LUNDS UNIVERSITET

FRTN40 Project in Automatic Control

Kristian Soltesz, Martina Maggio
Charlotta Johnsson, Anton Cervin

FRTN40 Project in Automatic Control

- Team effort
  - Task division
  - Communication
  - Integration
- Course knowledge
  - Modelling
  - Identification
  - Control design
  - Implementation
- Practical experience
  - Design methodology
  - Debugging and verification
  - General engineering skill development

Learning outcomes

Ctrl+C Ctrl+V from course syllabus (*kursplan*)

Knowledge and understanding

For a passing grade the student must

- have improved his/her basic knowledge and skills of control
- have developed new knowledge and skills within the area of the project

Competences and skills

For a passing grade the student must

- be able to perform several of the phases in a typical control project: modeling, identification, analysis, synthesis, and computer implementation
- present the results in both written and oral form
- show ability for teamwork and collaboration in project form

Course staff

Teachers

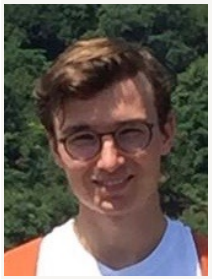
Kristian

Martina

Lotta

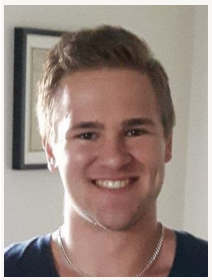
Anton

Supervisors



Marcus

Marcus



Nils

Tech & Admin

Pontus

Leif

Anders

Mika

Responsibilities

Overall	Kristian Soltesz
Your group	Supervisor
Lab, tools, materials	Pontus Andersson
Computer systems	Anders Nilsson
LaTeX template (written report)	Leif Andersson

If something is missing, doesn't work etc.
– contact your supervisor ASAP, don't wait.

Communicataion

All course info is available here:

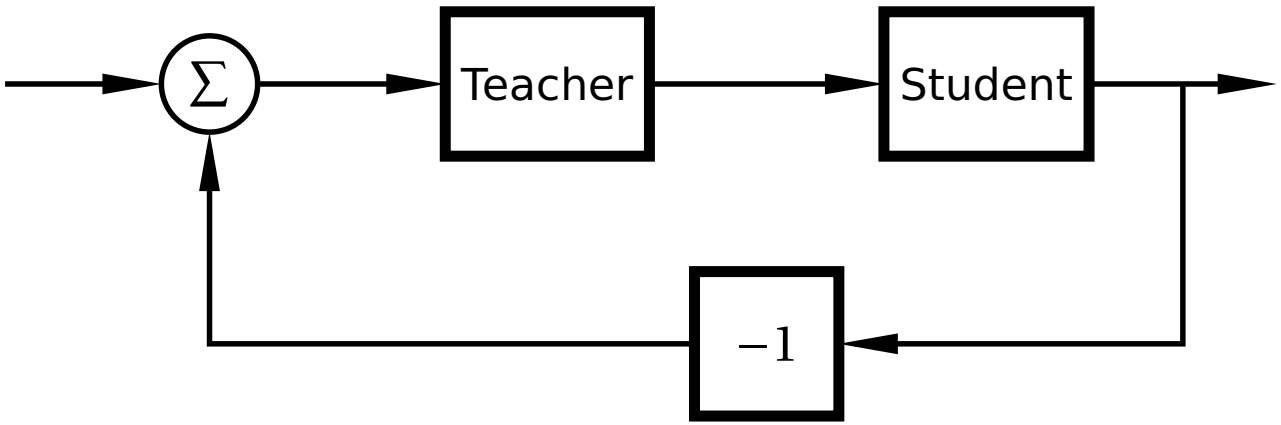
https://www.control.lth.se/FRTN40

- Your e-mail address (the one registered in LUCAT) has been added to a course mailing list, to which we will send updates. **Anyone hasn't received welcome e-mail?**
- Scheduled course events and supervision meetings (more on this soon)

Help us improve

We appreciate (negative) feedback!

