EDA Tools: System Specification Exercise / Practice

Agenda

1. Organisation of practice labs
2. CRC-4 example
3. Introduction to (formal) specification
4. Formal specification of a two-stage shift register

Agenda

1. Organisation of practice labs
2. CRC-4 example
3. Introduction to (formal) specification
4. Formal specification of a two-stage shift register

Chair Circuit and System Design

Practices

Components and Architectures

EDA-Tools

(SmartSensorSystems)

TECHNISCHE UNIVERSITÄT CHEMNITZ

Subscription to practice

is made in OPAL:

https://bildungsportal.sachsen.de/opal/dmz/

- Is possible after subscription to exercises only!
- Several groups available - please check the times of the lessons!
- Check the exact starting times and locations of the labs!
- Download the according manual and solve the preparation tasks!

Subscription to practice

The groups are used in:

- Components & Architectures of embedded Systems

  (room 2/W465)
- EDA-Tools (room 2/W451)
- Smart Sensor Systems (M_Es only,

  held by Chair Measurement and Sensing Technologies)

Your login is valid for all SSE master courses
(EDA-Tools 1+2, C&A, System Design 1,
Design of Heterogeneous Systems, Rapid Prototyping,
Software Environments for Smartphone Applications)

Requirements for exam admission

- passing all 4 lessons in winter semester (all master courses) of the according course
- submission of design in summer semester (IS, IC only)
  (students from SS14 may submit the solution up to January 4th,
  but no assistance by advisor possible in winter semester)
- successful students from former semesters keep their admission, but exam questions will rely on last lab cycle
  (but there are no major changes to 2013 lab cycle)

Workflow

- Read the lab manual carefully & print out the solution sheet
- Check again the according lecture and exercise material
- If you have a dedicated question: Contact advisor in advance
- If you have not enough preknowledge:
  Fill this gap by your own (e.g. go to library)
- Fill the preparation tasks on the solution sheet

  Attention: Personalized tasks!
- Come to the lab room at least 5 minutes before starting time
- 2 students work together on one PC
- There will be oral questions during the lab - be prepared!
- If you could not come for an important reason: Contact advisor immediately!
  (important reasons may be: serious own health problems, Ausländerbehörde)

Organization of practice

```
for (int LabNr = 1; LabNr < 5; LabNr++) {
  if (onTime AND preparationSheetFilled) {
    get_question_from_advisor();
    if (correctAnswer) {
       do_the_lab();
       if (labSolved AND solutionExplained)
         pass[LabNr] = TRUE;
  }} else {
    if (importantReason AND firstAttempt[LabNr]) {
      contact_advisor_immediately();
      show_proof();
      get_new_date();}
    else {
      repeat_all_labs(November_2015);
     break;}}}
```

If you don’t understand this - improve your C++ language skills!

Contact to advisors

Consultation hour:
every Thursday 10:45-11:30
in 2/W460 (Matthias Sauppe for EDA-Tools 1+2)
and 2/W430 (Erik Markert for other courses)

Other dates:
E-Mail erik.markert@etit.tu-chemnitz.de
or use E-Mail function in OPAL

All communication from advisors will be done by e-mail using OPAL:

- Check the according mailbox (usually TUC account) frequently
- Ensure that mailbox is not over quota

