System-Level Programming

2 Organization of the Lecture

Peter Wägemann

Lehrstuhl für Informatik 4
Systemsoftware

Friedrich-Alexander-Universität
Erlangen-Nürnberg (FAU)

Summer Term 2025

http://sys.cs.fau.de/lehre/ss25

Lecture

- Content and topics
  - Basic concepts of system-level programming
  - Introduction to the programming language C
    - differences compared to Python/Java
    - modular concept
    - pointers and pointer arithmetic
  - **“Bare-metal”** software development directly on hardware (ATmega µC)
    - mapping of storage $\leftrightarrow$ language constructs
    - **interrupts** & **concurrency**
  - Software development on **operating system (OS)**: Linux
    - operating system as a runtime environment for programs
    - abstractions and services of an operating system

Lecture

- 36 sections
  - slides on the web server `sys.cs.fau.de`
  - dates: see **semester overview**
  - → requirement for successful handling of the exercises
- Questions on the lecture
  - ideally ask **immediately**
  - in following lecture
- Q&A at the end of the term
- **Lecture does not replace the tutorials and hands-on exercises!**

Exercises

- Tutorial and hands-on exercise
  - Tutorial (Tafelübung)
    - distribution of and additional information for the programming assignments
    - joined development of an outline for the solution
    - discussion of the solution the subsequent week

Exercises

- Tutorial and hands-on exercise
  - Tutorial (Tafelübung)
    - distribution of and additional information for the programming assignments
    - joined development of an outline for the solution
    - discussion of the solution the subsequent week
  - Hands-on exercise (Rechnerübung)
    - independent programming
    - working with development tools
    - support from a exercise supervisor

Exercises

- Tutorial and hands-on exercise
  - Tutorial (Tafelübung)
    - distribution of and additional information for the programming assignments
    - joined development of an outline for the solution
    - discussion of the solution the subsequent week
  - Hands-on exercise (Rechnerübung)
    - independent programming
    - working with development tools
    - support from a exercise supervisor
- Appointments: choice of 8 + 1 groups
  - registration via Waffel (see website)
  - **seperate** group only for for SLP

Valid login for the Linux-CIP (computer lab) required for participation in exercises!

Exercises

WARNING!

There will be **no tutorials & exercises** during the winter term for students who failed the exam

WARNING!

Programming Assignments

- Practically apply lecture contents
  - **eight programming assignments** ↪ 2–10
  - including assignments in groups
- Solutions must be submitted in the SPiC-IDE
  - your solution is validated with the help of scripts
  - we manually correct the assignments give points and provide feedback
  - a solution will be presented by a student in one of the following tutorials **requires attendance!**

Programming Assignments

- Practically apply lecture contents
  - **eight programming assignments** ↪ 2–10
  - including assignments in groups
- Solutions must be submitted in the SPiC-IDE
  - your solution is validated with the help of scripts
  - we manually correct the assignments give points and provide feedback
  - a solution will be presented by a student in one of the following tutorials **requires attendance!**
- ★ Participation in the programming assignments is NOT mandatory;

↪ 2–9

however you can earn up to **10% extra points** for the exam!

Programming Assignments

- Practically apply lecture contents
  - **eight programming assignments** ↪ 2–10
  - including assignments in groups
- Solutions must be submitted in the SPiC-IDE
  - your solution is validated with the help of scripts
  - we manually correct the assignments give points and provide feedback
  - a solution will be presented by a student in one of the following tutorials **requires attendance!**
- ★ Participation in the programming assignments is NOT mandatory;

↪ 2–9

however you can earn up to **10% extra points** for the exam!

Plagiarising will lead to loosing **ALL extra points**.