- Anonymous online form
- Confidential talk with student representatives (*kursombud*)
- CEQ final evaluation
- ...and of course you are welcome to talk with the staff



Student representatives (*kursombud*)

Assignments

- Represent *all* course participants
- Help assemble the CEQ evaluation report
- More info online (Swedish only)

Two volunteers?

Important dates

Today	Intro meeting (this), git tutorial
Oct 23	Project and groups wish list
Nov 5	LaTeXtutorial
Nov 9	Project plan deadline
Nov 19	Feedback seminar 1
Dec 5	Feedback seminar 2 report deadline
Dec 10	Feedback seminar 2
Dec 20	Final report deadline
Jan 7	Report review deadline
TBA	Project demonstrations
Jan 18	Print-ready report deadline
Jan 25	LADOK registration deadline

Times and locations posted on course homepage.

Demo day

Preliminaries:

- Half day of Jan 10 (Thu) or Jan 11 (Fri).
- This is scheduled by LTH as exam week.
- There are no other activities centrally scheduled Jan 10–11.

Let’s have a vote:

- Which day?
- Before or after lunch?

Project plan

- An overview of the project
- Description of key parts of the project, including
  - equipment and materials
  - modelling
  - design (hardware, software, control principles)
  - implementation
- A decomposition of the project into subtasks and a suggested allocation of the project resources to key tasks
- A time plan

The plan should be detailed enough to track weekly progress. Make sure to take component delivery times into account.

Hints on project planning

- Break project into manageable subtasks
- Establish dependencies between subtasks
- Estimate time required for each subtask
- For each week, estimate how many hours every member of the team will work
- Plan deadlines for each subtask using the above estimate
- **Put any spare time you might have at the end of the schedule, not the beginning!**
- Every week follow up on your progress compared to your time plan, and schedule more time if you are falling behind.

Example plan is distributed in group's git repo.

Project organization

We will use the git version management system for *all* files. Tutorial today. Each of you will need to create an account at . Instructions at course web page, ask your supervisor if you have questions.

We will use the issue and milestone tracking capabilities of gitlab to keep track of progress, tasks and bugs. Also covered in tutorial.

Project file organization

Every group will organize their files in the following directory structure:

- doc
- report
- model
- src
- test

The intented contents are explained on the course web page.

Reproducability

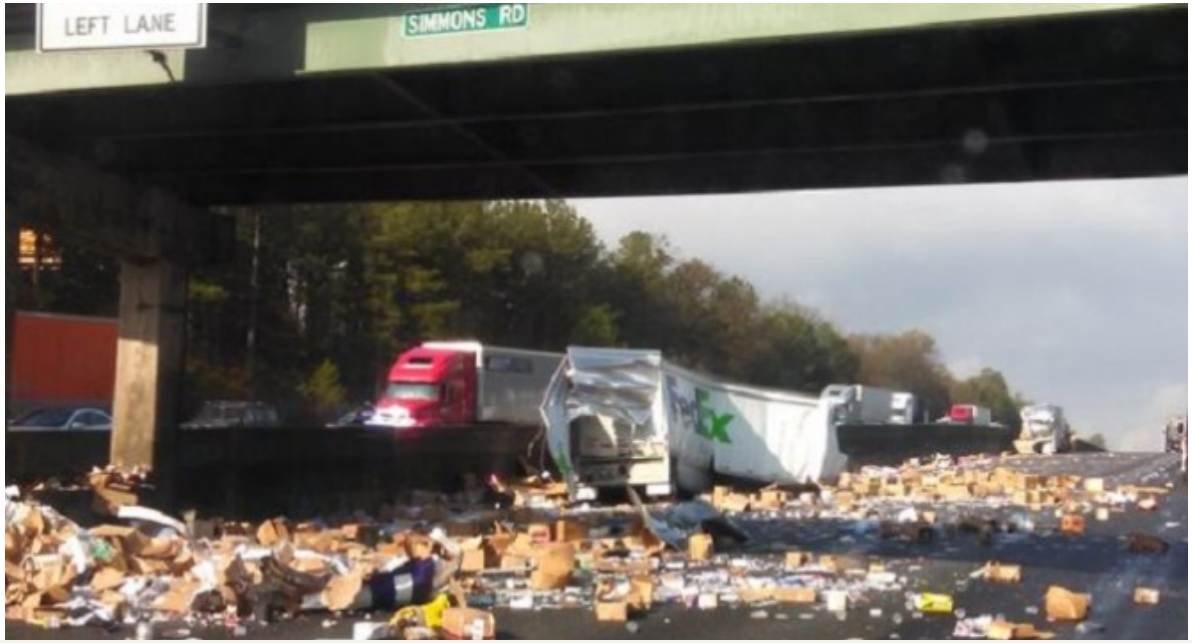
A person with similar background to yours should be able to reproduce your results from the documentation and access to your git repo.