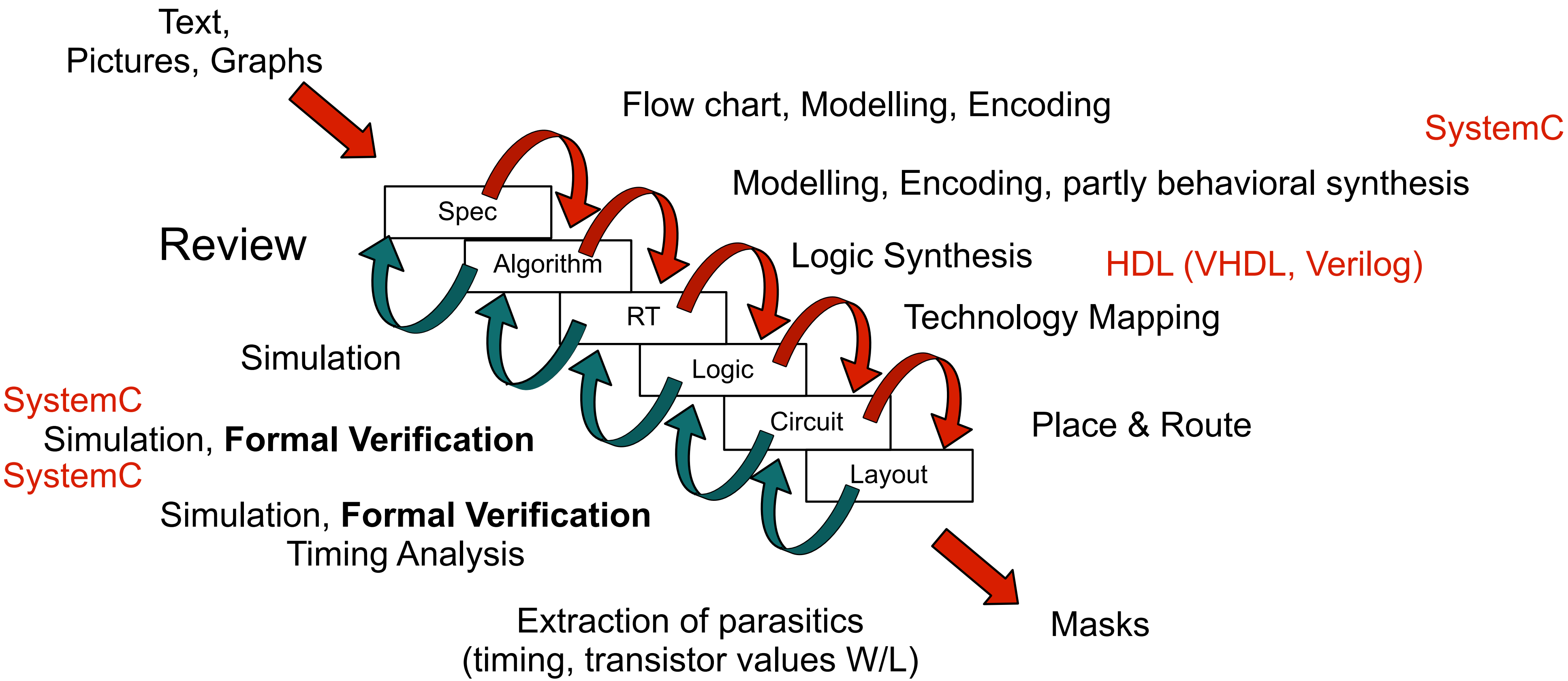
Agenda

1. Organisation of practice labs
2. CRC-4 example
3. Introduction to (formal) specification
4. Formal specification of a two-stage shift register

Agenda

1. Organisation of practice labs
2. CRC-4 example
3. Introduction to (formal) specification
4. Formal specification of a two-stage shift register

Waterfall Model



Text,
Pictures, Graphs

Flow chart, Modelling, Encoding

SystemC

Modelling, Encoding, partly behavioral synthesis

Spec

Algorithm

RT

Logic

Circuit

Layout

Review

Logic Synthesis

HDL (VHDL, Verilog)

Technology Mapping

Simulation

SystemC

Simulation, **Formal Verification**

SystemC

Place & Route

Simulation, **Formal Verification**
Timing Analysis

Extraction of parasitics
(timing, transistor values W/L)

Masks

System Specification

What is "System Specification"

- Describes system components
- Describes all functional and non-functional requirements
- Hierarchy: deducing requirements from the specifications of superordinate system components or the overall system specification, respectively
- Requirement resource for design and decomposition of system architecture
- Modification of a system component during a design step always requires to change the system specification
- Test specification defines tests to proof compliance of interfaces and requirements

http://h90761.serverkompetenz.net/v-modell-xt/Release-1.1/Dokumentation/html/

Why is specification so difficult?