Programming Assignments

- Practically apply lecture contents
  - **eight programming assignments** ↪ 2–10
  - including assignments in groups
- Solutions must be submitted in the SPiC-IDE
  - your solution is validated with the help of scripts
  - we manually correct the assignments give points and provide feedback
  - a solution will be presented by a student in one of the following tutorials **requires attendance!**
- ★ Participation in the programming assignments is NOT mandatory;

↪ 2–9

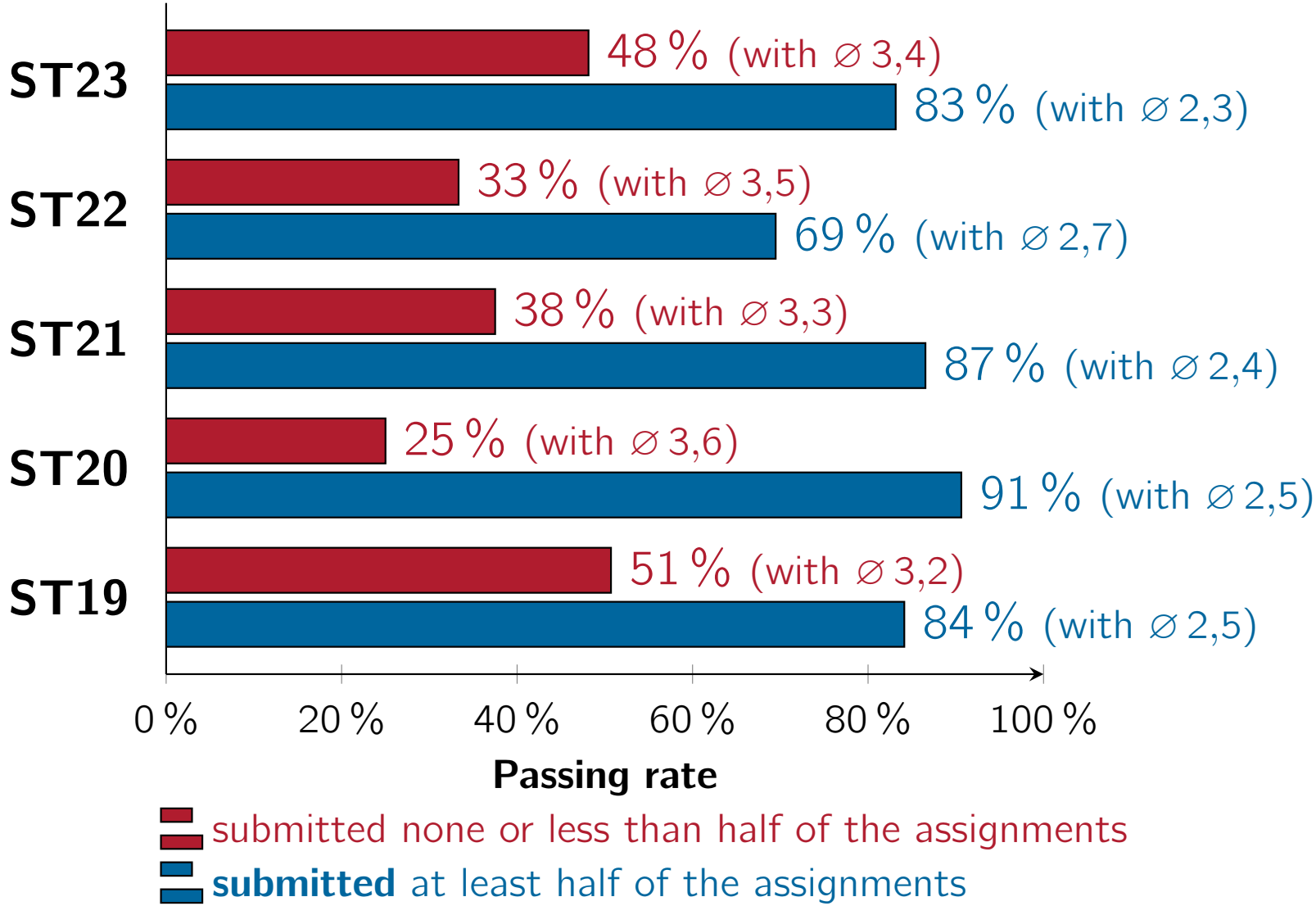
however you can earn up to **10% extra points** for the exam!

Plagiarising will lead to loosing **ALL extra points**.

Nonetheless, the participation in the assignments is **highly recommended!**

Passing Rate of the Exam (SPiC)

By activity of the participants in the programming assignments.

