From COSIDE® to Cadence®

Andreas Schröck
Infineon Technologies AG

Agenda

1. History – project background
2. Conversion from Coside to Cadence – WHY?
3. Problems and Shortcomings
4. The brighter future

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

Agenda

1. History – project background
2. Conversion from Coside to Cadence – WHY?
3. Problems and Shortcomings
4. The brighter future

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

History – project background (1/2)

› Infineon's Radar Success Story

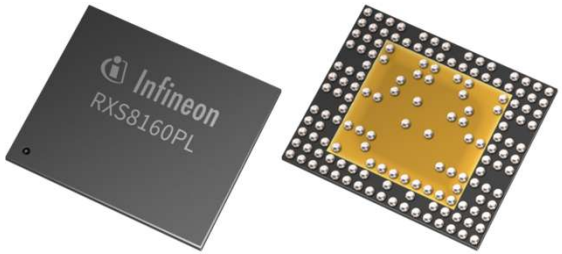
2003 - 2006:
- 77GHz 2RX 2TRX + VCO Prototype

2009:
- series Product (Porsche Panamera)

2012:
- SiGe:C & CMOS Companion

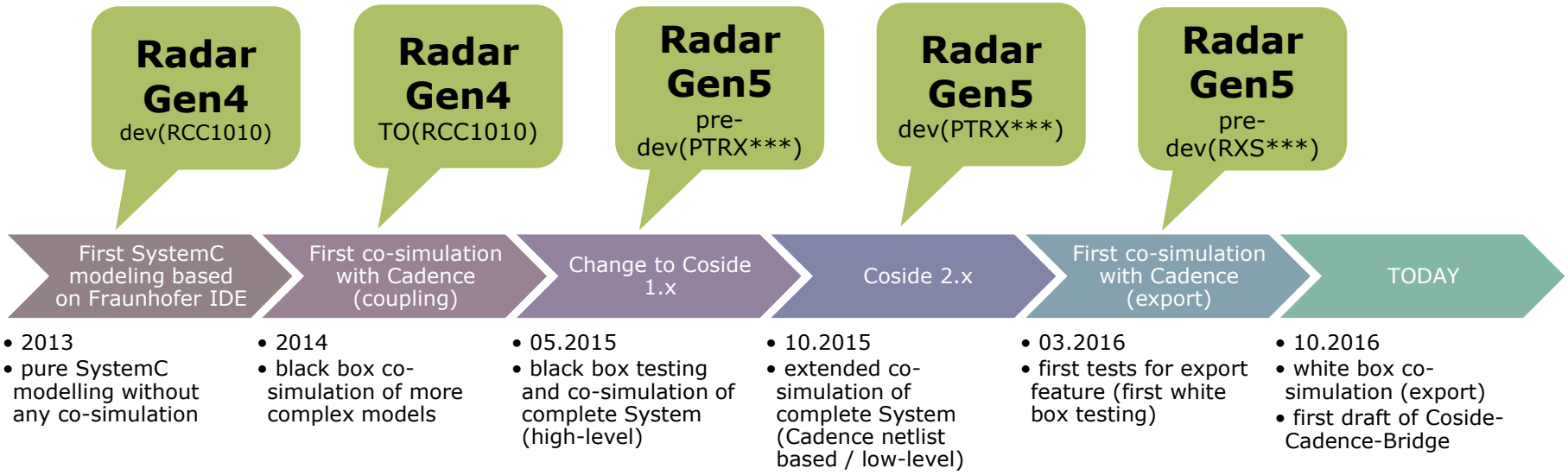
2014:
- BiCMOS Integration

2016:
- Full Integration of RF-IF-ADC-Digital Interface



Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

History – project background (2/2)