Lab & tools – let's keep things tidy

- Always ask your supervisor if you need to borrow or order something.
- Keep your stuff confined to you assigned space.
- Pack things up neatly whenever you leave the room.
- Any tools or components you borrow should be returned to your supervisor.
- Your supervisor will need to e-mail Pontus Andersson to confirm that you have cleared your workspace and returned everything, prior to LADOK registration.

Component ordering

- All orders go through your supervisor. Each group provides info of *all* needed components via order spread sheet (access circulated via mail list).
- Supervisors need to confirm with teachers prior to ordering items exceeding SEK 1000.
- Make sure to include order lead times in your planning (and be prepared for delays!).



Feedback seminars

Purpose

Two seminars provide an arena for receiving and providing feedback across groups and course staff. Also an opportunity to follow up on and revise your planning.

- *Feedback seminar 1* on modeling and design. Each group prepares 10 min presentation.
- *Feedback seminar 2* on design and implementation. Each group writes 4–6 page report and reviews another group's report.

Important:

Practical info regarding dates, format and review process are available at the course web page.

Demonstration

- Be well-prepared. Make sure everything runs the day before.
- We will mingle around. Rotate groups members so everyone gets a chance to both demo and see other demos.
- Staff from the department will be invited. Your friends are welcome, space allowing.

Awards

A price will be given to each member of the groups with the best:

- control engineering solutions/demo
- project documentation and final report

Meeting deadlines and attending course activities is necessary for qualification. The jury consists of the course staff.

Final report

- All reports written in LaTeX. Template provided in your git repo.
- A LaTeX tutorial will be given, see course web page.
- The reports will be printed into a proceedings booklet. You will receive one copy each.
- Each group is to provide a review of another group's report. Practical info is available on the course web.

The template has a heading structure, which should be followed. Make sure to communicate with your supervisor throughout the writing and review process.

Examination (check list)

For course credits, it is required to

- participate in the project work and all mandatory activities (see above). clean up after your group in the lab.
- return all borrowed items, such as tools, components, etc. and make sure to have your group's supervisor e-mail Kristian, with cc to Pontus and group members, that your group has returned everything and cleaned up.
- do the above within stipulated deadlines, and make sure to have each item approved by your group's supervisor

Course web page

Let us spend a few moments navigating the course web page.

Project wish list

- Make a list of three (3) projects.
  Sorting: most desirable first.
- Choose from presented proposals or make one up.
- If you make one up, talk with one of the teachers *before* submitting the wish list.
- If you have friends who you want to work with, propose a group (and only submit one list per roup).
- E-mail list to Kristian by deadline posted on course web. Don't forget to write your name(s) in the e-mail.