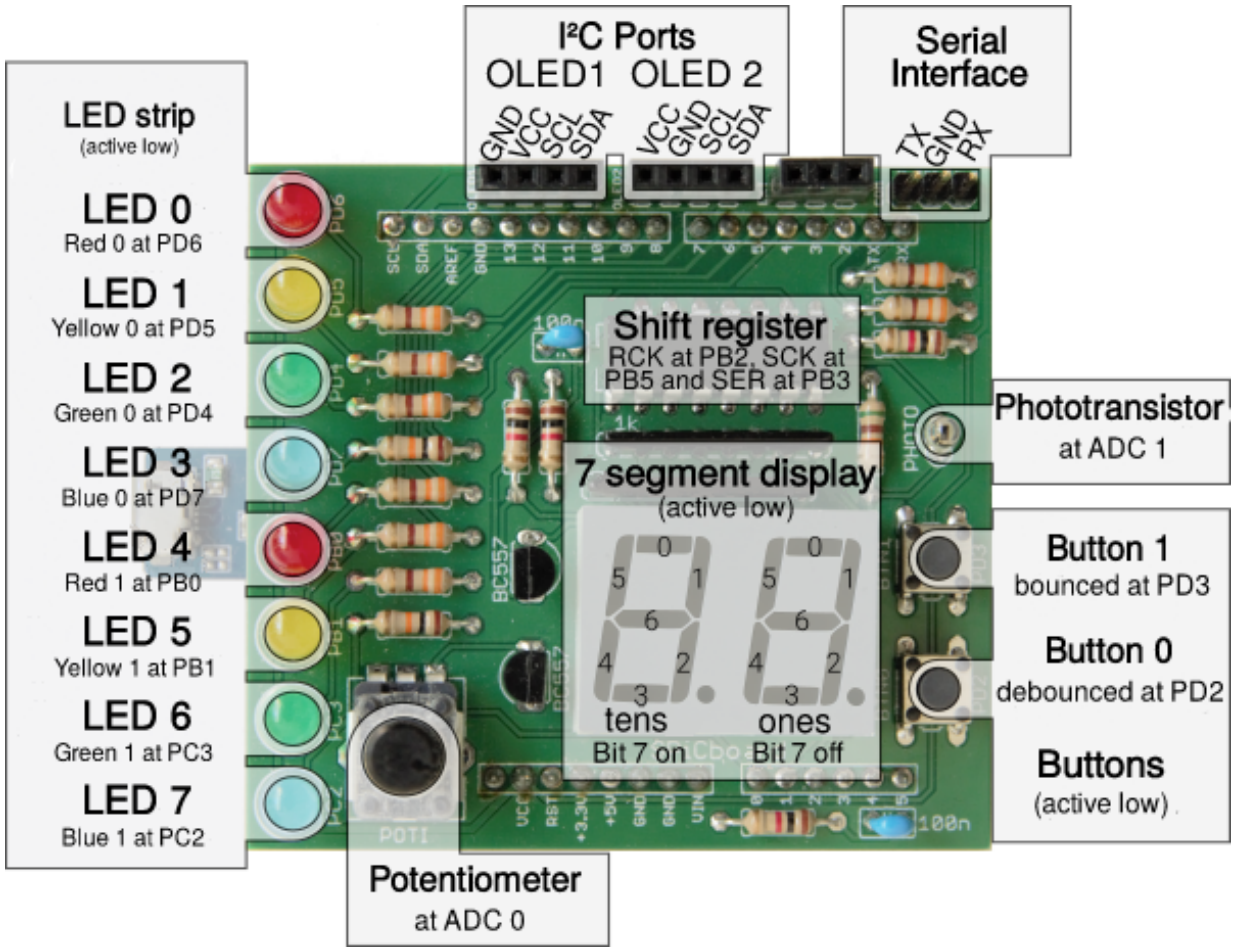


Exercise Platform: the SPiCboard

- ATmega328-µC
- USB port
- 8 LEDs
- 2 7-segment elements
- 2 buttons
- 1 potentiometer
- 1 photo sensor