- No comparable system engineered so far
- Requirements not comprehended correctly
- Requirements change during lifecycle (Development cycles)
- Complex interactions between services / components

Why is specification so difficult?

e.g. Software: Reasons for failure of software projects:

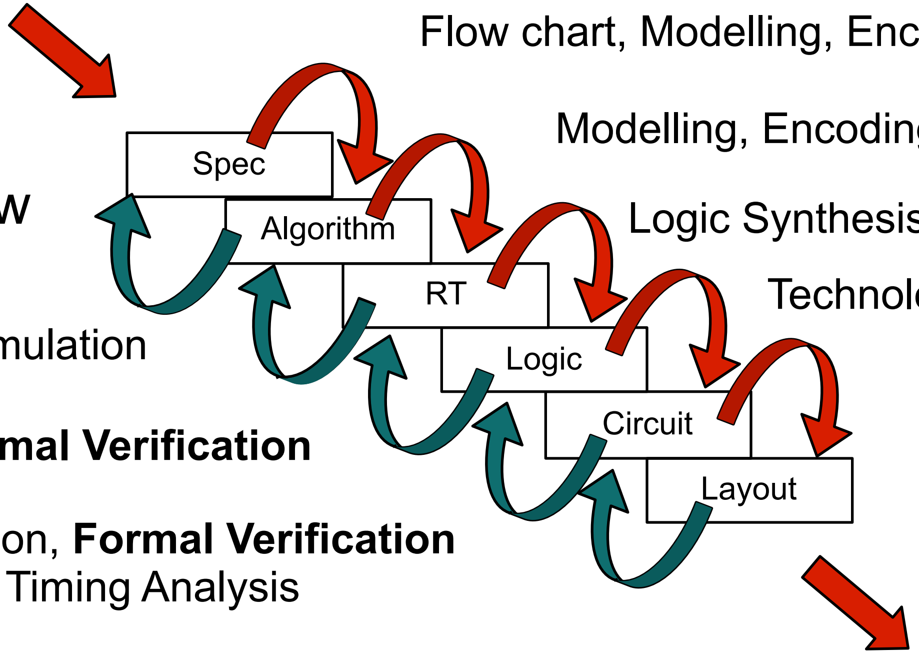
– incomplete requirements:	13,1%
– lacking inclusion of user:	12,4%
– lack of resources:	10,6%
– unrealistic expectations:	9,9%
– lack of management support:	9,3%
– changing requirements and specifications:	8,7%
– lack of planning:	8,1%
– system no longer needed:	7,5%

Relevance of early design stages

Hypotheses about the importance of good specification

- **Hypothesis 1:** The later an error is discovered, the more expensive it is to remove the error
- **Hypothesis 2:** Many errors remain undiscovered (or are discovered way too late)
- **Hypothesis 3:** Specifications contain many requirement errors
- **Hypothesis 4:** Requirement errors are typically: Incorrect facts, omissions, inconsistencies and ambiguities
- **Hypothesis 5:** Requirement errors can be discovered

Waterfall Model



Text,
Pictures, Graphs

Flow chart, Modelling, Encoding

SystemC

Modelling, Encoding, partly behavioral synthesis

Review

Logic Synthesis

HDL (VHDL, Verilog)

Technology Mapping

Simulation

SystemC

Simulation, **Formal Verification**

Place & Route

SystemC

Simulation, **Formal Verification**
Timing Analysis

Extraction of parasitics
(timing, transistor values W/L)

Masks

Relevance of early design stages

- Analysis of Navy A-7E Specification, 77% of faults are not typing errors
  - 49% incorrect facts
  - 31% omissions
  - 13% inconsistencies
  - 5% ambiguities

Main goal of formal specification methods

Verify specifications (automatically) as early as possible using complete formal methods

Characteristics of Requirement Specifications

- Correct
- Unambiguous
- Complete
- Verifiable
- Consistent
- Traced
- Tracable
- Design independent

Goal: Use formal methods during specification

Languages for Requirement Specifications

Native vs. formal notations

– native language

“*if the telephone earphone is picked up, then a dial tone sounds*”

+ expressive

+ understood by everybody involved in the design process (??)