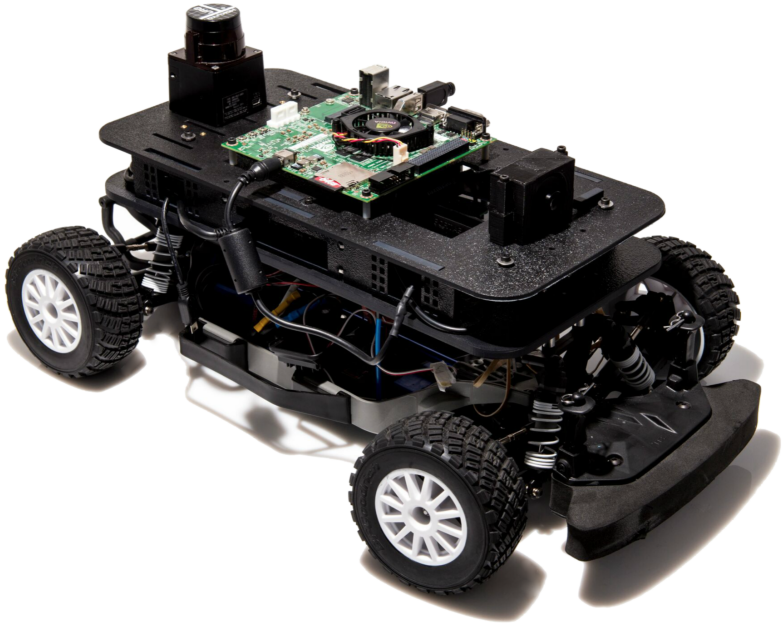
Questions?

Hope you will enjoy and learn from this course!

Let us know if you have any questions.

Project 1: Autonomous Driving (f1tenth-drive)

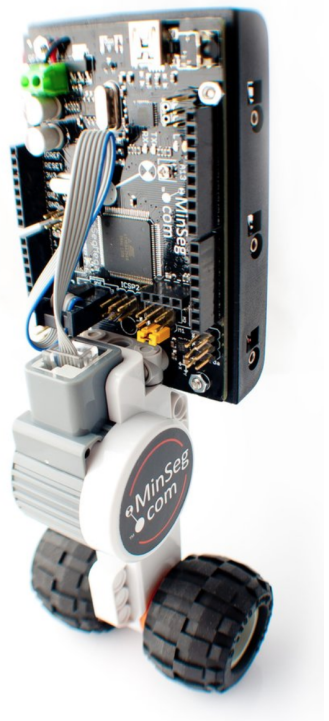
- Formula 1, a tenth of the size
- LIDAR sensor to measure the distance between the car and objects in the environment
- Drive on a path, keeping a given distance from the wall + *personalization*
- Tutorials and material available on the website
- Personalization examples:
  (1) model predictive control,
  (2) stopping at intersections

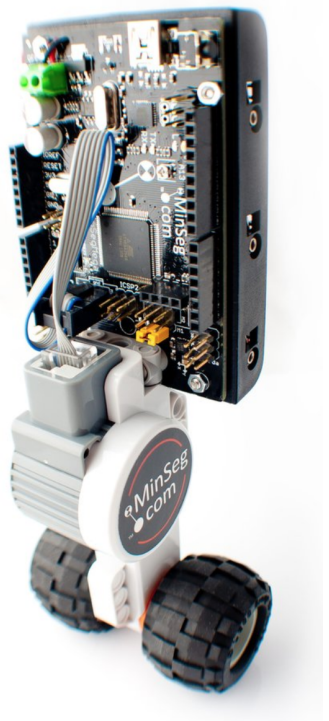


- Coding in python
- http://f1tenth.org

Project 2: Choreography (minseg-choreo)

- Create a choreography with four minseg and a camera
- Add a coloured dot on top of each minseg to distinguish them
- Process the camera frames on a computer to retrieve the minsegs positions
- Send them control signals with bluetooth and make them follow a given setpoint





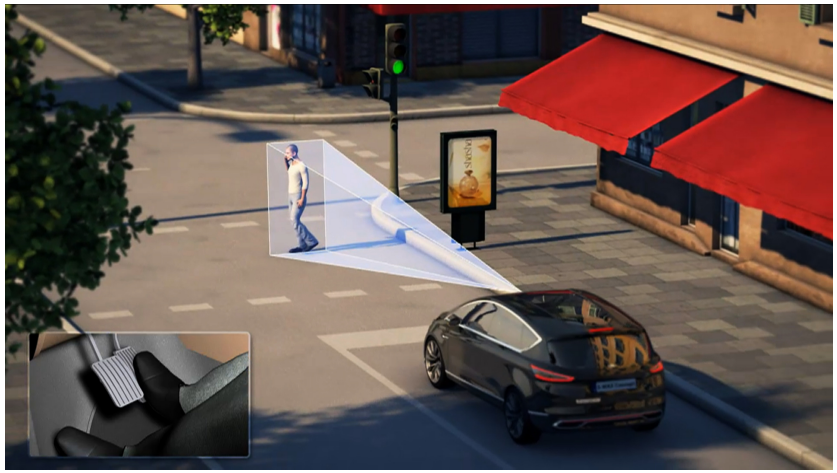
- https://minseg.com/

Project 3: Autonomous Parallel Parking (Lego)

