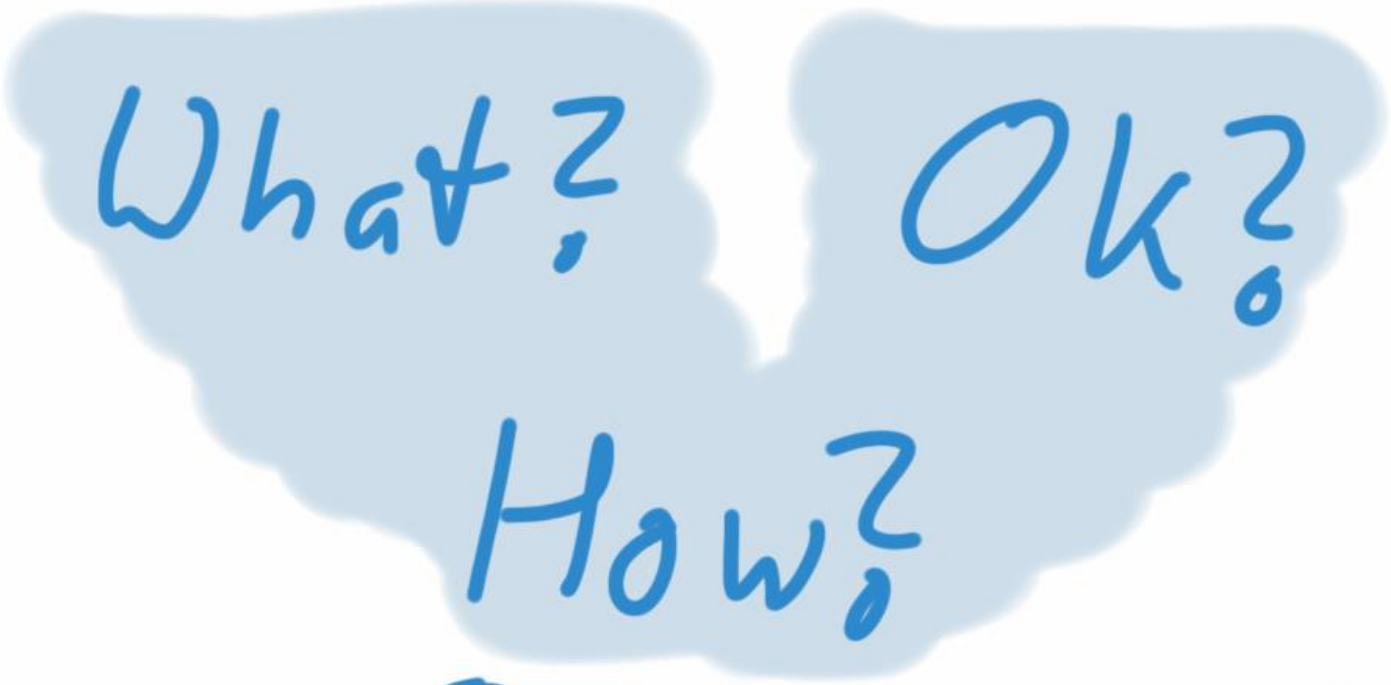
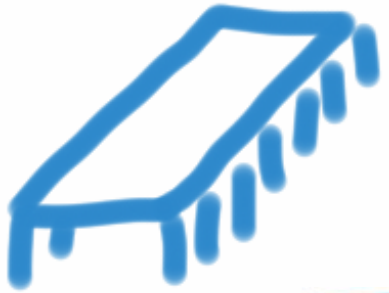
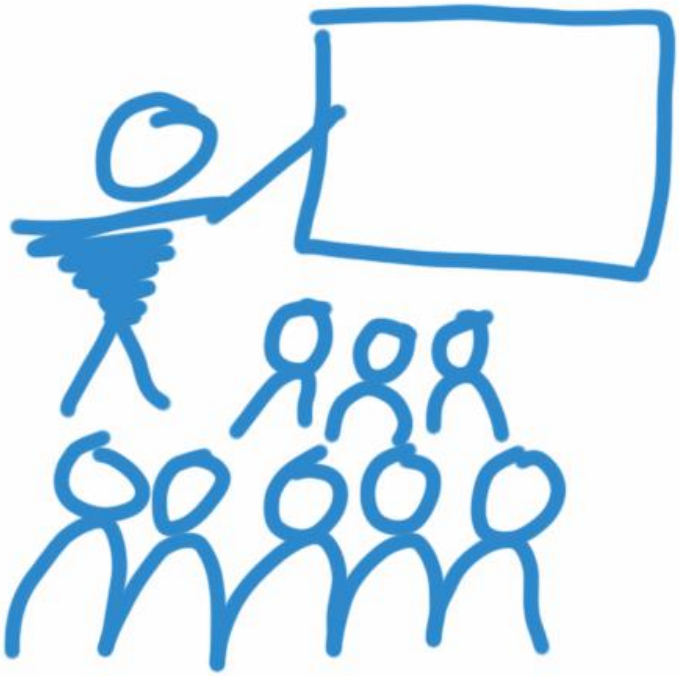
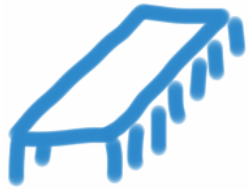
Learning Goals



