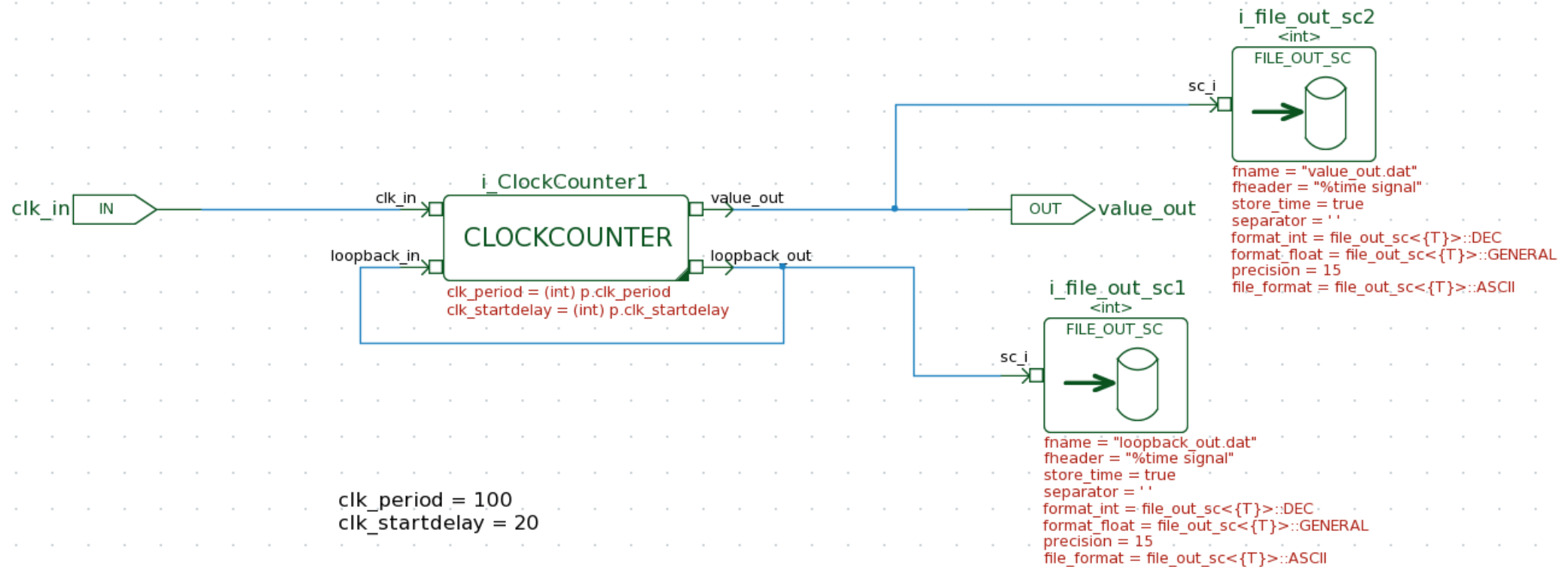
- Analysing need of dynamic tool connection
- Determining use-cases
- Brainstorming possible solutions
 - Getting Coseda & Synopsys onboard
- Starting tool development
- Testing and improving
- Simple proof of concept

