



Utilizing ASIP Designer for Industry Projects (BMBF), Teaching Activities, and Research

ASIP University Day 2021
17.11.2021

Lennart Reimann, M.Sc
Univ.-Prof. Dr. rer. nat. Rainer Leupers

Agenda



Institute Introduction

Teaching Activities

BMBF Project

Research Activities

Agenda



Institute Introduction

Teaching Activities

BMBF Project

Research Activities

ICE @ RWTH Aachen University

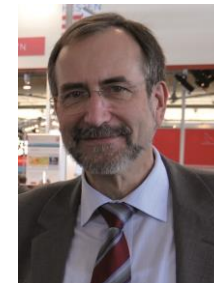
- **Institute for Communication Technologies and Embedded Systems**

- Chair for Distributed Signal Processing (DSP)
- Chair for Software for Systems on Silicon (SSS)

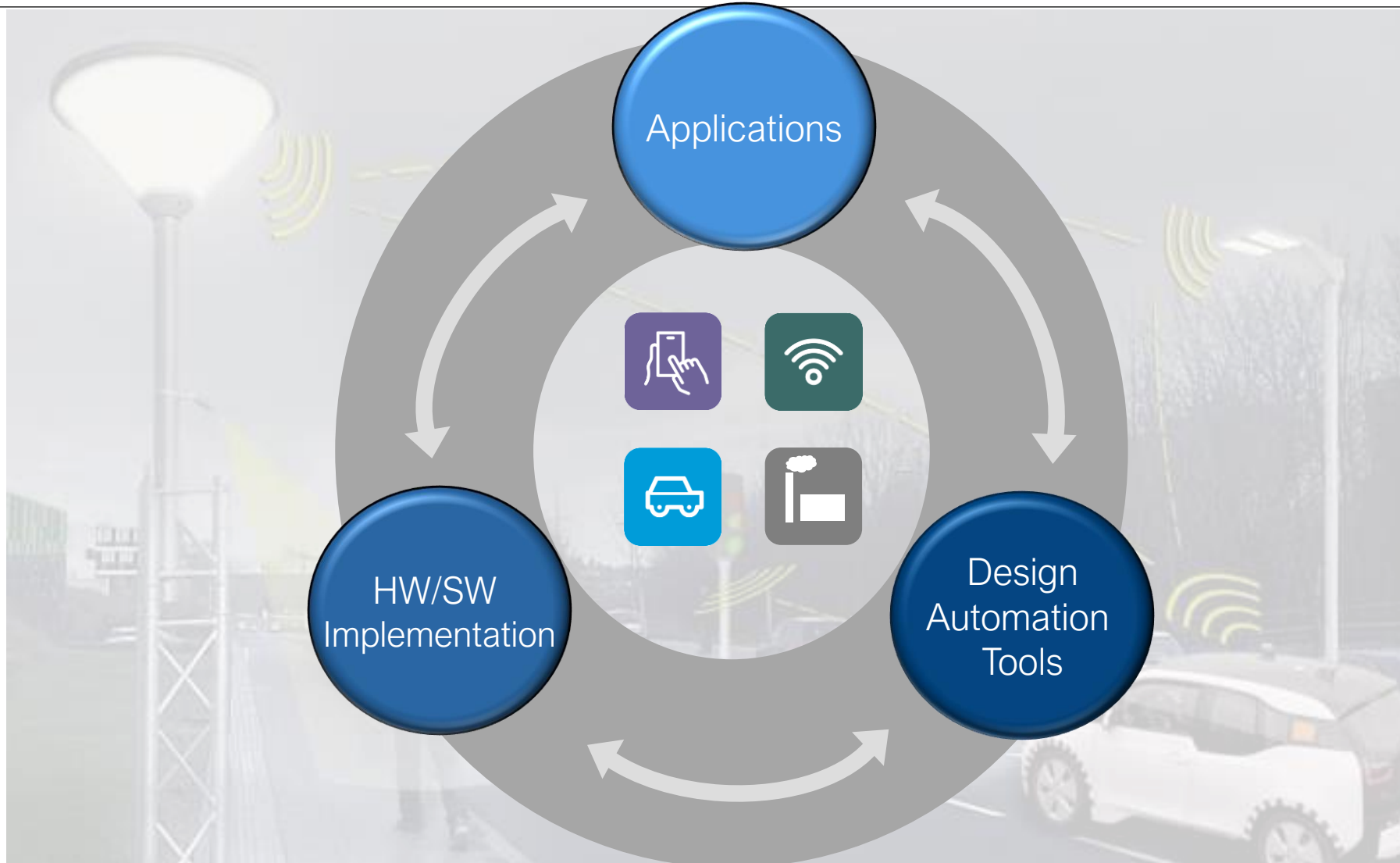
- **Team:**

- Jointly managed by two professors:
Haris Gacanin, Rainer Leupers
-
- Emeriti: Gerd Ascheid, Heinrich Meyr