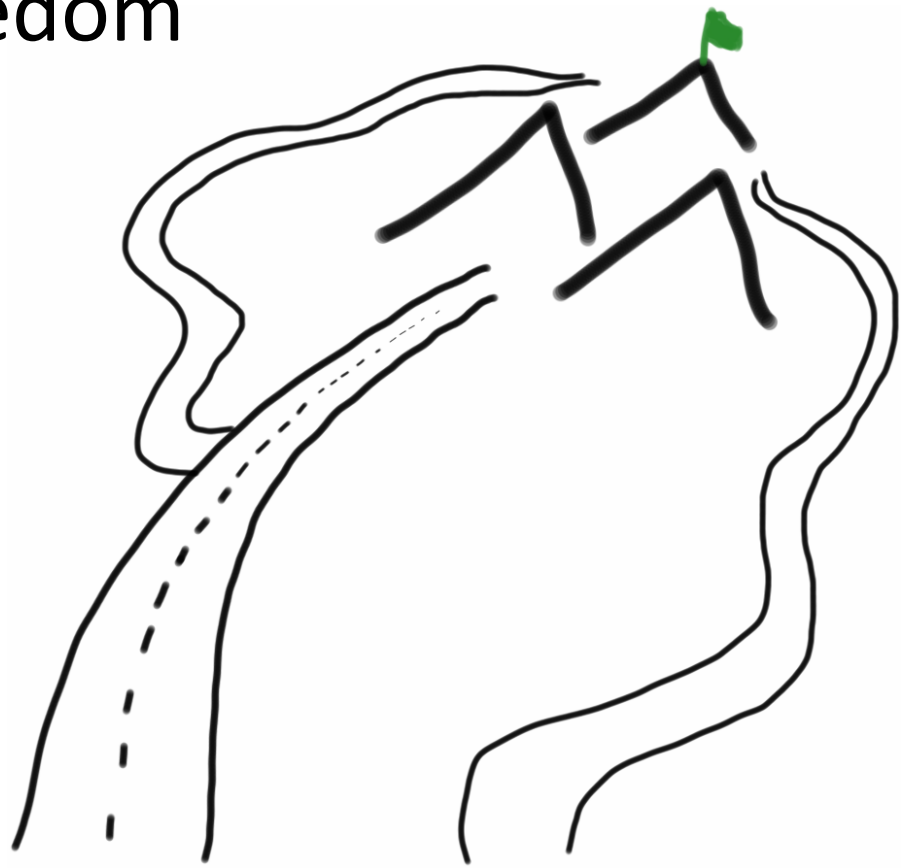




Learning Goals



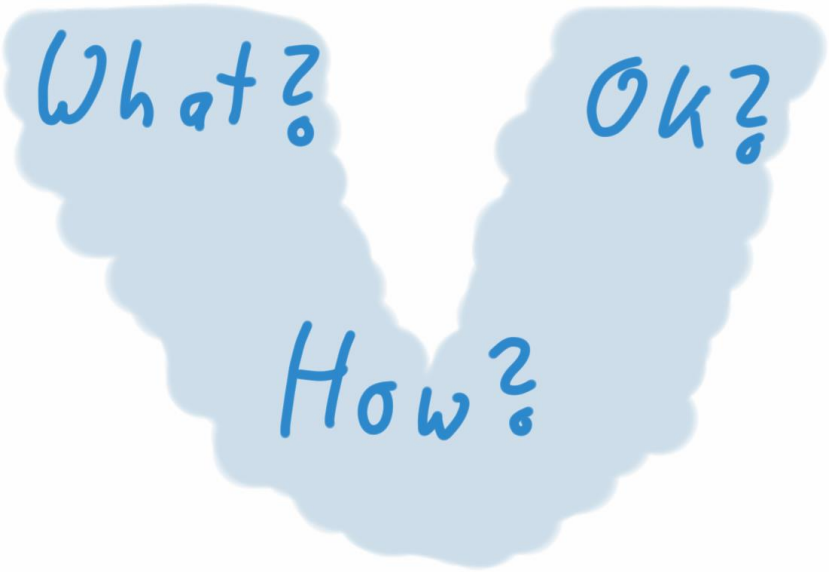
- Development for special hardware
- Efficient creation of good documentation
- Experiencing freedom in development
  - Results are demanded, not the way of fulfilling them



Content of the Course

- Foundation: Lectures about selected topics
  - Experts from industry will share their knowledge
- Your project
  - Development of a real-time system or comparable other topics
  - Paper about your project topic
  - Presentation of your project topic

Selected Topics



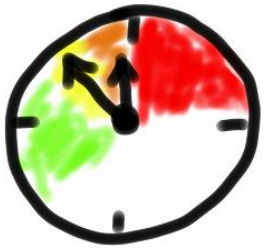
Scheduling

Operating System

Schedul-ability

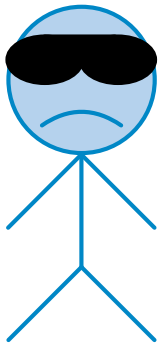
Multicore

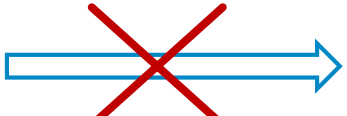
Real-time



Overview real-time systems

Resources





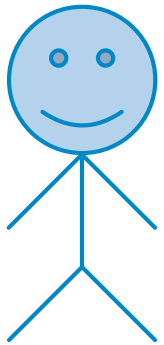
Security

System

Software

Hardware

Safety



Controller

Sensors

Actors

Project Details

- Different possibilities for your project topic – all are related to real-time systems
- Some ideas on the next two slides
- Any other (acceptable) topic related to real-time systems – the lecturer has to agree to the topic
- You might have to share equipment with other groups with the same topic

Project Details

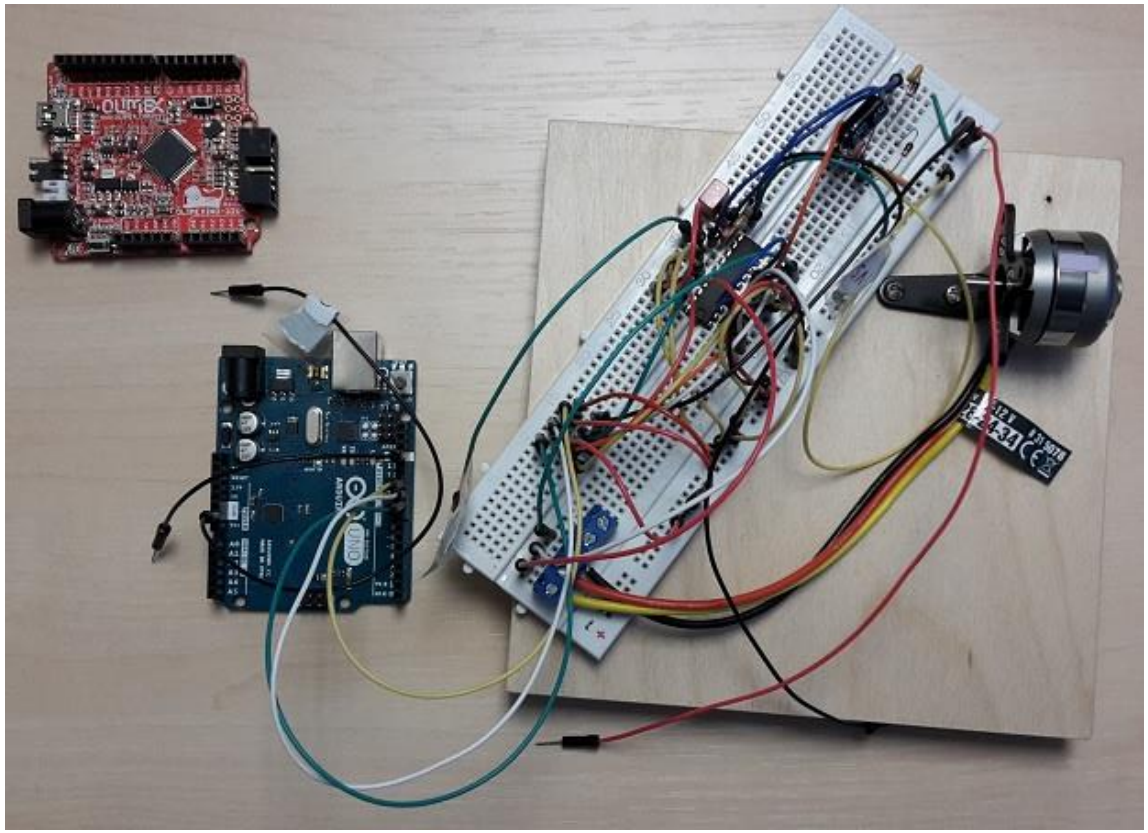
- Work with embedded hardware
  - Control of a BLDC motor (4 groups)
  - Profiling on the Arduino (4 groups)
  - Converter for RS-232 signals (2 groups)
- Multicore (4 groups)
- Simulators
  - Flight arrival and departure simulator (2 groups)
  - Antilock-Braking-System simulator (2 groups)