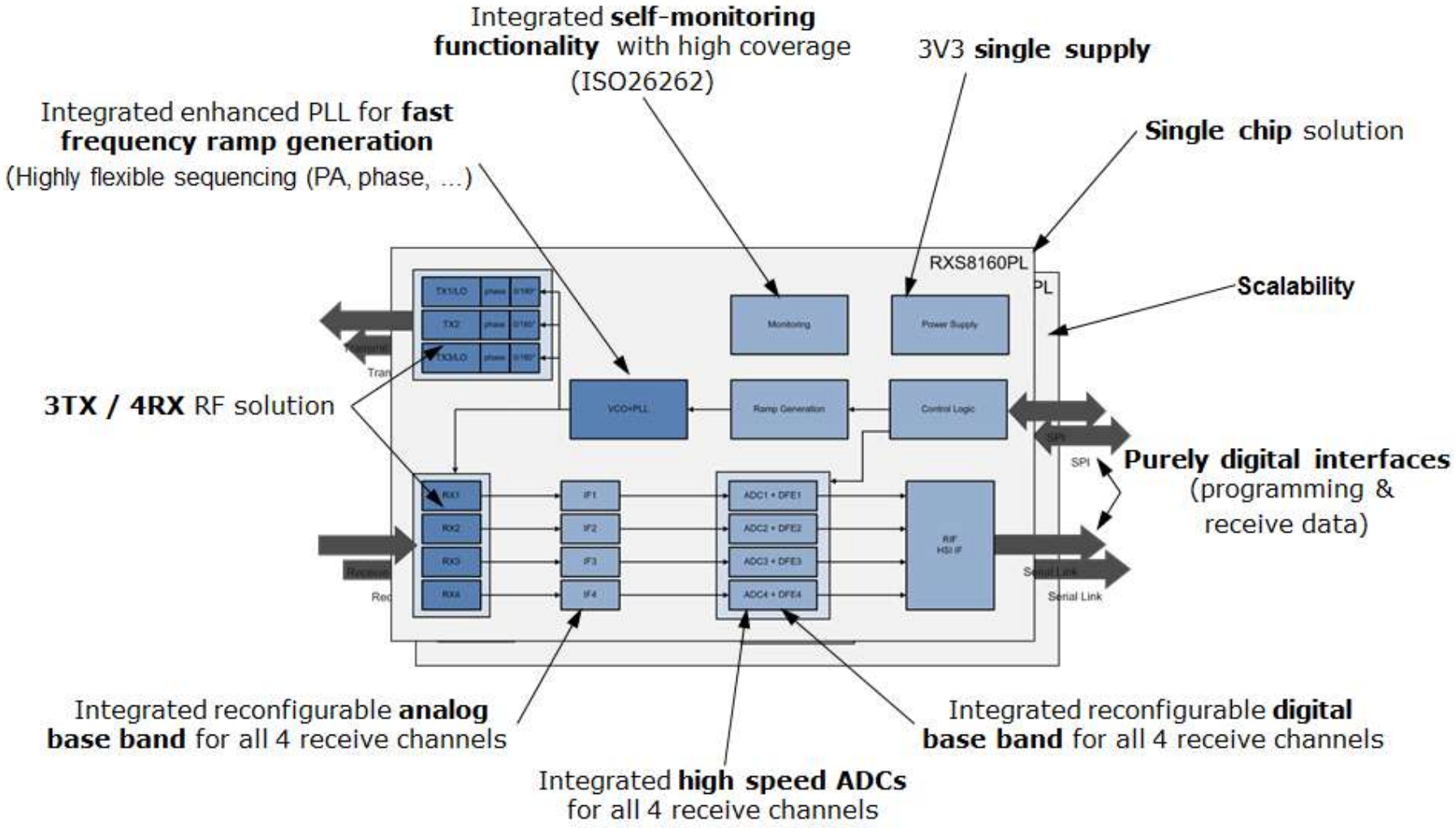
Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

Agenda

1. History – project background
2. Conversion from Coside to Cadence – WHY?
3. Problems and Shortcomings
4. The brighter future

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

From Coside to Cadence – WHY? (1/4)



Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

From Coside to Cadence – WHY? (2/4)