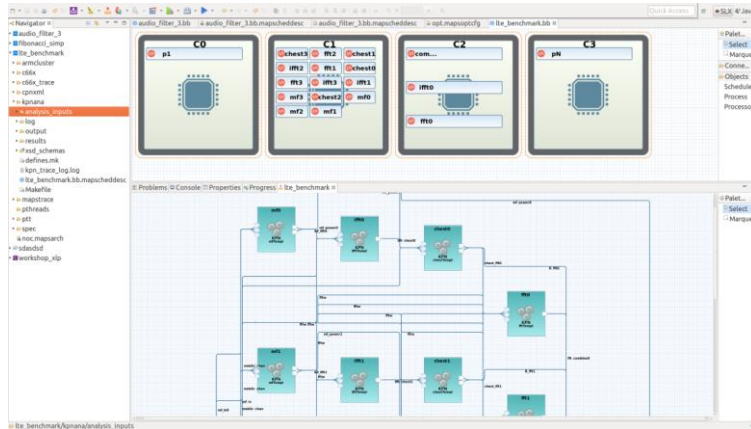
- Guest professors: A. Hoffmann, G. Dartmann
- ~20 PhD/Postdoc researchers
- 8 non-academic staff



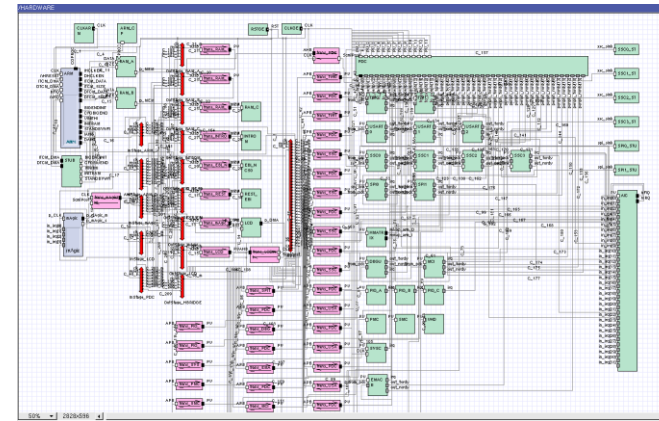
ICE Mission: Application-Specific System Design