Project Details – Lego Mindstorms

- Autonomous vehicles (3 groups)
  - Only Autonomous Vehicle 2
- Sound detection (1 group)
- Obstacle detection (1 group)
- Balancing robots (3 groups)

Project Details – BLDC motor

- Hardware
  - BLDC motor
  - Arduino controller
  - Several electronic parts, especially a three phase motor driver IC

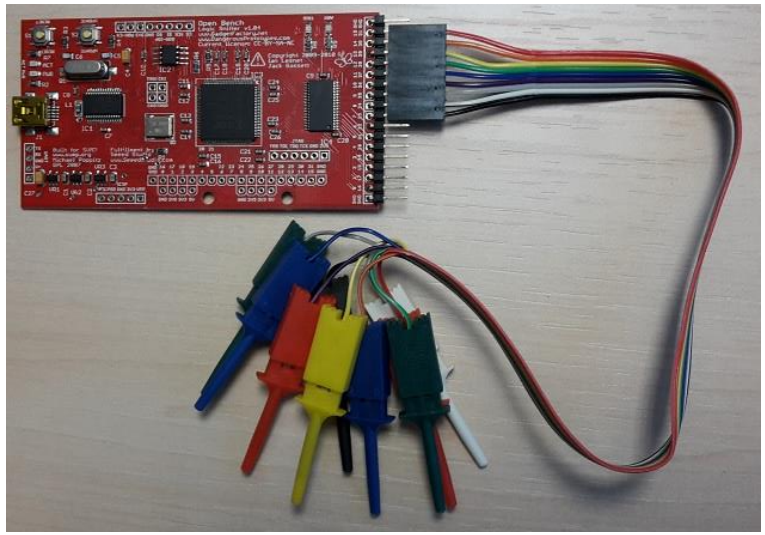


Project Details – BLDC motor

- The system shall control a BLDC motor in different speeds and in both directions.
- The speed shall be changed by two buttons for turning slower and faster.
- The direction shall be changed by another button.
  - The direction can be changed only if the motor is switched off. – Alternatively it is slowed down automatically, stops, and is accelerated again into the other direction.
- Another test program shall move the motor in a predefined way, e.g. half a turn clockwise, then wait one second, then three and a quarter turn counter-clockwise, …
- Introduction to electrical motors on 6.11.2017

Project Details – Profiling

- Hardware
  - Arduino Controller
  - ARM Cortex-M3 development board with touchscreen
  - Logic Analyzer



Project Details – Profiling

- Sometimes profiling is useful during development
- How can profiling be done on the Arduino?
- Identify and try out at least two completely different approaches
- Available hardware
  - Arduino controller (with display, if necessary)
  - Logic analyzer
- Write a cyclic program with some calculations and functions and apply your profiling approaches to analyze it

Project Details – Profiling

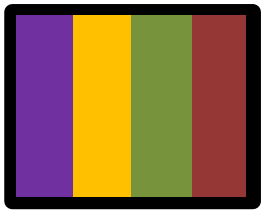
- Optimize the given program ProfilingProgram.py for the Arduino
  - Translate it from python first
    - Find appropriate constructs with similar runtime
  - Analyze and optimize it in the second step using
    - your profiling approaches
    - your knowledge about simplification of algorithms
  - Test it

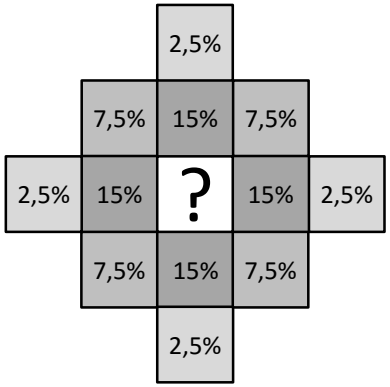
Project Details – Multicore

- The focus of that task is on multicore
  - This can also be done in a non-real-time environment
- Hardware: Raspberry Pi 3 Model B
- Software: Parallel programs from different domains
- Select an appropriate programming environment
  - Programming language
  - Multicore support
  - Runtime measuring

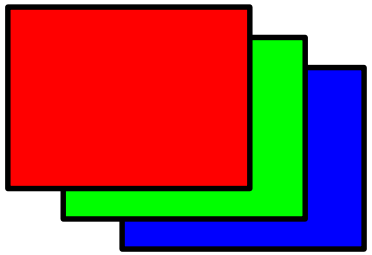
Project Details – Multicore

- Image-Processing ideas
  - Each core processes a part of the picture
    - Reformat picture by averaging the new color based on the colors of the neighbors?
      - What about unavailability of pixels at the edges?





