Topics for Seminar Talks

Digital System Integration and Programming

Barbara Gigerl, Rishub Nagpal

October 6th, 2021

IAIK – Graz University of Technology

Registration Process

1. Find a group
2. Register your group: `mailto:sip-team@iaik.tugraz.at`
3. Wait for the confirmation mail to get your group number
4. Decide when to pick up the HW (IF01052, Mo-Fr 10:00-16:00)
5. Choose a seminar topic
6. Register for a seminar topic: `https://bit.ly/2Ys9Cvi`

Deadline: 12.10., 23:59

Organizational Information

- Select a topic from the catalogue or send us your ideas!
  - SoC Basics
  - SoC Security
  - SoC Environment
- By choosing a topic you agree on presenting on one of the possible dates
- Length of presentation: 20 min + 10 min
- Optional: use the template
- Submit your slides until Monday evening - we will review your presentation and send you feedback.

Preliminary Timetable

SoC Basics	20.10., 27.10.
SoC Security	27.10., 3.11., 10.11., 17.11., 24.11.
SoC Environment	1.12., 15.12., 12.1., 19.1., 26.1.

SoC Basics

Topic #1: Architecture of FPGAs

FPGAs consist of configurable logic blocks (CLBs), connected by programmable interconnects. Depending on the manufacturer, these CLBs can further be divided into logic blocks.

- What is a CLB and what role do interconnects play?
- How and when are CLBs configured?
- What are IOBs (Input Output Blocks)?

Topic #2: ARM AXI Interface

SocS frequently use the ARM AXI interface for on-chip communication.

- What is the purpose of the AXI interface?
- Describe the AXI handshake mechanism.
- Which channels are described by the AXI specification?
- What is the purpose of AXI stream?

Topic #3: Example AXI peripheral access

The Zybo boards use the AXI protocoll to communicate with peripherals, e.g. the block ram.

- Explain example usage scenarios of the AXI protocol. Choose one meaningful example.
- Show the steps which are necessary to read from memory.
- Show the steps which are necessary to write from memory.
- Use visual diagrams, e.g. timing diagrams and explain all steps in detail.

Topic #4: Alternative SoC Bus Interconnections

Communication using buses is a critical aspect of SoCs. Several architectures besides AXI exist.

- Which types of bus technologies exist?
- What are the challenges when designing buses?
- Describe a few other protocols and explain the difference to the AXI protocol.

Topic #5

Topic #5: Network-on-Chip (NoC) designs

Network-on-Chip (NoC) can be seen as an alternative over traditional bus-based architectures.

- How is the network-on-chip design paradigm characerized?
- What are the differences/advantages to bus-based architectures?
- How could a sketch of a NoC look like?

SoC Security

Topic #6

Topic #6: FPGA Bitstream Encryption Basics

Most FPGAs provide a mechanism to encrypt bitstreams. This feature protects designs from being copied, altered or reverse engineered.

- Why is bitstream encryption needed on FPGAs?
- How does bitstream encryption work?

Topic #7

Topic #7: FPGA Bitstream Encryption Vulnerabilities

Bitstream encryption isn't perfect. Research has shown that some can be broken in various ways.

- Give an overview of existing attacks on bitstream encryption.
- Ender et. al, *The Unpatchable Silicon: A Full Break of the Bitstream Encrypton on Xilinx 7-Series FPGAs*, In: USENIX'20 (2020).

Topic #8: Hardware Trojan Attacks in FPGAs

Hardware trojans pose serious security concerns. In recent years, researchers showed that FPGAs are vulnerable to such attacks.

- What is a hardware trojan and to what extent are they dangerous?
- Wang et. al, *Hardware Trojan Attack in Embedded Memory*, In: JETC, Volume 17, Issue 1 (2021).
- Which countermeasures exist?

Possible Dates: 27.10.

Topic #9: Fault attacks on FPGAs

FPGAs are vulnerable to fault attacks. In this attacks, incorrect system behavior is, for example, triggered by voltage drops.

- Explain the basics of fault attacks.
- Which attack scenarios for fault attacks on FPGAs exist?
- Krautter et. al, *Remote and Stealthy Fault Attacks on Virtualized FPGAs*, In: DATE 2021 (2021)
- What are possible countermeasures?

Topic #10

Topic #10: EM Side-Channel Attacks on SoCs

SoCs are vulnerable to ElectroMagnetic (EM) side-channels. In this attacks, incorrect system behavior is, for example, triggered by voltage drops.

- Explain the basics of EM side-channel attacks.
- Longo et. al, *SoC it to EM: electromagnetic side-channel attacks on a complex system-on-chip*, In: CHES 2015 (2015).
- What are possible countermeasures?

Topic #11: Security Co-Processors

Sometimes it's necessary to keep things separate to get both security and performance. Security co-processor can do that.

- When to use a security co-processor?
- How do they communicate?
- How do they perform in comparison?
- Which guarantees can they give?
- Steinegger et. al, *A Fast and Compact RISC-V Accelerator for Ascon and Friends*, In: CARDIS 2020 (2020).

Topic #12: Reverse Engineering ICs

Vendors of ICs invest significant effort to counteract attempts to reverse engineer their product. Still, there exist several approaches to reverse engineer ICs.