R&D topics in HW/SW design technologies



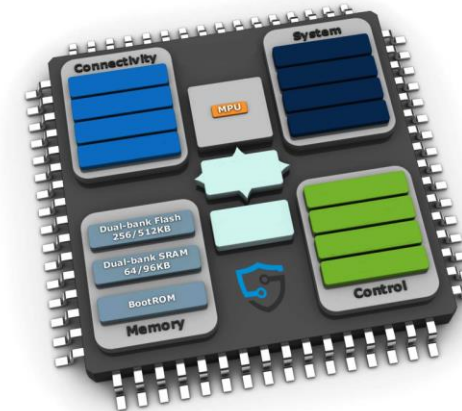
Programming tools



System-on-chip design



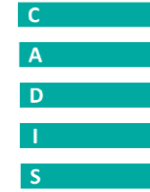
Virtual prototyping



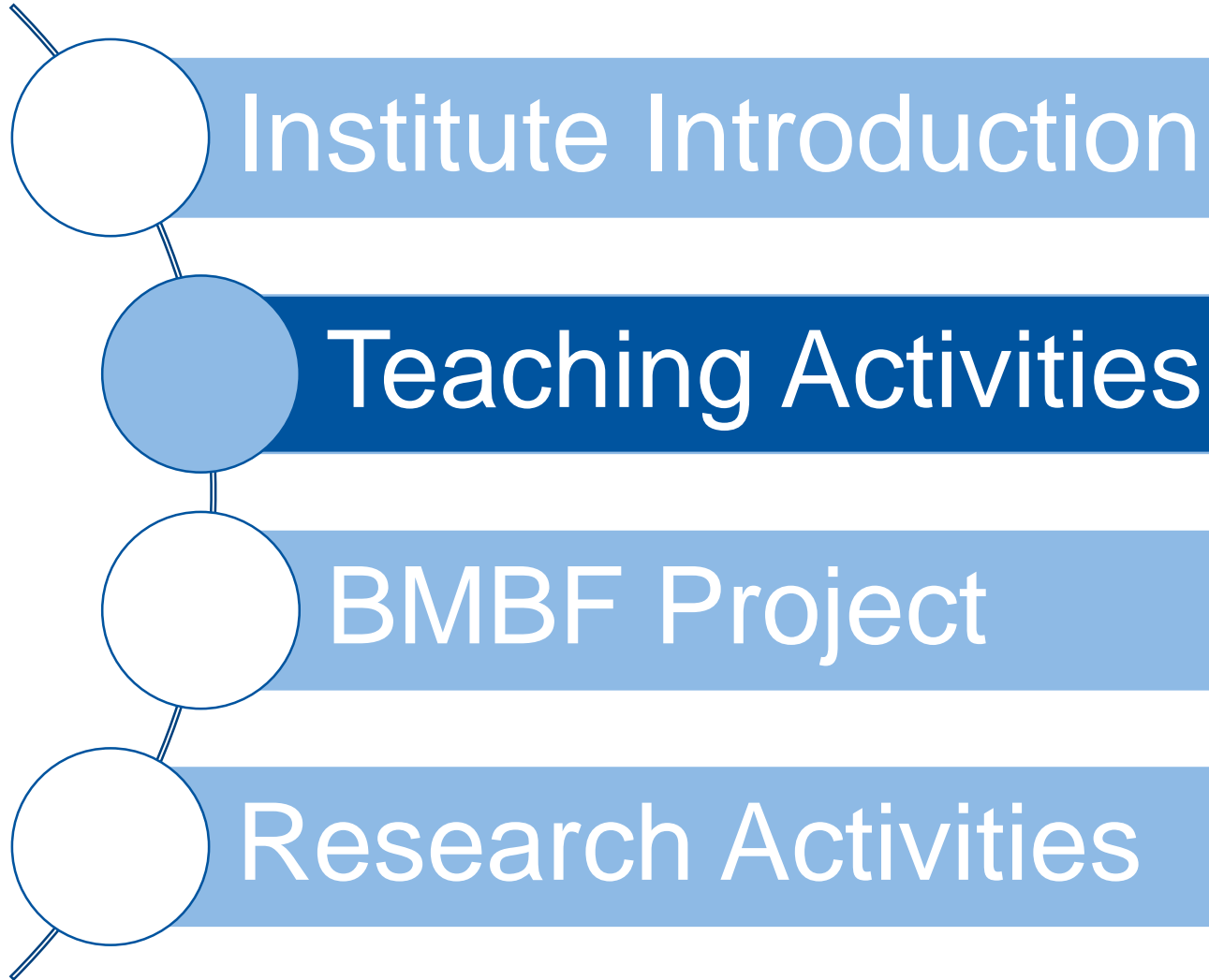
Hardware security

ICE Spin-off history

- Cadis
 - Design and simulation of DSP algorithms
 - acquired by Synopsys in 1994
- ATecoM (1994)
 - ATM hardware design
- Axys
 - Fast simulation and SoC design tools
 - Acquired by ARM
- LISATek (2002)
 - Embedded processor design automation
 - Acquired by Synopsys in 2010 (via CoWare Inc.)
- ICE Gateway (2013)
 - Integration of intelligent street lighting and M2M
- Silexica (2014)
 - Multicore programming tools
 - Aquired by Xilinx
- [Secure Elements] (2016)
 - High-security OS and CPU



Agenda



Project: Embedded Processor Design and Optimization

- Teaching project during summer semester
- Average of **10 participants** (master course), many students start as student assistants afterwards
- Project starts with theoretical lectures about processor design
 - What is a program counter unit?
 - What is pipelining?
 - What is a register file?
- Goal: Optimize a given **RISC-V** processor (RV32IE) for two cryptographic algorithms:
 - SHA-256
 - AES-GCM-256
- What do the students learn?
 - How in-order processor architectures work
 - How to do:
 - Profiling and identify hotspots
 - Use ASIP Designer to modify the control flow, add instructions, add memories
 - Use Design Compiler to synthesize the generated RTL code

How is the project conducted?