  - Each core processes one aspect for the whole picture
    - Calculate the average RGB-values



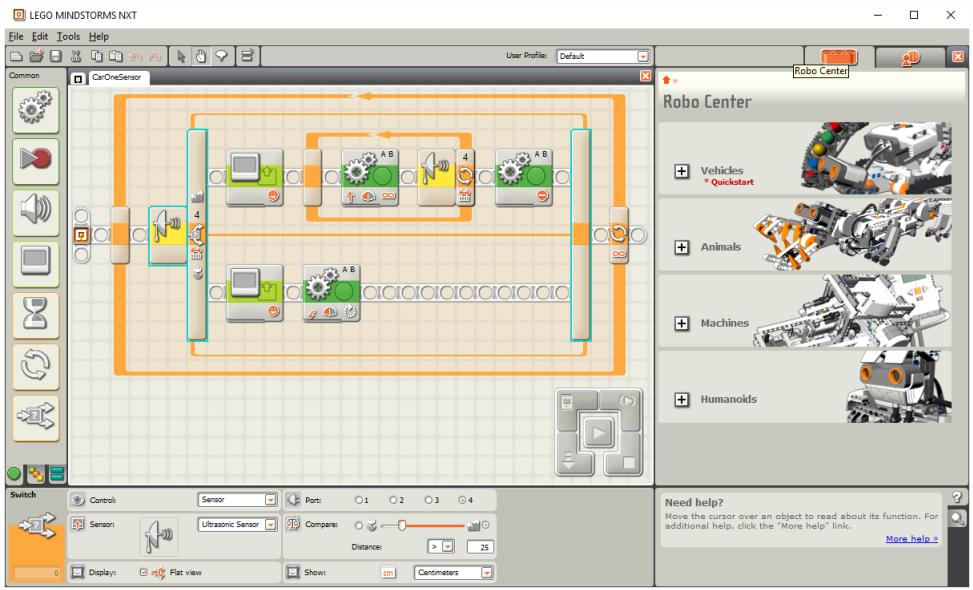
- Further computer science or mathematical problems
- ProfilingProgram.py

Project Details – Multicore

- The task in detail
  - Compare the runtime of (at least three) programs from three different domains when using 1, 2, 3, and 4 cores in each experiment
    - Explain the kind of parallelization
    - Explain the measured values
    - Can race conditions happen in the respective case?
  - Implement (at least two) synchronization mechanisms
    - They certainly can be used by your programs
    - How to test or prove the correctness?
  - Implement a simple runtime measurement and use it

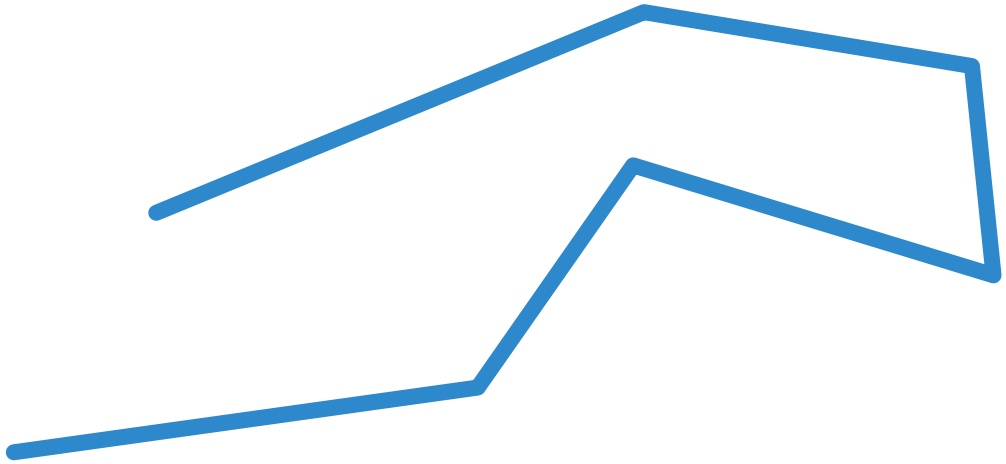
Project Details – Autonomous vehicles with Lego Mindstorms

- These vehicles drive around and avoid collisions
- Programming language: LEGO NXT



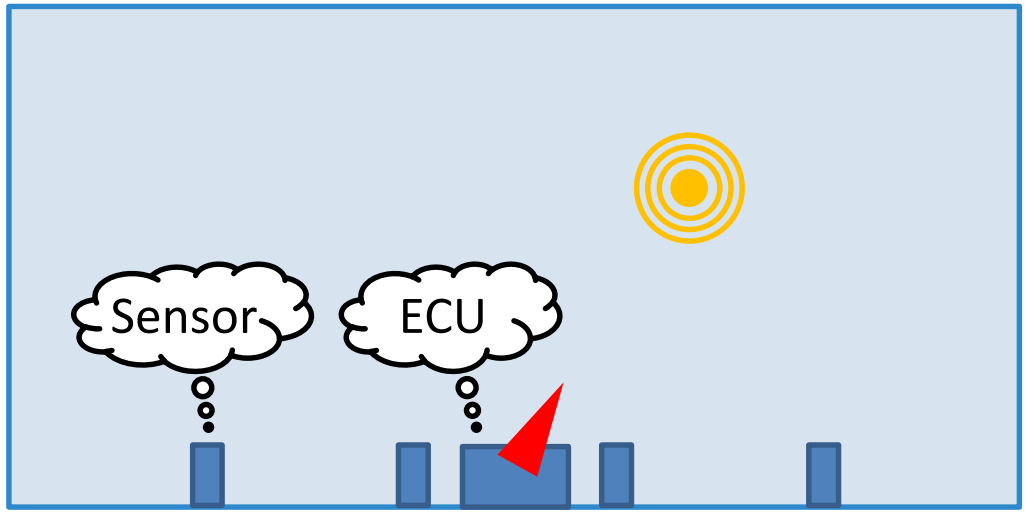
Project Details – Mindstorms Autonomous Vehicle 2

- Vehicle that follows a colored line or an illuminated line on the ground
  - Autonomous steering is required
- The vehicle shall be as fast as possible
- Sensors: Two light or RGB sensors



Project Details – Mindstorms Sound Detection

- The system shall point into the direction with the highest volume within an area of 180 degrees
- Sensors
  - 4 Sound sensors – calibration of sensors necessary?