Simple proof of concept

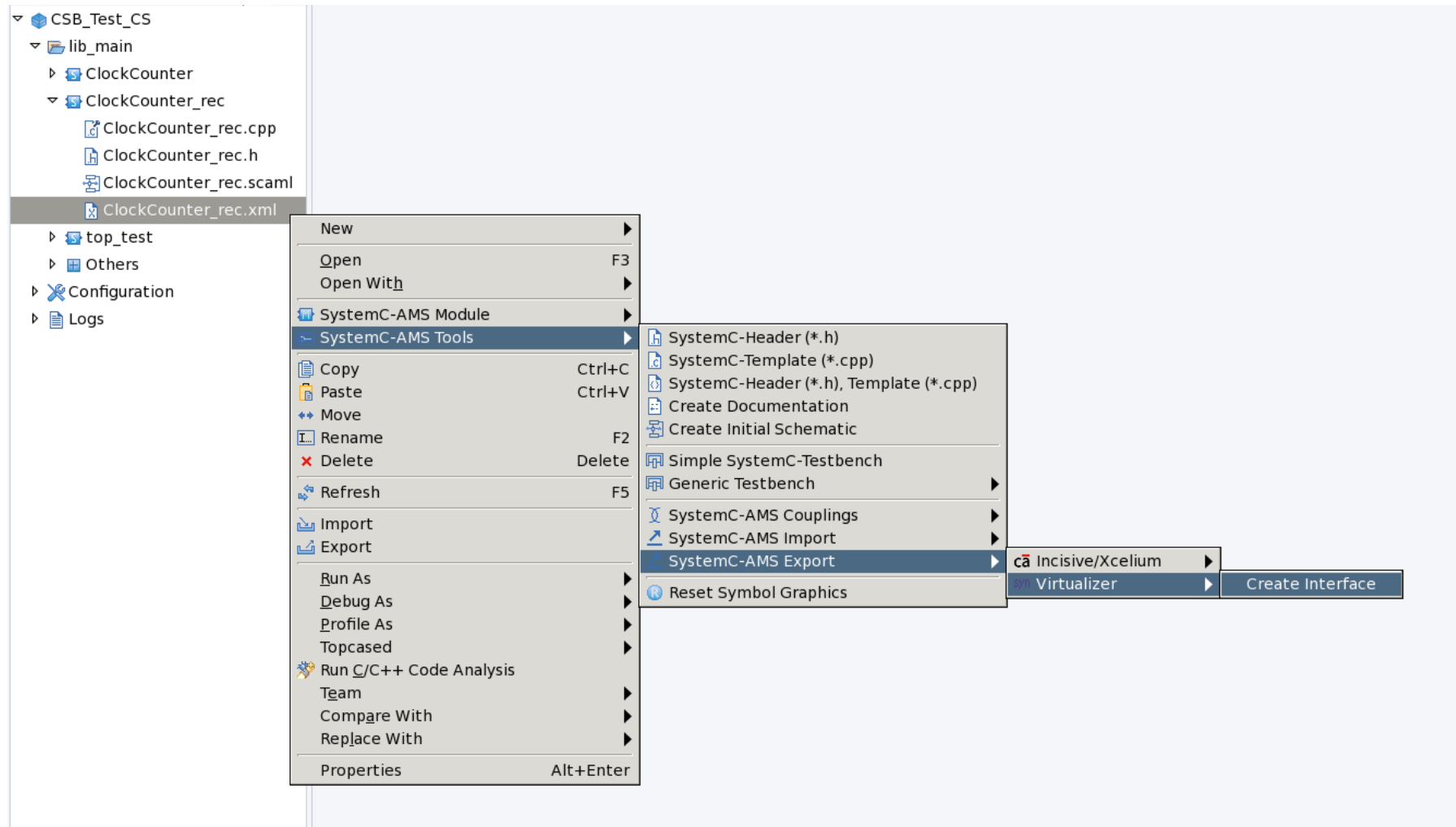


Simple proof of concept