- ambiguities

– formal notations and languages (using mathematical semantics)

$$(\forall t_1, t_2 \mid t_1 \leq t_2)(ringtone(t_1) \Rightarrow dialtone(t_2))$$

+ unambiguous

+ automatically analyzable to a large extent

- inexpressive (especially if automatically analyzable)

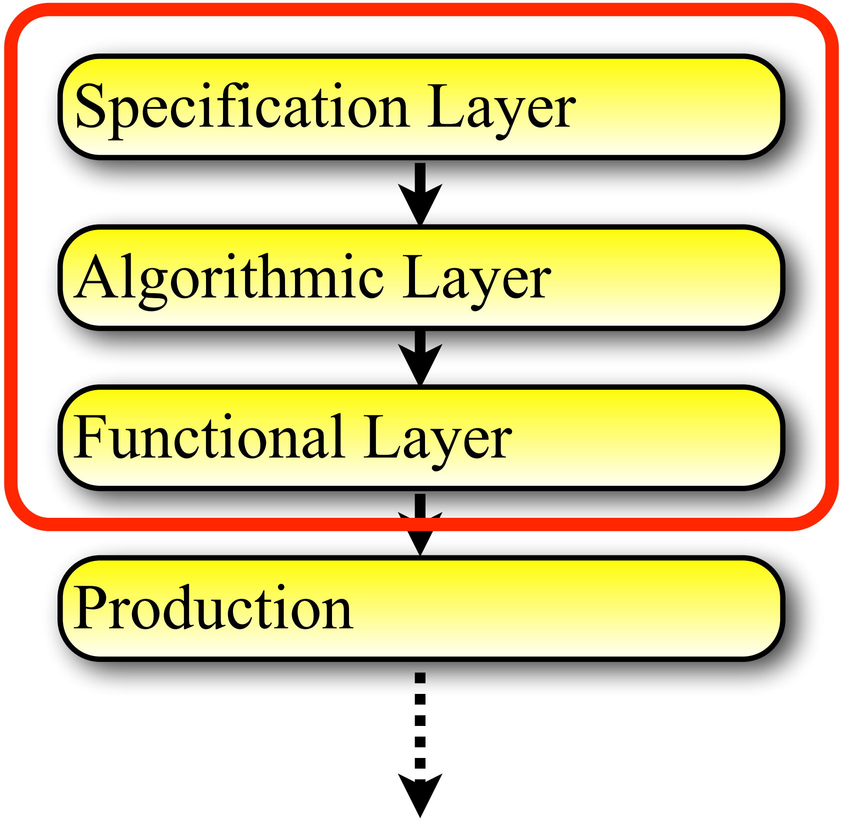
- only mastered by some of the people involved

Main problem

Bridge the gap between specification and implementation!