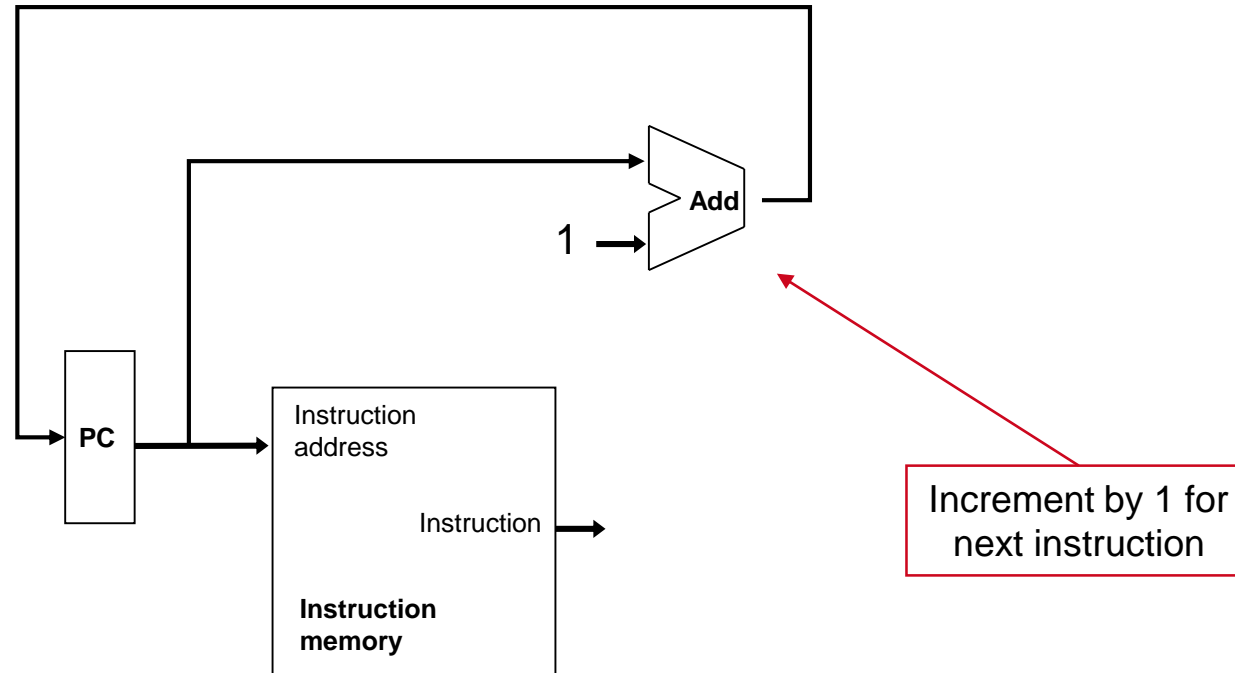
1. Three **theoretical lectures** about:

1. Hardware design (critical paths, combinational and sequential logic)
2. Processor Design (pipelining, registers, memories,..) and ASIP Designer
3. Logic Synthesis & Applications (SHA-256 and AES-GCM-256)

2. Getting to know ASIP Designer

1. Handing out the processor model and example applications (sorter,...)
2. Handing out a **quiz** that requires the students to click around in the development environment
 - Where are the memory interfaces defined? Where is the ALU defined? At what memory address is variable „s“ stored for the application ‚sorter‘?....
3. A **hands-on tutorial** on how to optimize the processor for the SHA applications, based on the SHA-workshop by Synopsys
4. Handing out a new application and stating a design performance goal
 - Students will have to **work on their own** from here on and achieve the design goal