- What are the general steps when reverse engineering an IC?
- Which methods exist?
- Azriel et. al, *A survey of algorithmic methods in IC reverse engineering*, In: JCE 2021 (2021).

Topic #13: Reverse Engineering: FPGAs vs ASIC

FPGAs and ASICs tend to be proprietary black boxes. How to find out what going on.

- Which methods exist to reverse engineer?
- What are their limitations?
- What are manufacturers doing to prevent it?

Topic #14

Topic #14: Security by Obfuscation for FPGAs

Obfuscation methods are often applied by vendors to protect FPGAs from reverse engineering.

- What is security by obfuscation?
- How can it be achieved?
- Labafniya et. al, *An Obfuscation Method Based on CFGLUTs for Security of FPGAs*, In: ISeCure 2021 (2021).
- Karam et. al, *Robust Bitstream Protection in FPGA-based Systems through Low-Overhead Obfuscation*, In: ReConFig 2017 (2017)

Topic #15

Topic #15: TEEs and Enclaves

Trusted execution environments and enclaves allow for secure code execution without separate hardware.

- Which commercial and academic approaches exist?
- How do they differ?
- Attackson/utilizing TEEs/Enclaves

SoC Environment

Topic #16: Alternative HDLs

Today, most applications are still traditionally written in Verilog, System Verilog or VHDL. However, there exist many more alternative HDLs, including Bluespec and Chisel.

- What alternatives to traditional HDLs exist?
- Using code snippets, what are their characteristics?
- What are the major road blocks of replacing traditional HDLs?

Topic #17: Verification of SoCs

Verification and Testing is an essential part of the SoC design process. As the complexity of SoCs increases, the need for efficient verification method increases.

- What is the goal of SoC Verification?
- What is the difference between SoC verification and testing?
- Which strategies and tools exist?

Topic #18

Topic #18: Open-Source Hardware Toolchains: SymbiYosys and Yosys

Proprietary tools can be cumbersome and expensive. Open-source should run on open-hardware built using open-source tools.

- What are the tools doing?
- How do they compare to their proprietary counterparts
- How to use them? Give a short demo!

Topic #19: Hardware/Software Co-Verification

Co-verification of SoCs addresses one of the most critical steps in the embedded system design process, the integration of hardware and software.

- How is Hardware/Software Co-Verification defined?
- Which Co-Verification methods exist?
- Herber et. al, *Combining Model Checking and Testing in a Continuous HW/SW Co-verification Process*, In: TAP 2009 (2009)

Topic #20: Design of Mixed-Signal SoCs

Digital is fine, but the world is analogue. Applications like wireless communication require both

- Which applications require mixed-signal SoCs?
- How is this done in SoC designs?
- How can this be realized with FPGAs?

Topic #21: Booting Linux

You press a button and suddenly there's a shell. How did the device get there?

- How does Linux boot on ARM/RISC-V/x86?
- Which role plays the BIOS/UEFI?
- What is Secureboot and what can it do?
- What is a bootloader and why is there one called the Berkeley Bootloader (BBL)?

Topic #22: Soft Cores and ARM/RISC-V Processors

In a SoC, processors are often placed as a standalone unit, but can also be delivered as a HDL design which is then synthesized and used as an FPGA configuration.

- What are soft cores and what is the difference to hard cores?
- Which ARM soft-cores exist?
- Which RISC-V soft-cores exist?

Topic #23: FPGAs and Neural Networks / ZynqNet

FPGAs and SoCs are often used for neural computing, for example ZynqNet, which is based on the Zynq SoC.

- Why are neural networks on FPGAs and SoCs so popular?
- What is the application area?
- Describe ZynqNet.
- Gschwend, *ZynqNet: An FPGA-Accelerated Embedded Convolutional Neural Network* (2016).

Topic #24: FPGAs in Space

FPGAs have been used in space for more than a decade. In order to be feasible for space applications, FPGAs need to fulfill a row of requirements, including radio tolerance.

- To which extent are FPGAs suitable for space?
- What are the main challenges when using FPGAs in space?
- Which manufacturers provide such technologies?

Topic #25: Rocket Chip Generator

The RISC-V Rocket Chip Generator is an open-source SoC design generator. It can be used to generate synthesizable RTL.

- What is the idea behind the generator?
- How can a core be configured and which HDL is used?
- For which RISC-V cores has the generator already been used?

Topic #26: High Level Synthesis with Google XLS

The XLS project by Google represents a high-level synthesis toolchain to generate synthesizable designs from high-level specifications.

- What is high-level synthesis?
- Which toolchains exist for high-level synthesis of hardware?
- What is the idea behind the Google XLS project?

Topic #27

Topic #27: System-on-Chip simulation

SoCs are often very complex systems, which is why pre-silicon verification has become very important. Sophisticated simulation tools are needed in order to achieve these goals.

- What are the application areas of SoC simulation?
- What are the advantages and disadvantages of SoC simulation?
- Which tools exist for this task?

Topic #28: Hardware Package Manager: FuseSoC/Bender

Designs with a large number of modules can get messy really quick. Package Managers can help

- How do they work?
- Which problems can they solve?
- What are the limitations?
- How do they resolve dependencies?
- Demonstration on how to use them.