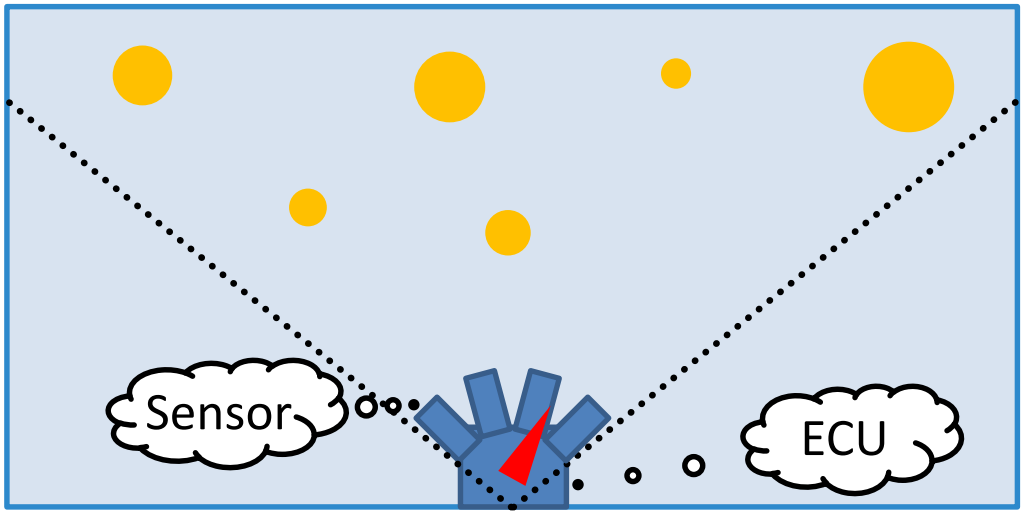


Project Details – Mindstorms Sound Detection

- Analyze the quality of the sensors – this is necessary for the implementation
  - Detection range
  - Detection error
  - Differences of the sensors
- Think about how to test this
  - Realize the test

Project Details – Mindstorms Obstacle Detection

- The system shall point into the direction with the largest distance to an obstacle within an area of 120 degrees
- Sensors
  - 4 ultrasonic sensors – calibration of sensors necessary?

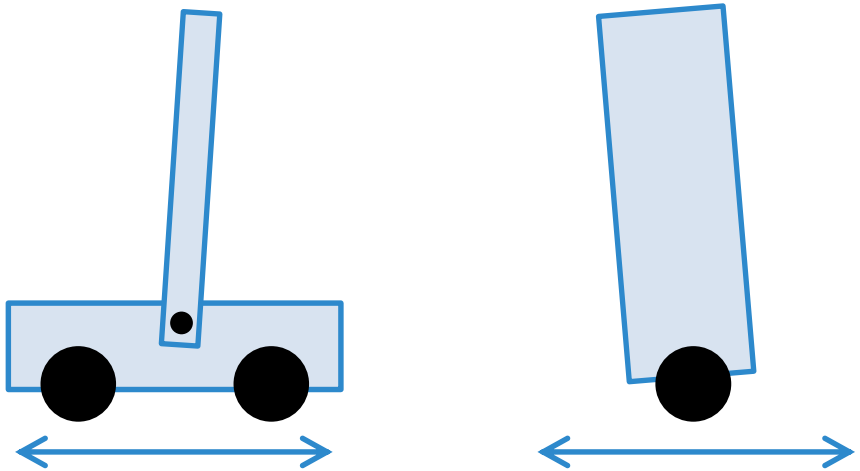


Project Details – Mindstorms Obstacle Detection

- Analyze the quality of the sensors – this is necessary for the implementation
  - Detection range
  - Detection error
  - Differences of the sensors
- Think about how to test this
  - Realize the test

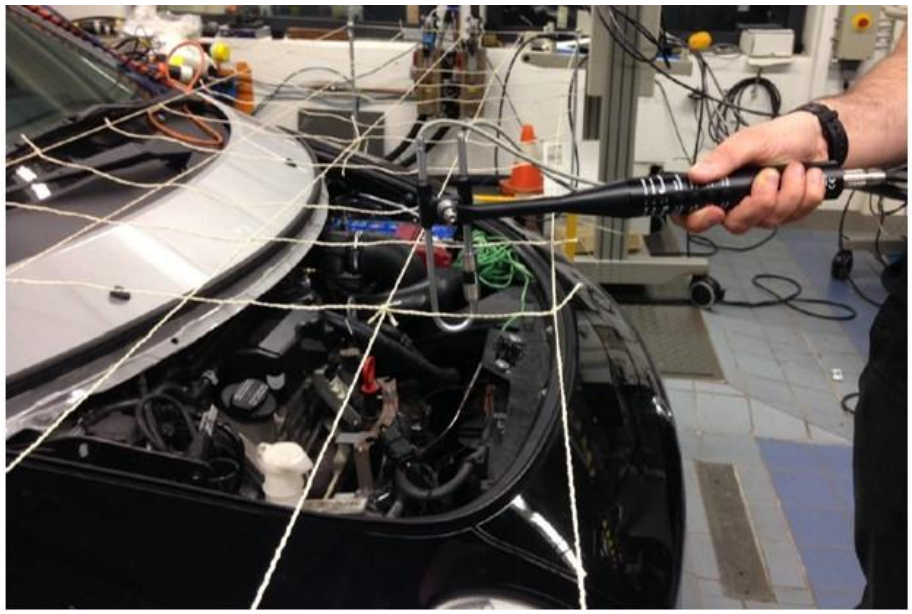
Project Details – Mindstorms Balancing Robots

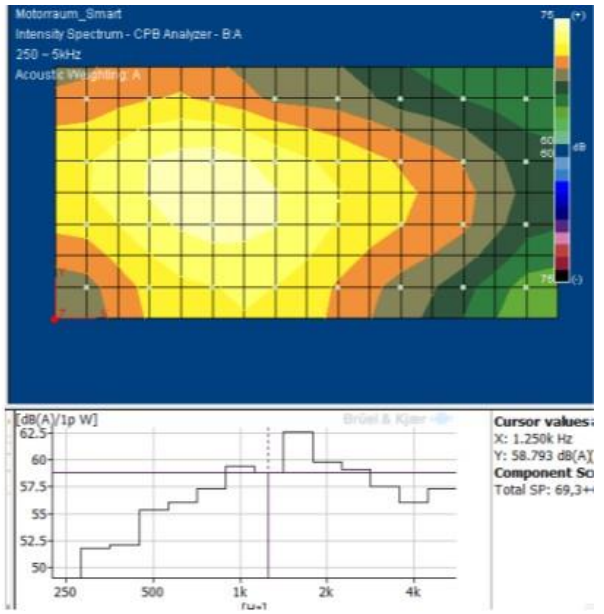
- Create some moving object balancing around one axis, e.g.
  - Sketches below (balancing by wheel rotation)
    - Let the vehicle move and stop with keeping the object in balance
  - Bike (balancing by steering)
- Hardware
  - Motor(s)
  - Gyro sensor



Project Details – Converter for RS-232 Real time noise analyses command transmitter – by Alexander Pfaff

Current situation: In the laboratory for automotive engineering there is a sound intensity probe to create sound intensity maps. Once placed, the sound intensity at that point can be measured (middle image). For this purpose, a remote control is provided for facilitating the handling of the probe, whereby measurements can be started and stopped.