› Increased complexity

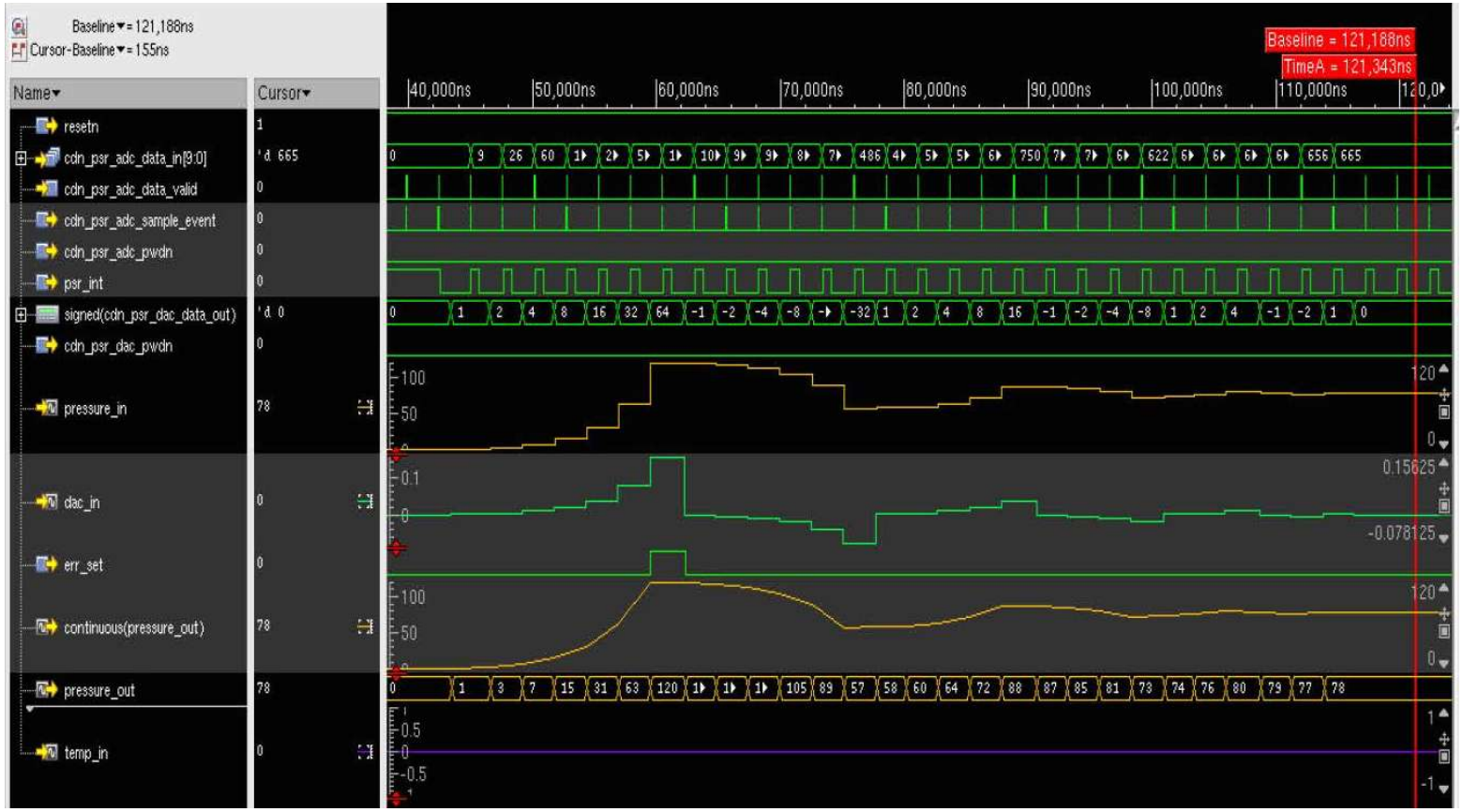
› Demanding customers

› Faster time to market

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

From Coside to Cadence – WHY? (3/4)

› Enabling transient co-simulations on system level



Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

From Coside to Cadence – WHY? (4/4)

- Reduction of simulation time
  - Simulation of PLL

SystemC	Cadence Spectre	Matlab
VCO = 2.6GHz; resolution 83ps; 700us sim time		

- Higher quality of design
  - Use SystemVerilog UVM methodology on System-Level with SystemC models for analog and “real” RTL for digital