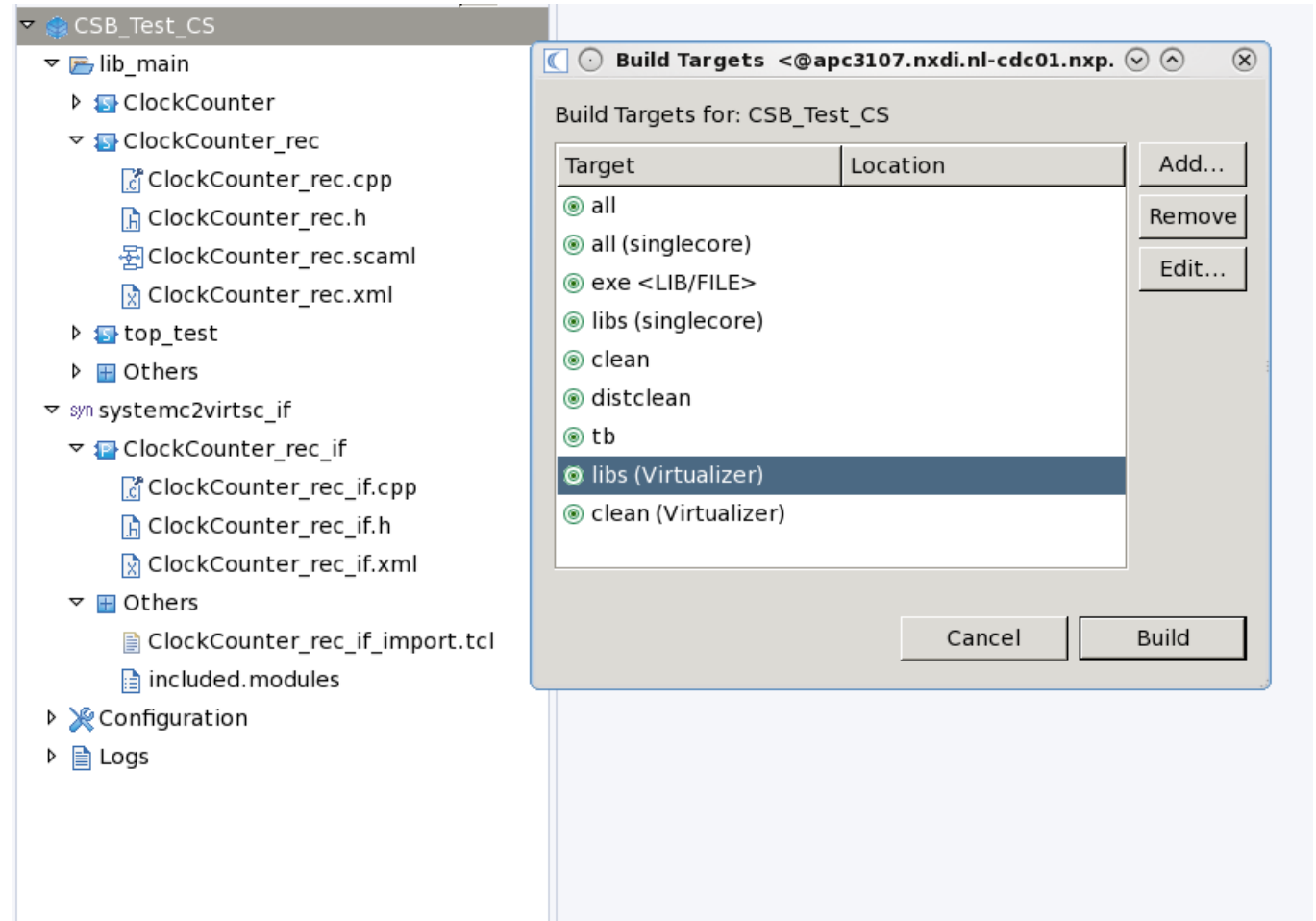
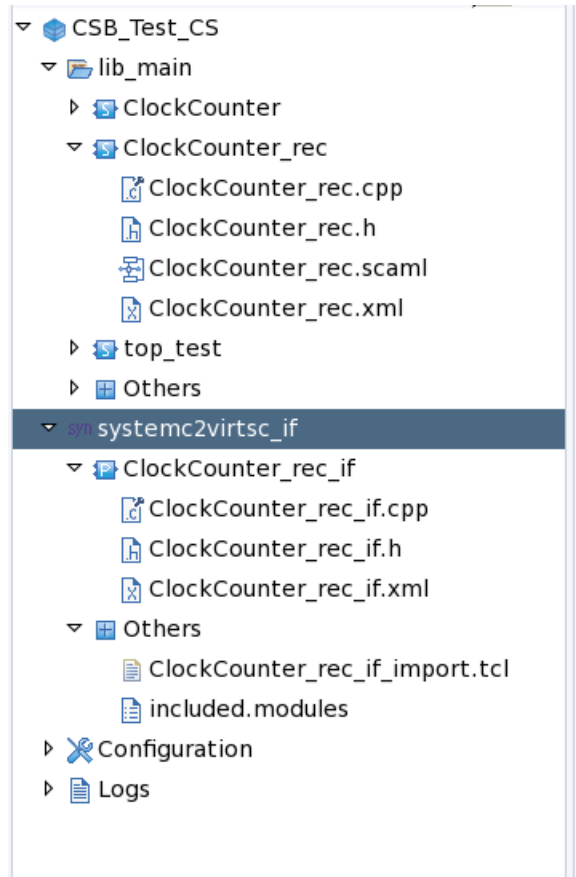
*ClockCounter_rec.scaml 88