FROM: Datapath – Instruction Fetch and Program Counter

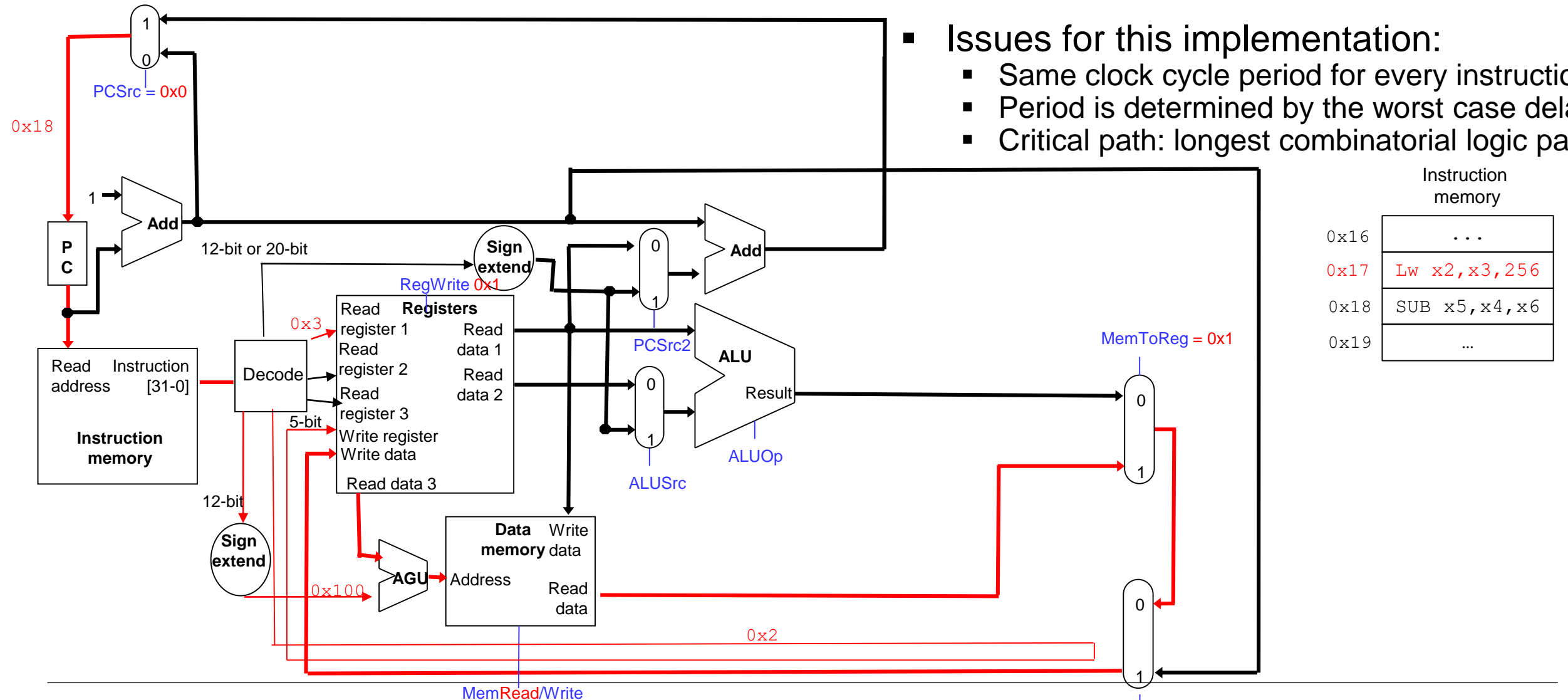


- Instruction fetch and PC increment

TO: RISC-V Integer – Single-Cycle Microarchitecture

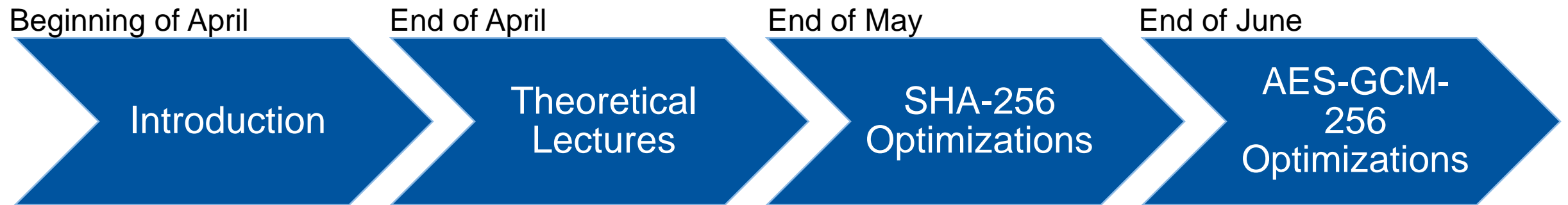
Issues for this implementation:

- Same clock cycle period for every instruction
- Period is determined by the worst case delay
- Critical path: longest combinatorial logic path

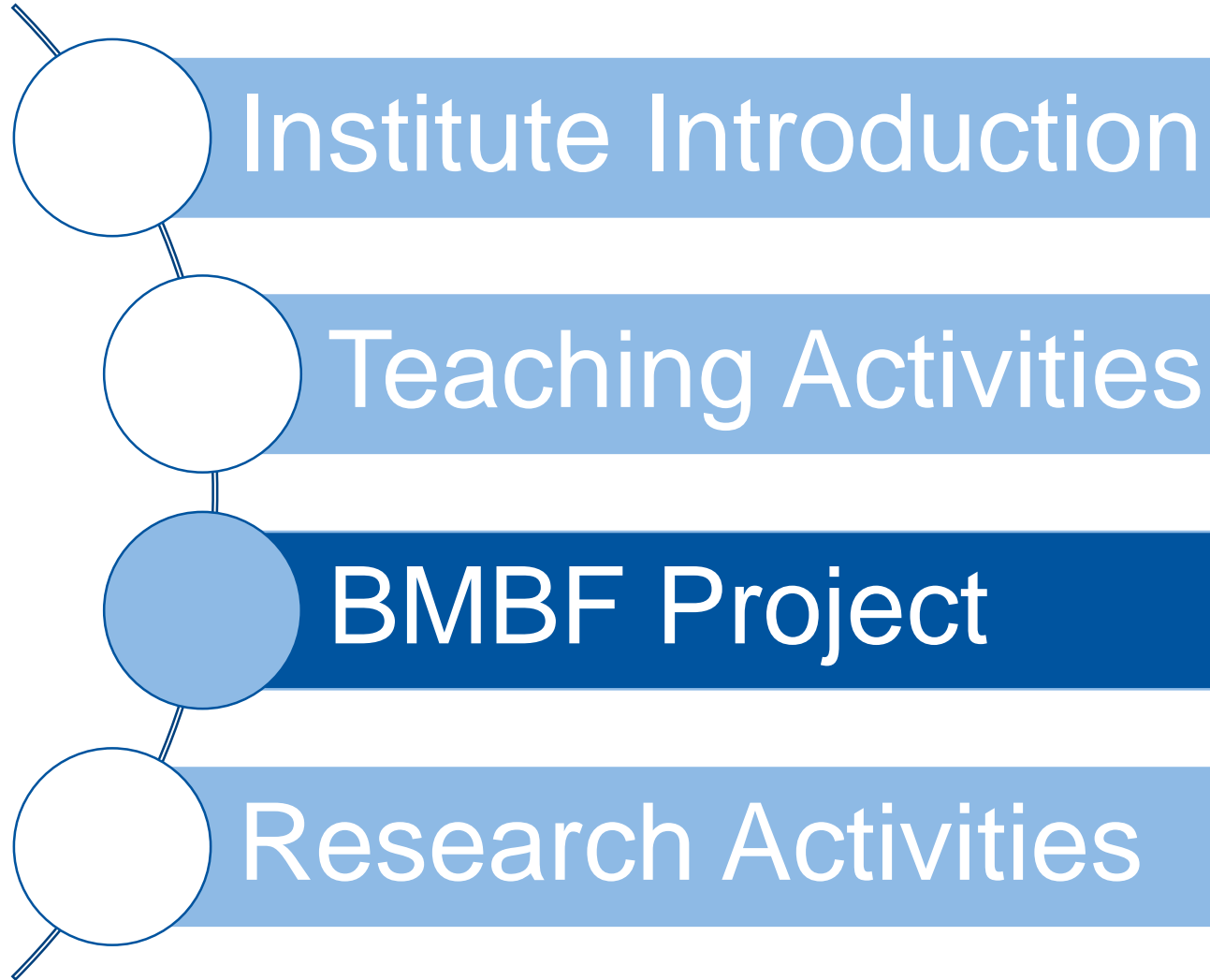


Time Schedule

- Project starts in April and finishes at the end of July
- Project concludes with a short presentation of the individual AES-GCM optimizations and their evaluation