- Parallel parking of a Lego trailer
- Using camera on top of trailer OR from bird's-eye view

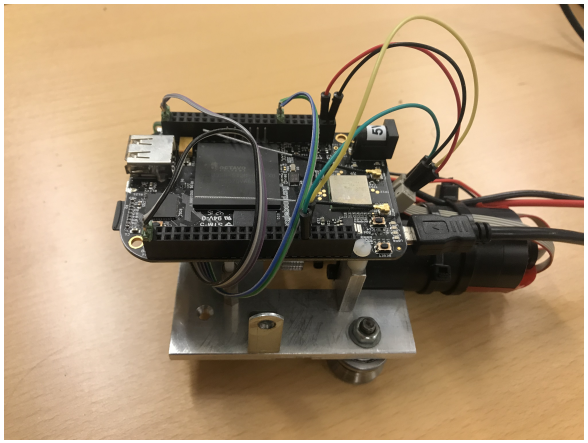
Project 4: Obstacle Avoidance (Lego Trailer)

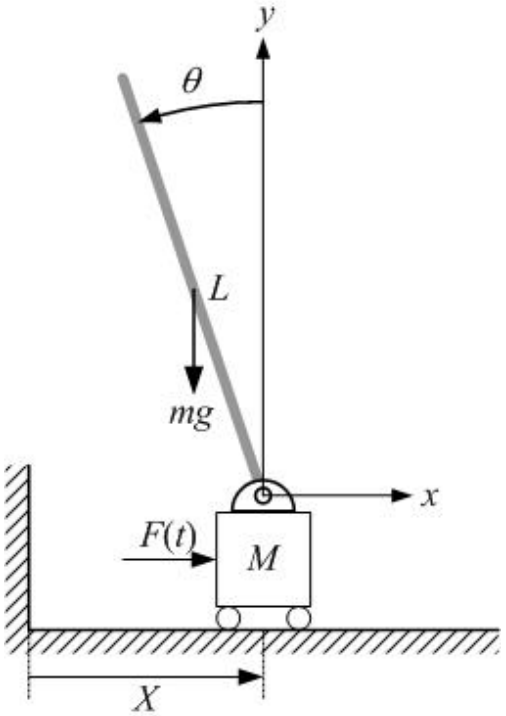
- Real-time obstacle avoidance for Lego trailer system
- Path planning and image analysis in real-time
- Moving Lego trailer (driving in reverse or forward)



Project 5: Linear pendulum

- Model and control a linear (cart on rail) pendulum
- Beablebone black, quadrature encoders, DC motor driver
- Writa a lab manual, so that what you make can be used in our education program





Project 6: Robot Control (Robot Arm)

- Real-time object tracking and path planning
- Motion- and kinematics control
- Catch ball, move things, etc.
- Your own suggestions?