SpecScribe

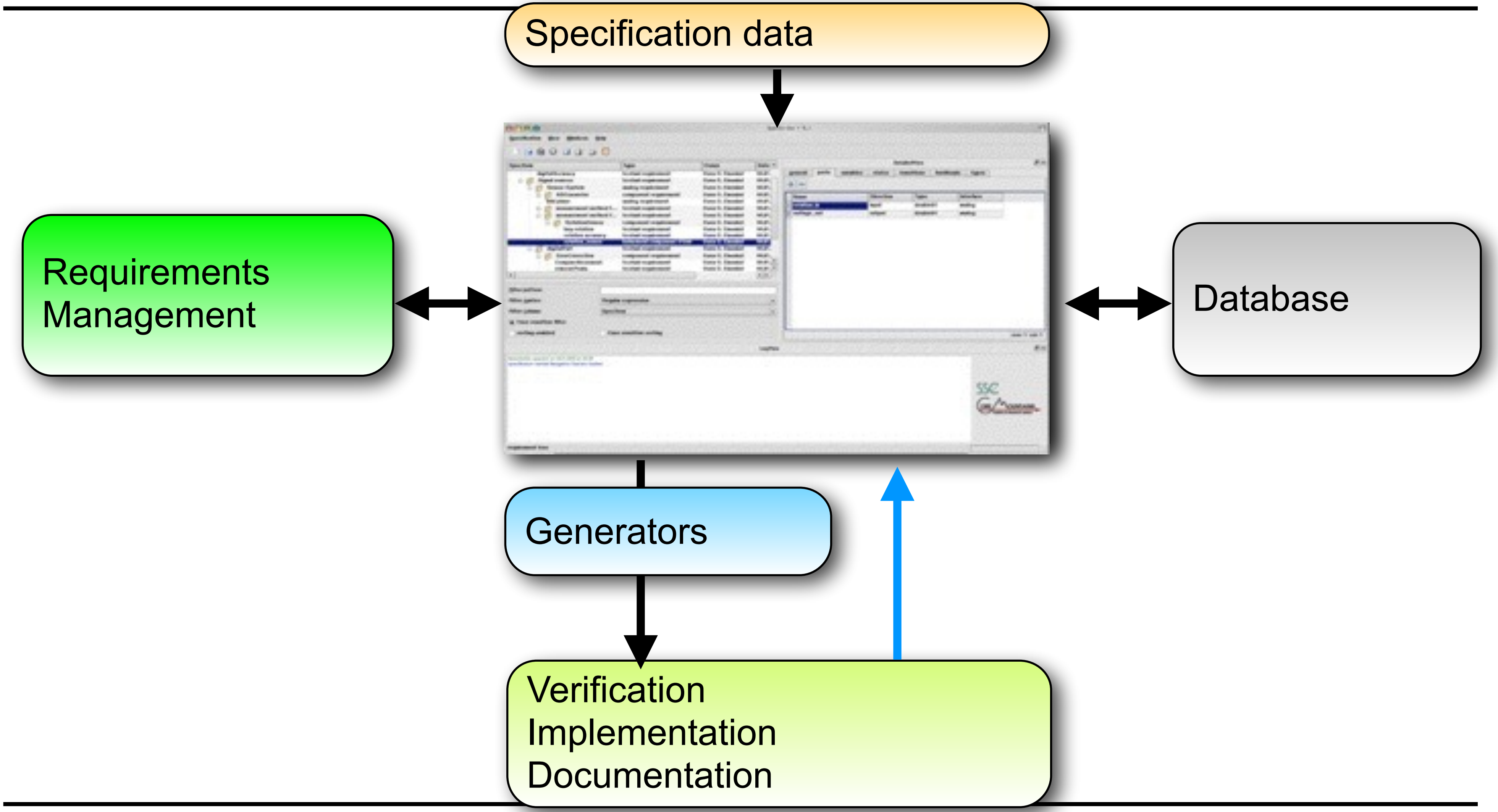
- Specification platform on **high abstraction level**
- Formal and/or informal specification
- Combine **industrially proven methods** with new approaches
- **Linkage** to existing design flows



Goal

- Combine several tools
  - Word processor (generation of documents)
  - Requirements Management Systems (tracking and tracing of requirements)
  - Tools for exploration, implementation, modelling
  - Tools for reliability and lifetime analyses

Overview



Specification data

Requirements Management

Database

Generators

Verification
Implementation
Documentation

Agenda

1. Organisation of practice labs
2. CRC-4 example
3. Introduction to (formal) specification
4. Formal specification of a two-stage shift register