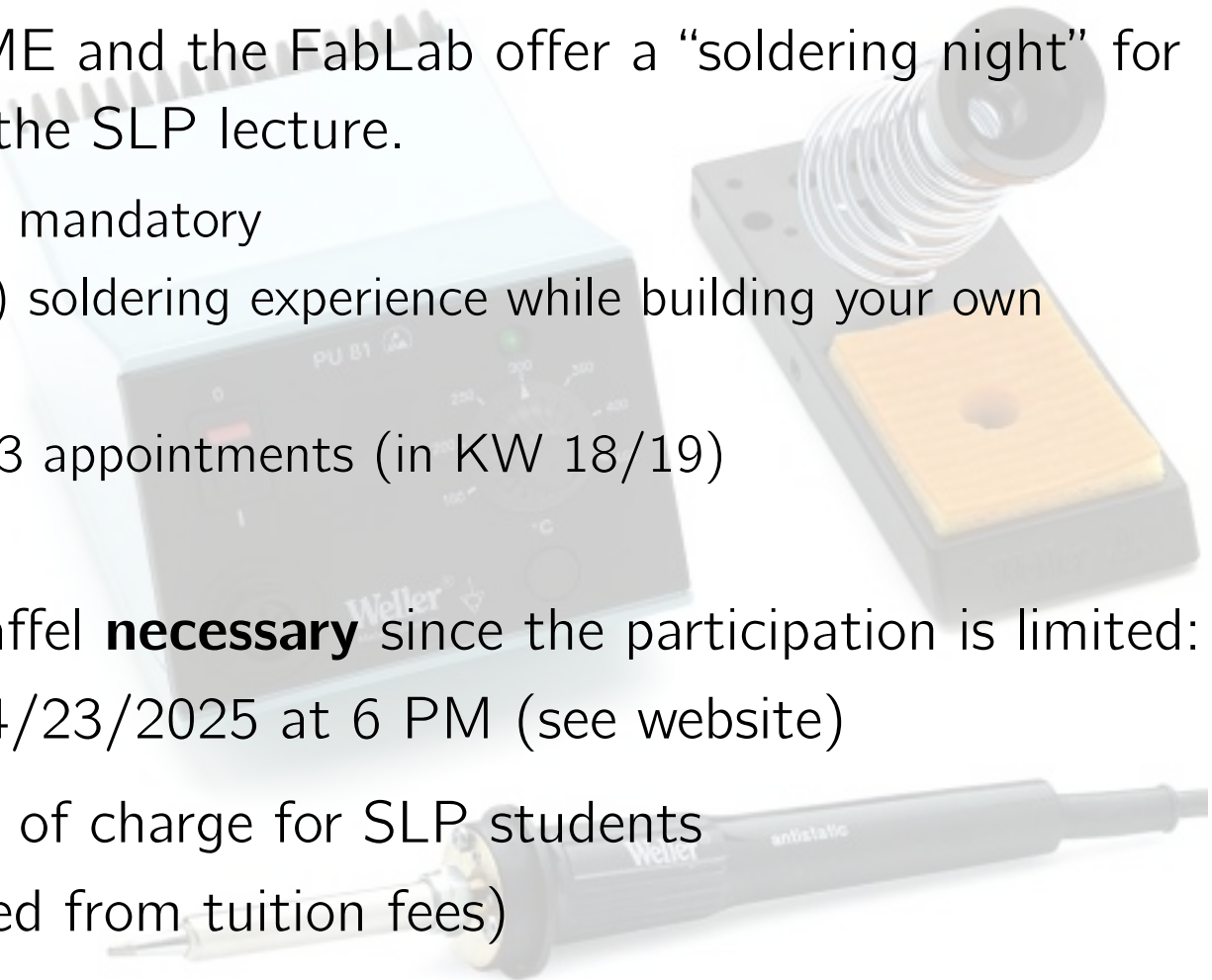
optional:

- OLED display



- can be borrowed during hands-on exercises
- better option: ↪ solder one by yourself!
- alternatively: development in simulator, which is integrated into the IDE

SPiCboard-Soldering Night



- The FSI EEI, FSI ME and the FabLab offer a “soldering night” for the participants of the SLP lecture.
  - participation is not mandatory
  - you can gain (first) soldering experience while building your own SPiCboard
  - there will be likely 3 appointments (in KW 18/19)
- Registration via Waffel **necessary** since the participation is limited: from Wednesday 04/23/2025 at 6 PM (see website)
- Participation is free of charge for SLP students (materials are funded from tuition fees)

The date you choose to register is binding!