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

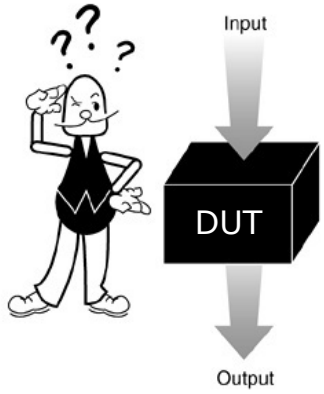
Agenda

1. History – project background
2. Conversion from Coside to Cadence – WHY?
3. Problems and Shortcomings
4. The brighter future

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

Debugging of SystemC Models in Cadence

- Using the coupling method to export SystemC models from Coside leads to a “BLACK-BOX”



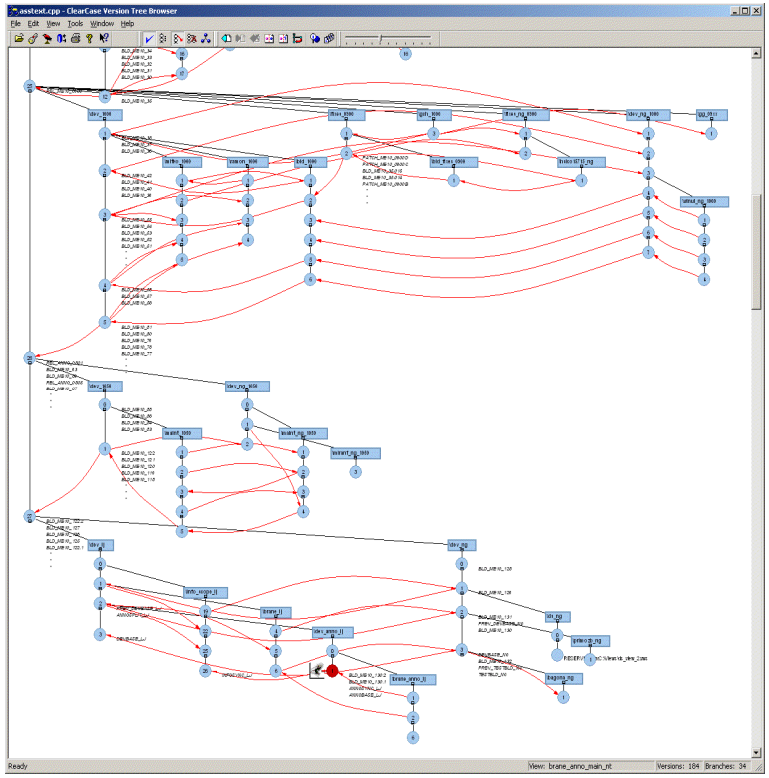
Black-Box Testing

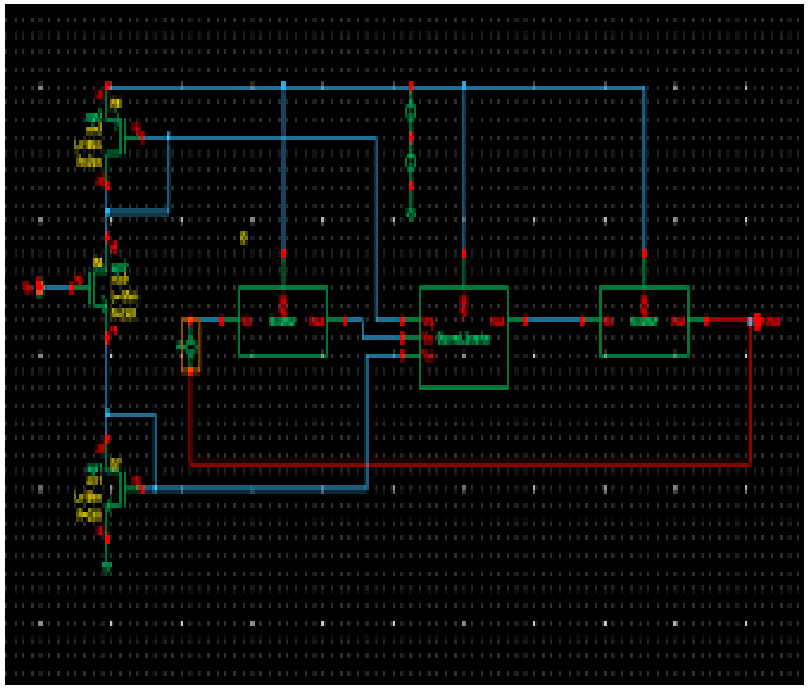
  - No internals of model visible in Cadence
  - Multiple instances not easily possible with shared-objects
  - Bug fixing needs many iterations
  - Conversion issues within connect modules (CMs) are hard to find