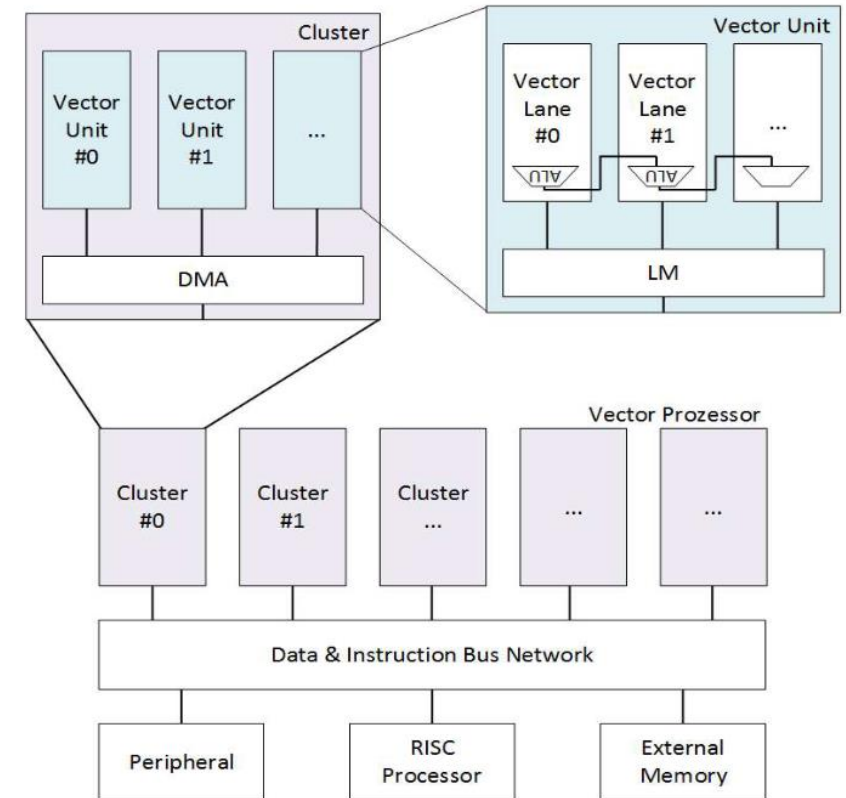


Agenda

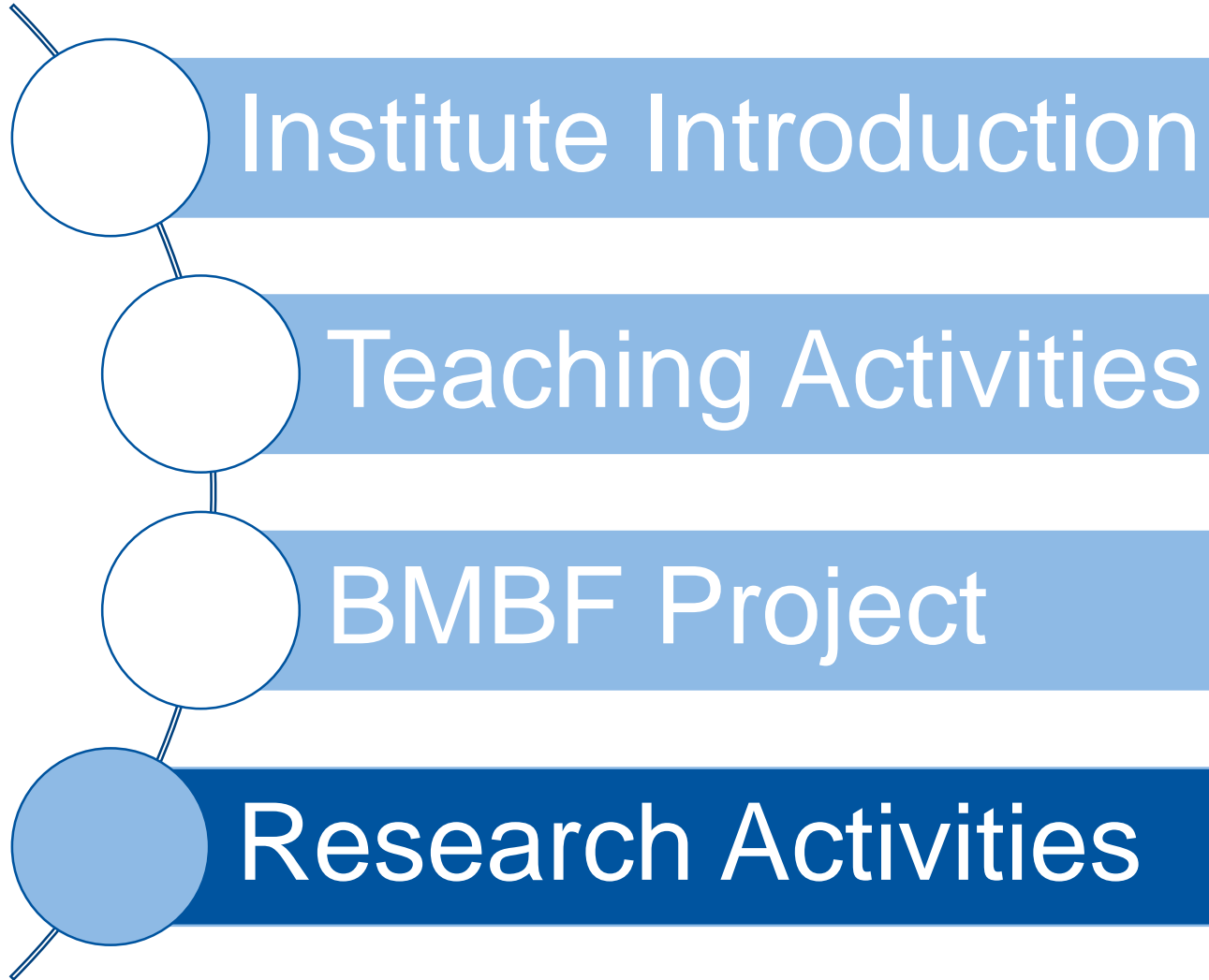


BMBF Project: ZUSE-KI-AVF

- Project goal: Developing a configurable massively-parallelizable vector co-processor architecture
 - Should run radar, lidar, image processing algorithms
 - Additionally, should speed up convolutional neural networks
- The processor architecture comes from the IMS, Hannover
 - It is controlled by a RISC-V architecture
- The ICE will provide a compiler to program the RISC-V and his co-processor (right)
- ASIP Designer is normally not used to program main & co-processor
 - New challenge
- First approach is going to use intrinsics to communicate with the co-processor
- Following steps? Vector data types? Compiler optimizations for intrinsics?