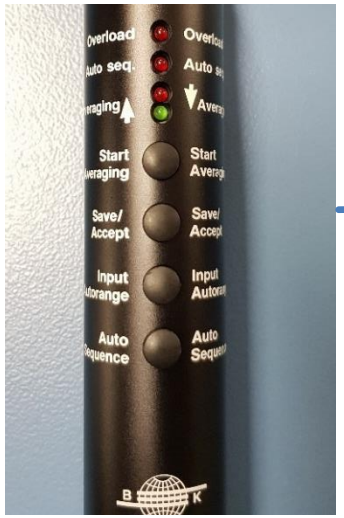


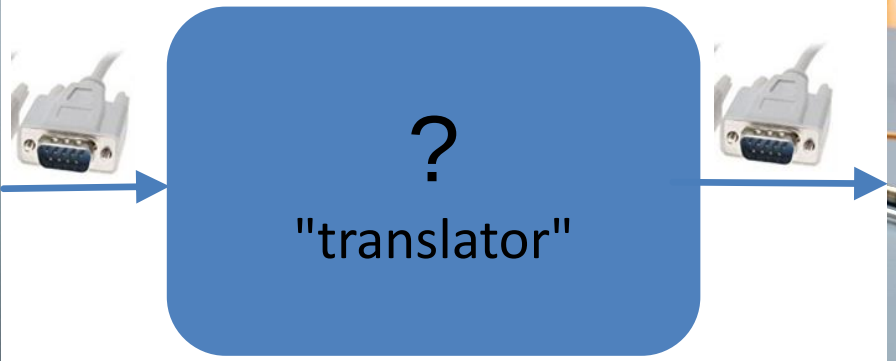


Project Details – Converter for RS-232
Real time noise analyses command transmitter
– by Alexander Pfaff

Problem: The commands of the remote control are no longer supported.

Task: Design and construction of a "translator", which enables the remote control function. The system shall finish sending the respective output-command after having detected the input-command within 15 milliseconds.





Project Details – Flight Arrival and Departure Simulator

- Write a simulation for arrivals and departures
  - Fair schedule decreasing the overall waiting time
- Parameters
  - Number of runways: 1, 2, or 3
  - Number, frequency, randomness, … of arriving and departing planes
  - Scheduler (different algorithms?)
- Hardware: PC
  - Graphical user interface for setting the parameters
  - Graphical output

Project Details – Antilock-Braking-System Simulator

- Write a simulation for one wheel of a vehicle
  - When full braking, the brake pressure shall be as high as possible without locking of the wheel, i.e.
    - when locking is detected, then the brake pressure shall be decreased until the wheel is not locking any more
    - when no locking is detected, the brake pressure shall be increased until the wheel is locking

Project Details – Antilock-Braking-System Simulator

- The vehicle shall brake on a simulated road with changing frictional coefficient
  - Test it with different roads and measure the braking distance depending on the road
- Hardware: PC
  - Graphical user interface for setting the parameters
    - Algorithm
    - Road
  - Graphical output
    - Graph with friction data of the road, speed, and braking pressure and distances

Project Details – Other Topic?

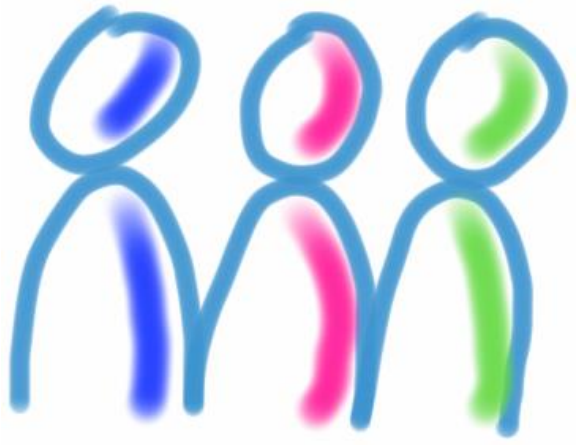
- Perhaps you already have an idea … ???
- The topic has to be related to real-time systems
- The topic must not be the same as other topic from preceding semesters
- The lecturer has to agree to the topic

Work Products in Moodle

- Which ones?
  - Handouts by the lecturers
  - Deliverables by the participants
    - Language: English
- Where?
  - https://moodle.frankfurt-university.de/
  - Course: Advanced Real-Time Systems – Winter 2017/18
  - Access code: G4m7Ta

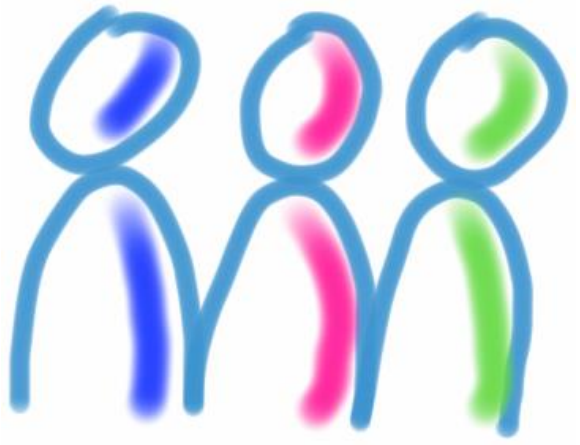
Every participant needs to register as early as possible

Grading

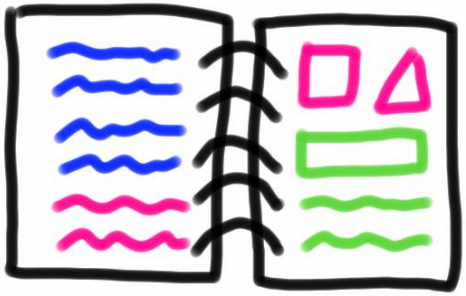


- One personal grade
  for every participant
- Each of the items counts as much as the others
  - Quality of written paper
  - Working project – based on the project type:
    - Running software(-system)
    - Working schedulability analysis
  - Quality of presentation