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

“Problems” with Cadence and SystemC / Coside

- Keep the design data between Cadence and Coside consistent
  - Cadence uses OA database
  - ClearCase is also not helpful
  - Analog design data is schematic based
  - Changes hard to find

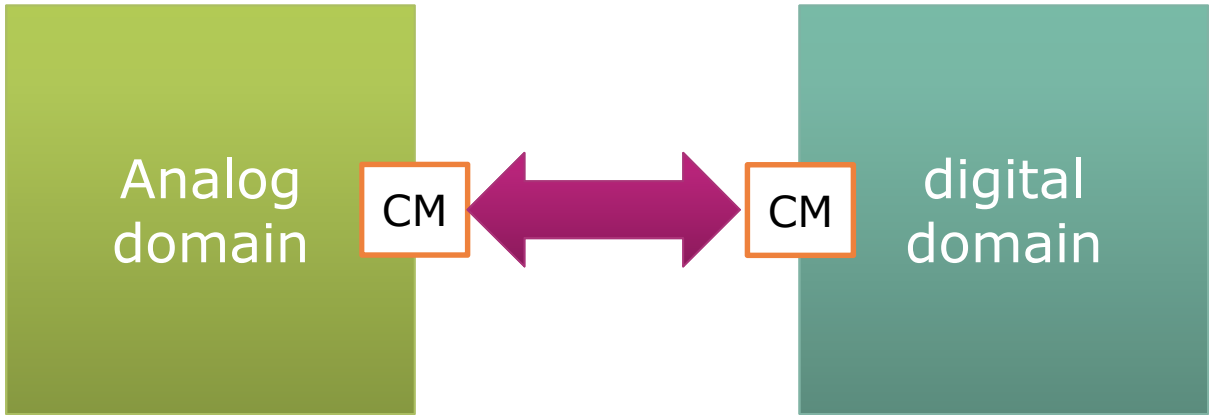




Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

Interaction between different domains (1/2)