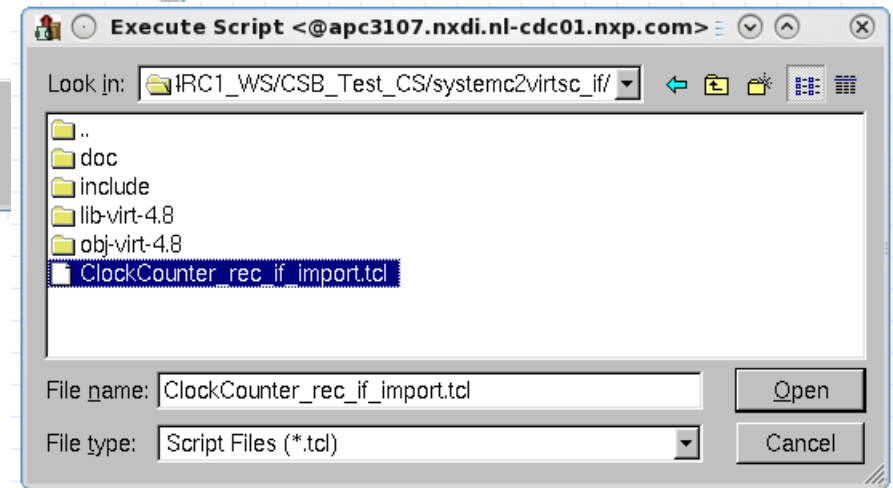
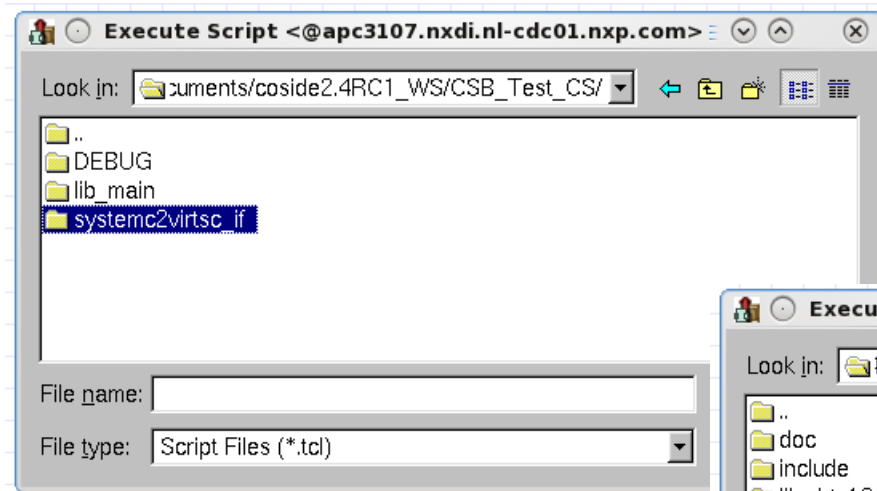
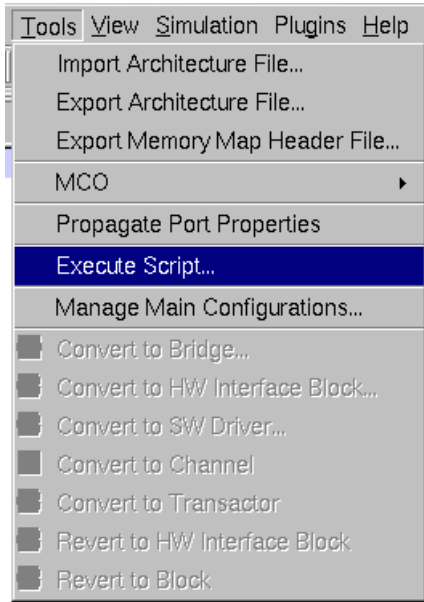
Simple proof of concept