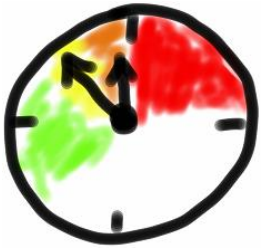
Grading



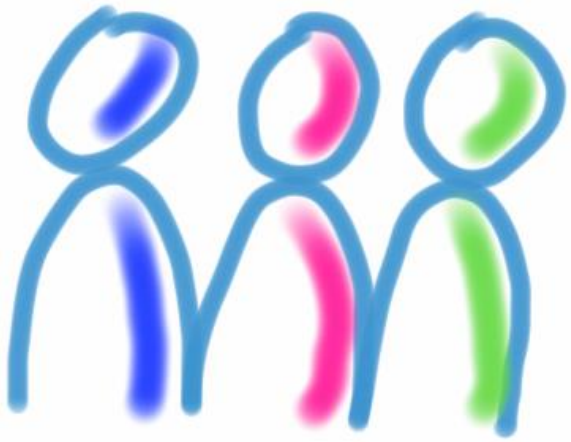
- Paper
  - Delivered by every participant
- Project
  - Delivered by the team
  - It has to be obvious who contributed to which part, e.g. add a simple file that describes who contributed to which part or mark the parts



- Presentation
  - Every participant presents one part in the team presentation



Grading

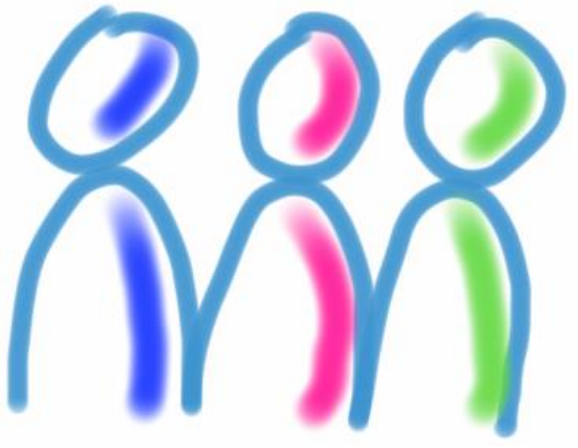


- Work products have to be delivered in time
- Language: English
- Work products have to be understandable
- Work products have to fulfill scientific standards
- Plagiarism is an attempt of deception
  - No copies of parts of relevant size from other resources (also not from other team members!)
  - Correct citations of used work products
- Deviations from these points influence the grade

Plagiarism

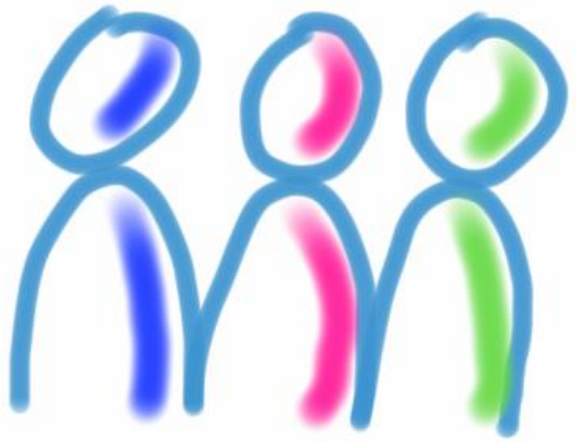

NO
Plagiarism!

Grading



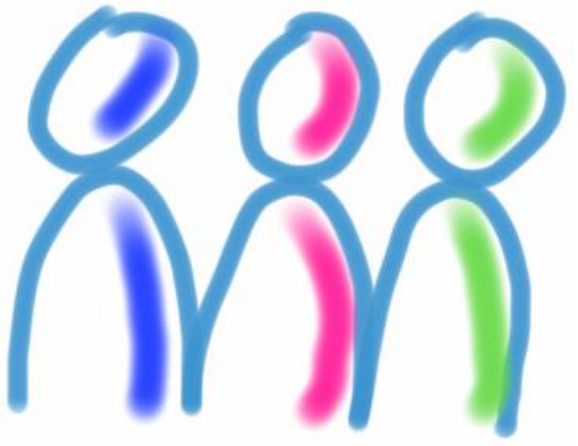
- Paper
  - Needs to include your name, matriculation number, group number and the name of the course
  - Well structured, e.g.
    - Abstract
    - Citation of necessary references
- Deviations from these points influence the grade

Grading



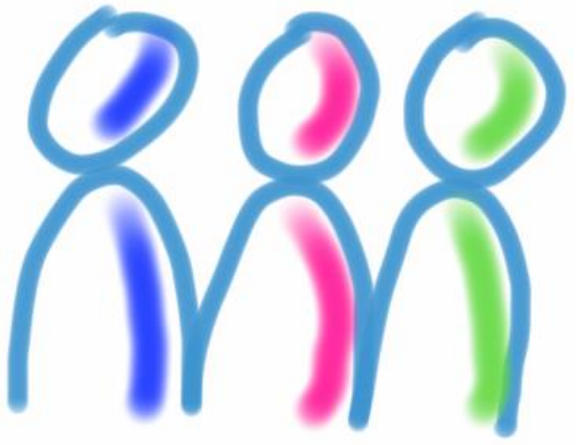
- Paper
  - Author's (not the group's) contribution to the project
  - Embed your contribution into a meaningful context, depending on the contribution, e.g.
    - Theoretical foundation
    - Software design
    - Testing
    - Visualization
    - Examples
- Deviations from these points influence the grade

Grading



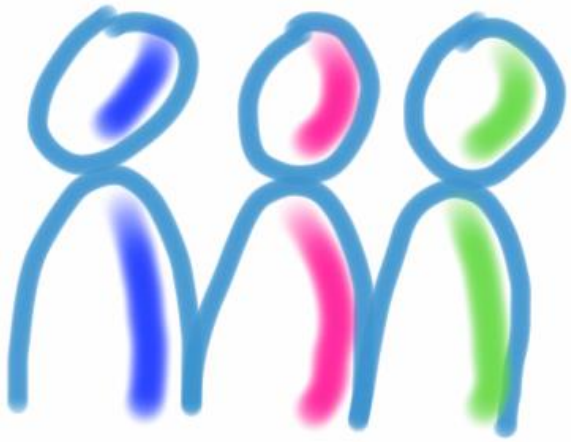
- Paper
  - Two-weekly reports to be included in the paper
    - Author's contributions to the project
    - Two-week-periods starting after the topic has been selected
  - Task for the group: Split topics for the papers such that every paper contains different aspects related to the project
- Deviations from these points influence the grade