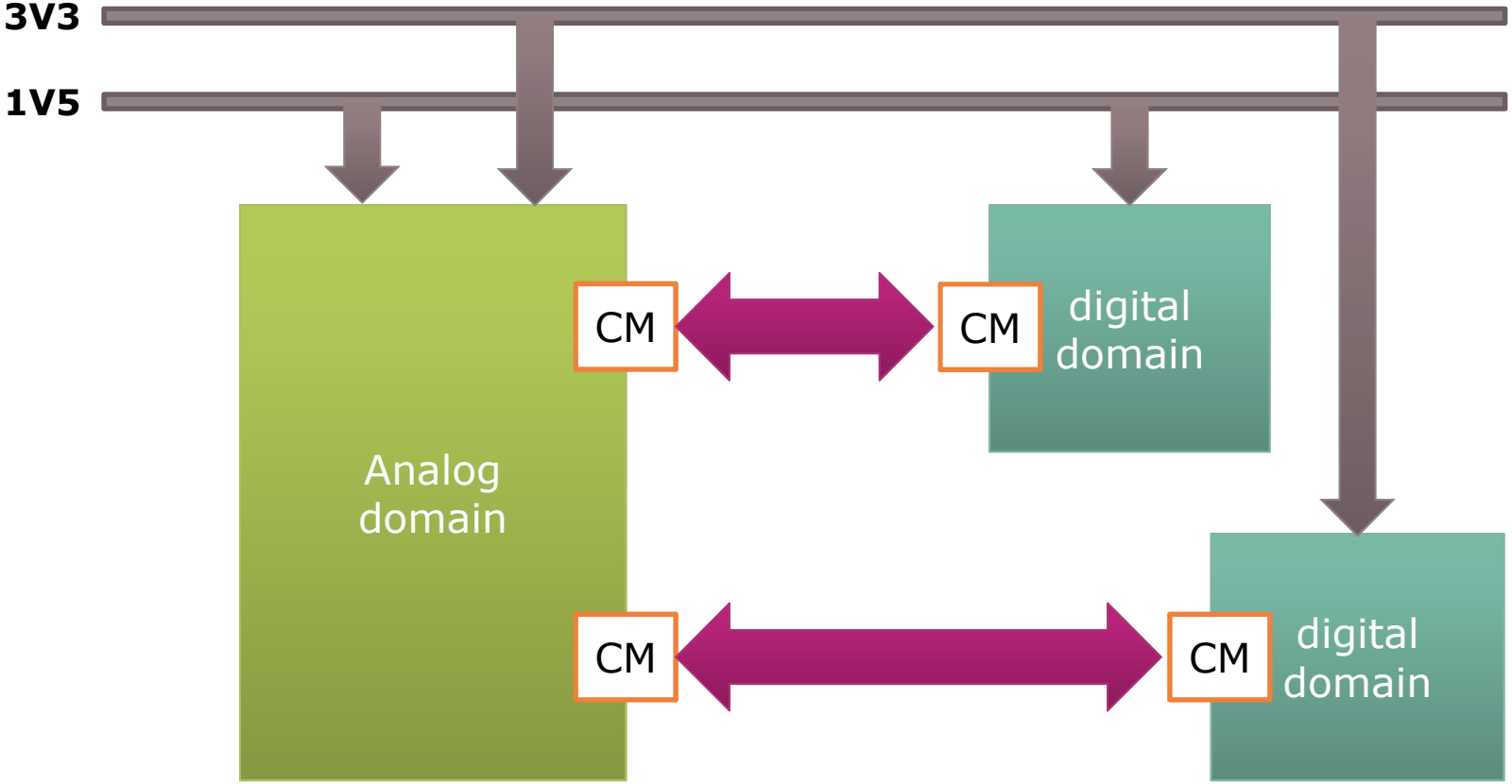
- Data exchange between analog and digital is only possible with "special" conversion cells – connect modules (CM)
  - Insertion shall be / is done automatically
  - no schematic changes / manual insertion allowed



Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

Interaction between different domains (2/2)

- More problems with multi-voltage designs
  - How to define voltages for CMs?
  - Handwritten VAMS wrapper necessary (supply aware)



Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

“Problems” with complex designs

- Interaction between different languages
  - **Digital**: Verilog / VHDL
  - **Analog Models**: SystemC / C++ / Verilog-A / Verilog-AMS
  - **Top-Level TB**: SystemVerilog / Specman
  - **Analog**: spice

- Interaction between different simulators
  - **Digital**: INCISIVE / NC-SIM
  - **Analog**: Spectre / Ultrasim / P-Spice
  - **SystemC Kernel**: C++

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

Agenda

1. History – project background
2. Conversion from Coside to Cadence – WHY?
3. Problems and Shortcomings
4. The brighter future

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

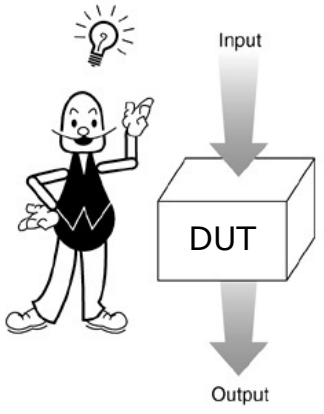
White-Box integration in Cadence

- Use Cadence gcc + kernel to integrate SystemC models in the AMS co-simulation
  - Coside offers a “make” based solution to generate a white-box module for Cadence

make
- Compiles modules using Cadence
- Generates a irun-scripts & resource files

irun
- Compiles a module to .pak & .so
- Additional .res file can be used in AMS-Simuation

ncsim
- Loads .res file to identify available modules
- Simulates SysC modules a white-box instances