Simple proof of concept



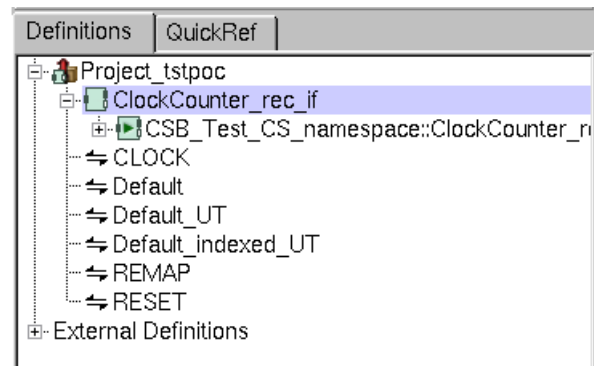
Simple proof of concept



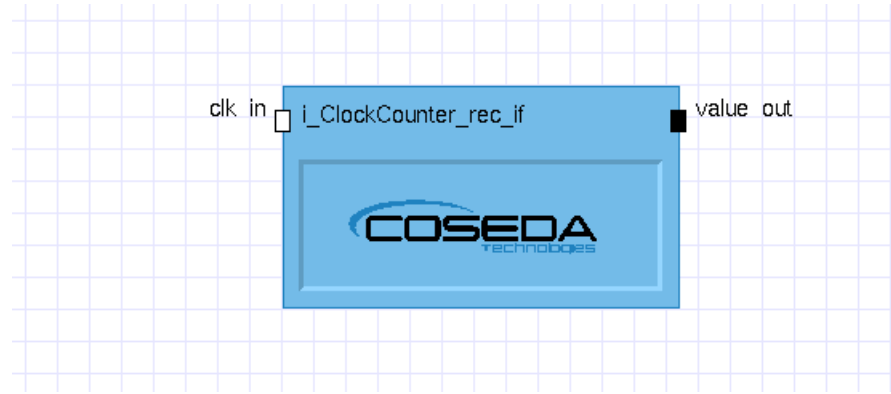
Simple proof of concept

```
Console
=====
= Copyright(R) COSEDA Technologies GmbH. All rights reserved.
= COSEDA - Synopsys Bridge (CSB) - Version: 1.1
= support: support@coseda-tech.com
= phone: +49 351 32149080
=====
Starting COSIDE-Virtualizer Bridge (CSB) for module "ClockCounter_rec_if" executing "ClockCounter_rec_if_import.tcl"...
CSB::INFO: import done...
```

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



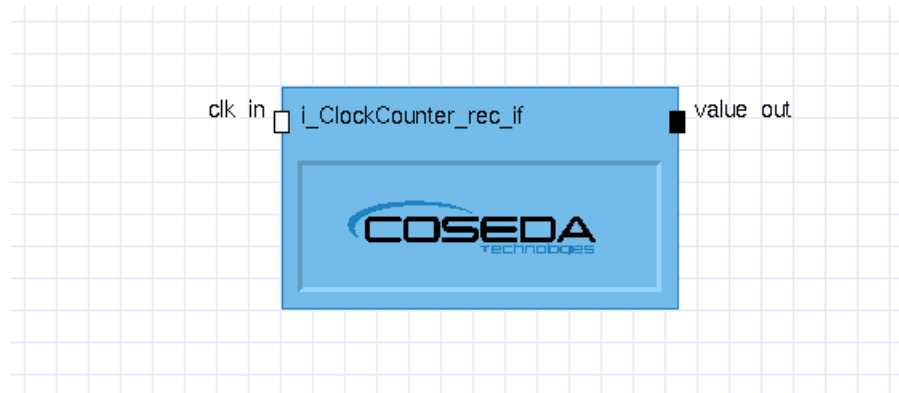
Simple proof of concept



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

ConsoleMessagesBuild OutputParameter EditorMemory Map TableMemory MapsConstraint Editor