Exam and Final Grade

- Exam (written test)
  - date: expected in early august
  - length: 90 min (SLP)
  - contents: questions on the lecture + programming exercise
- Exam grade $\mapsto$ final grade
  - (Usually) 50% of the exam's maximum possible points (EP) are necessary to pass.
  - Only if you passed, your grade can be improved by your bonus points from the programming exercises.
    - minimum: 20% of possible bonus points (BP)
    - bonus points get divided in equal parts to match the interval [50%;80%] of possible BP

    $\rightsquigarrow$ having 80%-100% of possible BP $\mapsto$ +10% of the maximum EP

Semester overview

KW	Mo	Di	Mi	Do	Fr	Themen
16	21.04.	22.04.	23.04. LEC1	24.04. LEC2	25.04.	Introduction, Organisation of the Lecture, Java/Python vs. C – Some Examples Software Layers and Abstraction, Language Overview, Basic Data Types
17	28.04. E1	29.04. LEC3	30.04.	01.05. Tag der Arbeit	02.05.	Operations and Expressions, Control Structures, Functions, Variables E1 (blink)
18	05.05. E2	06.05. LEC4	07.05.	08.05.	09.05.	Preprocessor, Program Structure and Modules, Pointers and Arrays E2 (snake)
19	12.05. E3	13.05. LEC5	14.05.	15.05.	16.05.	Pointers and Arrays, Composite Data Types, µC-System Architecture – Preface, µC-System Architecture – Processor, µC-System Architecture – Periphery E3 (led-modul)
20	19.05.	20.05. LEC6	21.05.	22.05.	23.05.	Interrupts, Interrupts – Example, Interrupts – Concurrency
21	26.05. E4	27.05. LEC7	28.05.	29.05. Christi Himmelfahrt	30.05.	Dynamic Allocation of Memory, Organisation of Memory, Organisation of Memory – Stack, Organisation of Memory – Summary E4 (spiel)
22	02.06.	03.06. LEC8	04.06.	05.06.	06.06.	Additions: Pointers, Additions – In-/Output, Additions – Error Handling, Operating Systems
23	09.06. Pfingstmontag E5	10.06. Pfingstdienstag	11.06.	12.06.	13.06.	E5 (ampel)
24	16.06.	17.06. LEC9	18.06.	19.06. Fronleichnam	20.06. Vorlesungsfrei	File Systems – Introduction, File Systems – UNIX
25	23.06. E6	24.06. LEC10	25.06.	26.06.	27.06.	Programs and Processes, Programs and Processes – UNIX, Signals E6 (concat)
26	30.06. E7	01.07. LEC11	02.07.	03.07.	04.07.	Multi Processors, Concurrent Threads, Concurrent Threads – praxis E7 (printdir)
27	07.07. E8	08.07. LEC12	09.07.	10.07.	11.07.	Exam Preparation E8 (mish)
28	14.07.	15.07. Q&A	16.07.	17.07.	18.07.	Question & Answer

Details: http://sys.cs.fau.de/lehre/ss25

Contributing Individuals, LS Informatik 4

Lecturer

Volkmar Sieh

Jürgen Kleinöder

Peter Wägemann

Organization of the tutorial and exercises

Eva Dengler

Contributing Individuals, LS Informatik 4 (continued)

Tutorial mentors

Eva Dengler

If there are Questions or Problems

- Take a look at the lecture or tutorial slides
- Consult the FAQ on our website
- Hands-on exercise
- Only if you still have no answer or in special cases, write an email to
  → all tutorial advisors i4spic@lists.cs.fau.de (content-related)
  → all academic staff (of this lecture) i4spic-orga@lists.cs.fau.de (organisational questions)

  Chatroom for students:
  `https://to.chat.fau.de/#/room/#spic:fau.de`