White-Box Testing

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

From hand-made to generated (1/3)

- Supply aware VAMS wrapper must be generated based on available design information

- **Needed Information:**
  - Block / module supplies
  - Related supplies for ports
  - Custom defines for AMS-Simulator

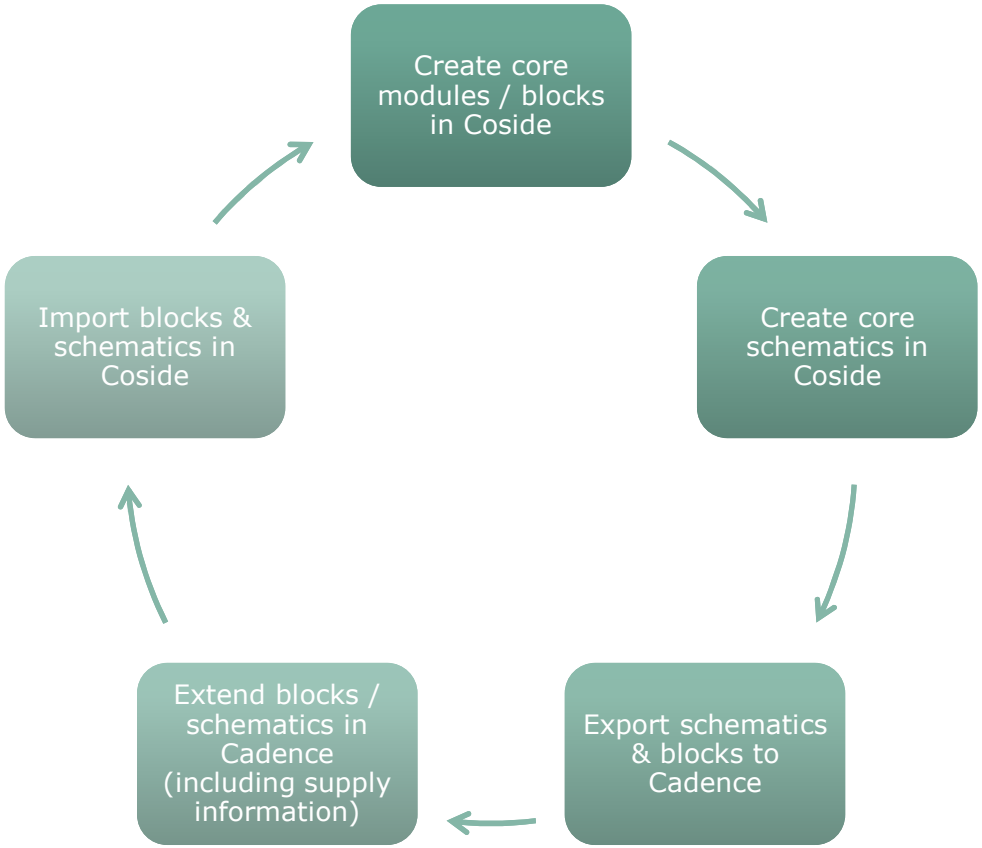
- **Solution from Coseda:**
  - Add all necessary information as port / schematic VAMS-attributes
  - Attributes are inherited through hierarchies

- **Drawback:** Data insertion in Coside might become a little tricky for bigger designs. Is there a better way? → YES

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

From hand-made to generated (2/3)