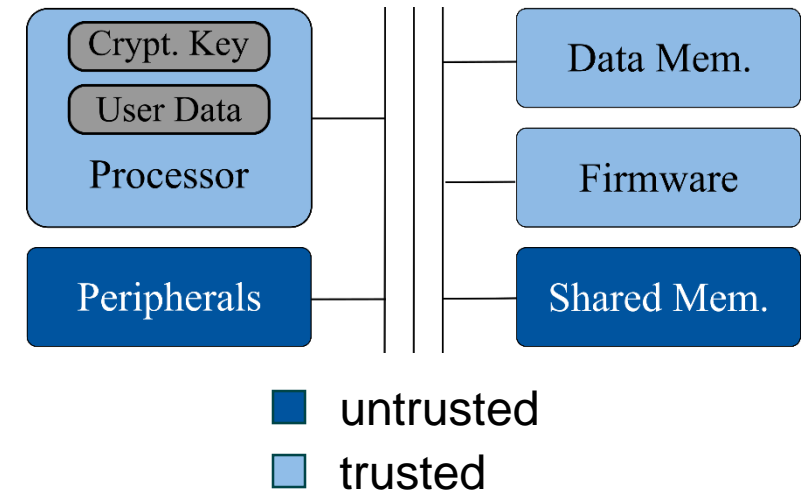


Agenda



Software Verification using ASIP Designer as a DE

- **Information flow analysis** can give you the leakage paths of sensitive signals
 - You can either remove those leakage paths or
 - Verify that those leakages are not used for a given software
- For our elaborations we use the
 - RTL Generator
 - Generated Compiler:
 - Call Graphs
 - Hex Files
 - ...
 - Our tool for the information flow analysis: QFlow_[1]
- Changes inside the hardware description are easily integrated into the compiler
 - Everything can be automated



Questions

If you have any questions related to any of the three topics, feel free to contact me:

Lennart.Reimann@ice.rwth-aachen.de