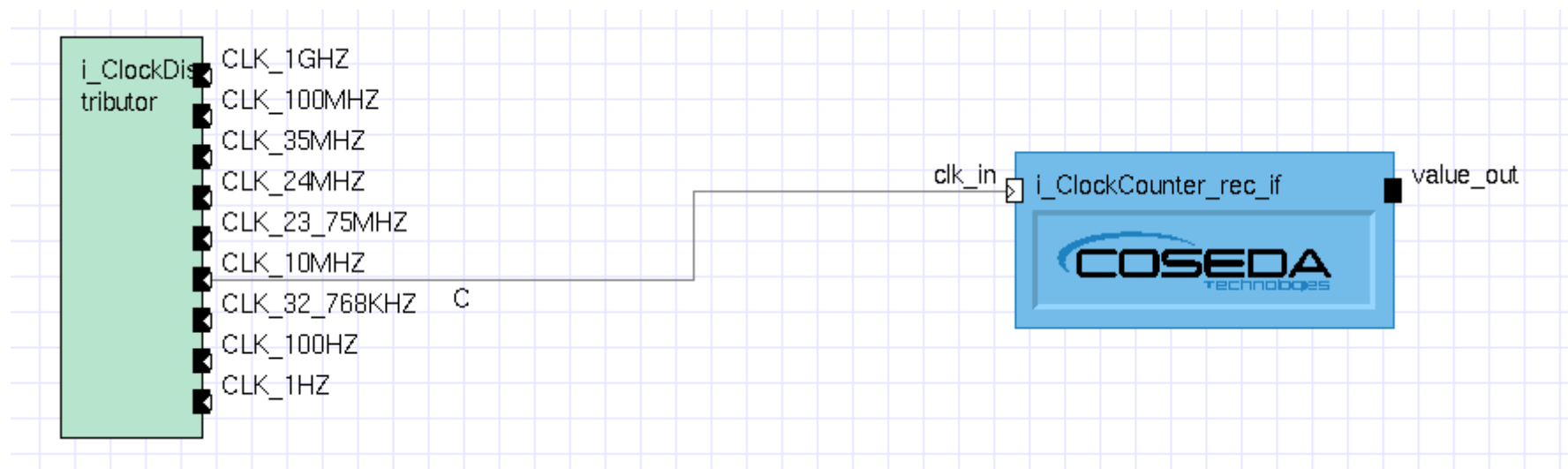
Simple proof of concept



pct.protocol = CLOCK

Parameters - /HARDWARE/i_ClockCounter_rec_if/clk_in			
Name	Value	Visibility	Editability
clk_in			
Port Properties			
Direction	In		
MasterSlaveness	Slave		
Category	Clock		
Protocol	CLOCK		

Simple proof of concept



Simple proof of concept

Current Simulation (suspended at 0:00:00)			
Design Hierarchy			
HARDWARE			
i_ClockCounter_rec_if			
i_ClockCounter_rec			
i_ClockCounter1			
i_add2_sc1			
i_add2_sc2			
i_add2_sc3			
i_const_src_sc1			
i_const_src_sc2			
i_const_src_sc3			
i_const_src_sc4			
i_dff_sc			
i_dff_sc1			
i_mul2_sc1			
i_file_out_sc1			
i_file_out_sc2			
i_ClockDistributor			
Item	Offset	Value	
clk_in		0x00000000	
loopback_in		0	
loopback_out		0	
value_out		0	
unnamedNet1		0/0	
unnamedNet2		100/100	
unnamedNet3		1/1	
unnamedNet4		0/0	
unnamedNet5		0/0	
unnamedNet6		0/0	
unnamedNet7		0/0	
unnamedNet8		-1/-1	
p		{}	
c		0x2976878	

Simple proof of concept

<div>Current Simulation (suspended at 0:00:00)</div> <div>Design Hierarchy</div> <div>HARDWARE</div> <div>i_ClockCounter_rec_if</div> <div>i_ClockCounter_rec</div> <div>i_ClockCounter1</div> <div>i_add2_sc1</div> <div>i_add2_sc2</div> <div>i_add2_sc3</div> <div>i_const_src_sc1</div> <div>i_const_src_sc2</div> <div>i_const_src_sc3</div> <div>i_const_src_sc4</div> <div>i_dff_sc</div> <div>i_dff_sc1</div> <div>i_mul2_sc1</div> <div>i_file_out_sc1</div> <div>i_file_out_sc2</div> <div>i_ClockDistributor</div>	<div>Item</div> <div>clk_in</div> <div>loopback_in</div> <div>loopback_out</div> <div>value_out</div> <div>unnamedNet1</div> <div>unnamedNet2</div> <div>unnamedNet3</div> <div>unnamedNet4</div> <div>unnamedNet5</div> <div>unnamedNet6</div> <div>unnamedNet7</div> <div>unnamedNet8</div> <div>p</div> <div>c</div>	<div>Offset</div>	<div>Value</div> <div>0x00000001</div> <div>2</div> <div>2</div> <div>100</div> <div>3/3</div> <div>100/100</div> <div>1/1</div> <div>200/200</div> <div>0/0</div> <div>200/200</div> <div>2/2</div> <div>-1/-1</div> <div>{}</div> <div>0x2976878</div>

What's next?

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

Accomplishments



Thomas Arndt

Karsten Einwich



Martin Barnasconi

Martin Klein



Christian Schuster

Stefan Thiel





SECURE CONNECTIONS
FOR A SMARTER WORLD