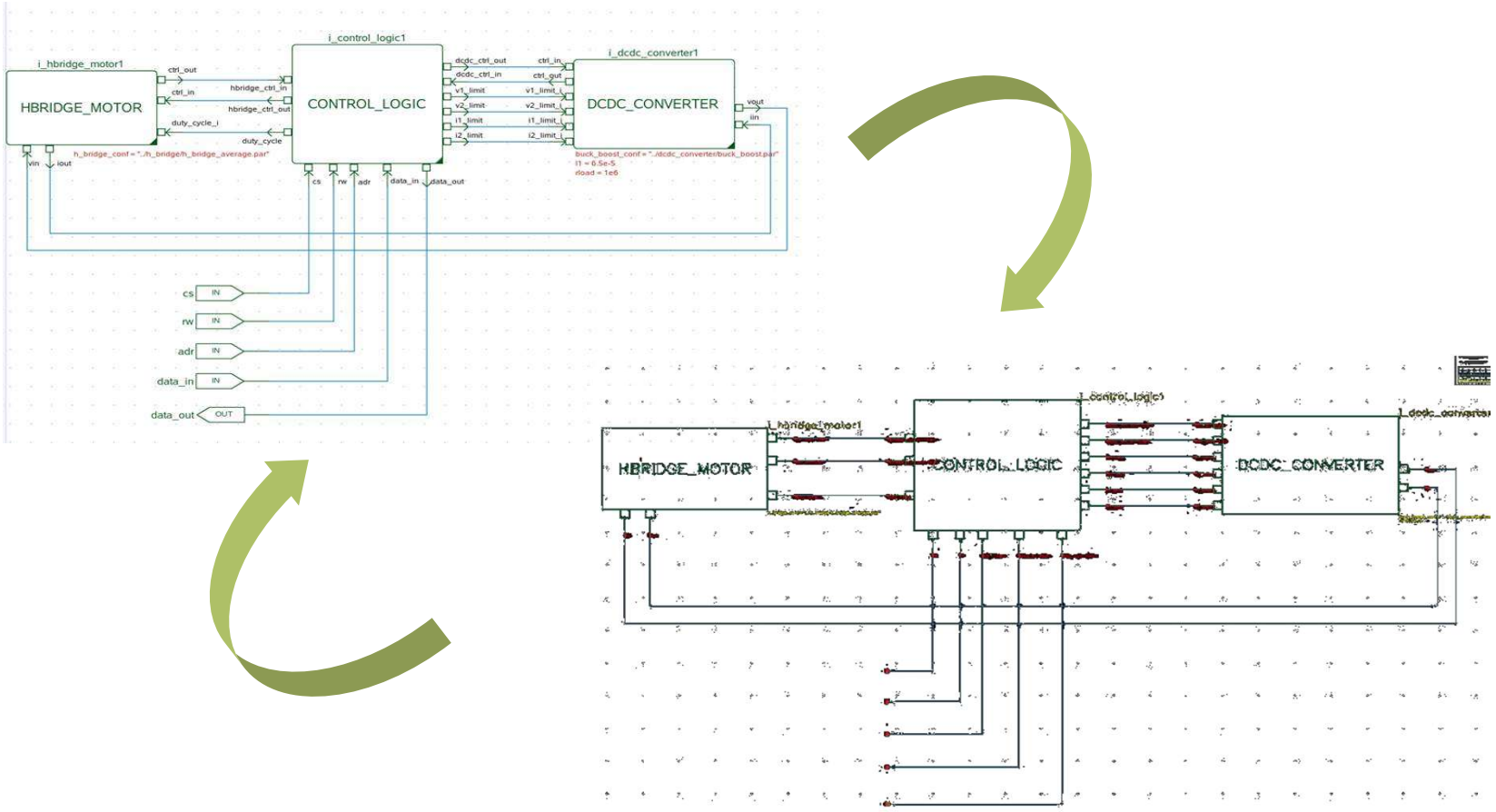
- The better way to insert power supply information
  - Use the export / inport feature from Coside to exchange data and attributes between Coside and Cadence (Coside-Cadence-Bridge)



Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

From hand-made to generated (3/3)

› Schematic & block conversion from / to Coside / Cadence to keep data in sync



Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

Q&A

› Questions?

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**

Part of your life. Part of tomorrow.

infineon

Conclusion

- Coside / SystemC helps us during the development of bigger designs to achieve
  - Better quality of the system
  - Faster development cycle

- There is still room for improvement

Copyright © Infineon Technologies AG 2016. All rights reserved. **Infineon Proprietary**