Grading



- Paper
  - Maximum of 6 pages
    - Quality, not quantity!
    - If there are many figures and/or large tables in the paper, then up to 8 pages is acceptable
  - Font size: 12 pt
  - One or two columns
  - Format: pdf
- Deviations from these points influence the grade

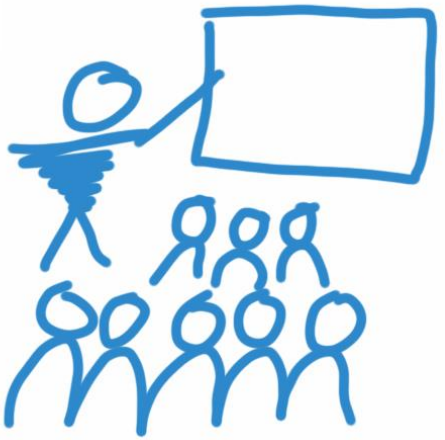
Grading



- Delivered project has to fulfill its requirements
  - Implementation of the theoretical foundation in the papers
- Project presentation by the group
  - Based on the content of the papers
  - Includes practical presentation and theoretical foundation
  - Each participant: 5 to 7 minutes
    - Task for the group: Split topics for the presentation such that every person talks about different aspects related to the project
- Naming convention for deliverables to be used
- Deviations from these points influence the grade

Important Dates

- Selection of your topic: 13.11.2017, 8:30 (or earlier)
- Your presentation:
  29.01.2018 or 05.02.2018
  - Date and time planning during the presence time



- Delivery dates
  - Work products of your project: 09.02.2018
  - Your paper: 09.02.2018

Presence Dates and Time in Room 1-250

- Presence dates (Monday)
  - 23.10.2017
  - 30.10.2017
  - 06.11.2017
  - 13.11.2017
  - 20.11.2017
  - 27.11.2017
  - 04.12.2017
  - 11.12.2017
  - 18.12.2017
  - 15.01.2018
  - 22.01.2018
  - 29.01.2018
  - 05.02.2018

Time for working on your project topic in the presence time.

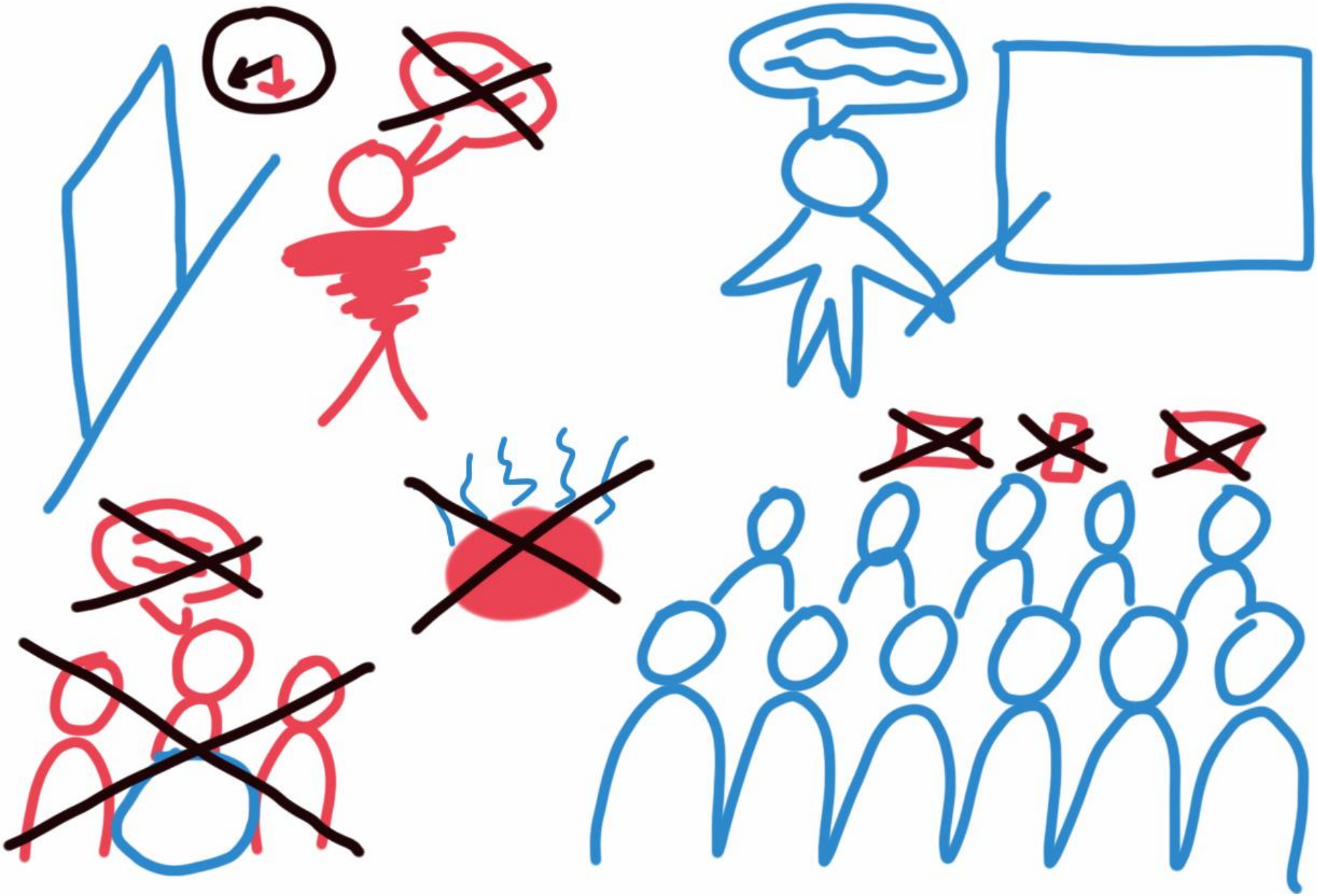
Lectures about selected topics on
- 06.11.2017
- 13.11.2017
- 04.12.2017
- 11.12.2017

Presentation of your results

- Presence time
  - 8:15 – 9:45
  - 10:00 – 11:30

Group A / B – two slides

Rules of the Game



Next Step: Grouping

- Form teams based on interests for the topics
  - The whole group works on one project
- Two or three people in a team
  - Not one, not four!
- Consensus of all participants
- Fill in your names etc. in the forms