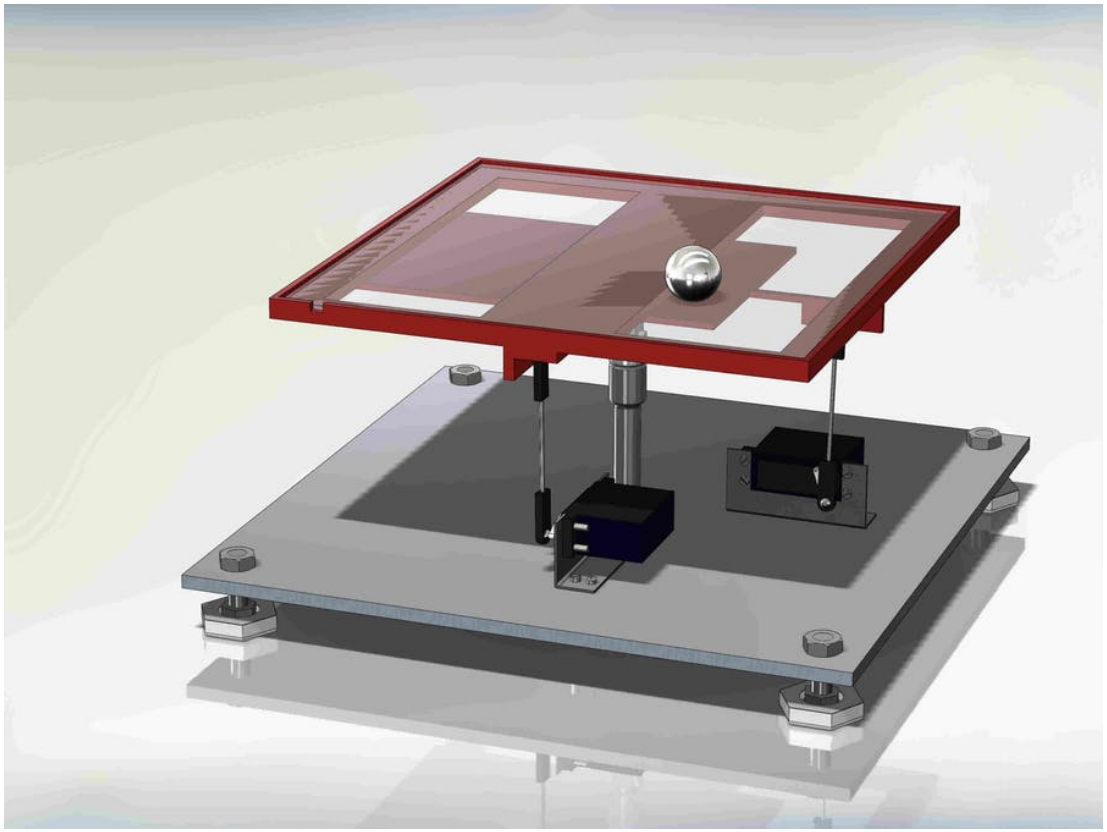
THIS COULD BE YOUR PROJECT!! ⟶

Project 7: Lego Segway Robot

- Build a Lego segway robot from scratch
- Implement control for self-balancing and driving
- Drive the robot via bluetooth using e.g smartphone

Project 8: Ball and Plate

- Build a ball and plate process
- Control position of ball using Arduino or Raspberry Pi
- Camera or touch panel to track ball position



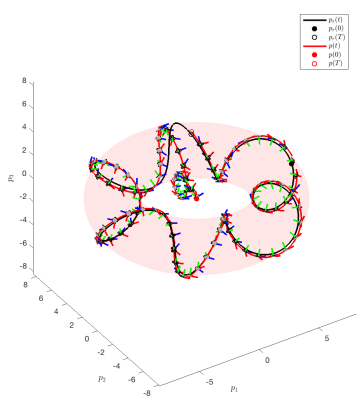
Project 9: UAV control

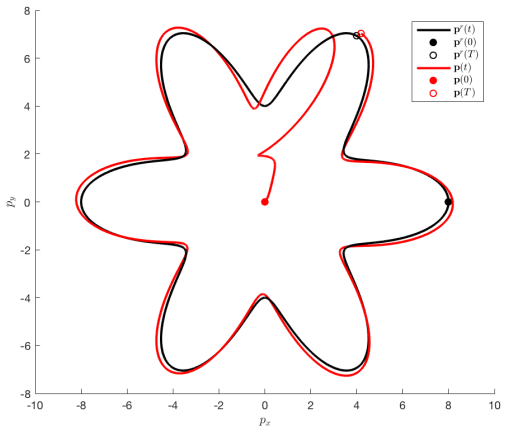
Rough outline

- Analyze a controller (MIMO, PID or LQR)
- Implement controller in ROS
- Make a nice UAV demo

Selling points

- It flies!
- Lots of freedom with the demo!





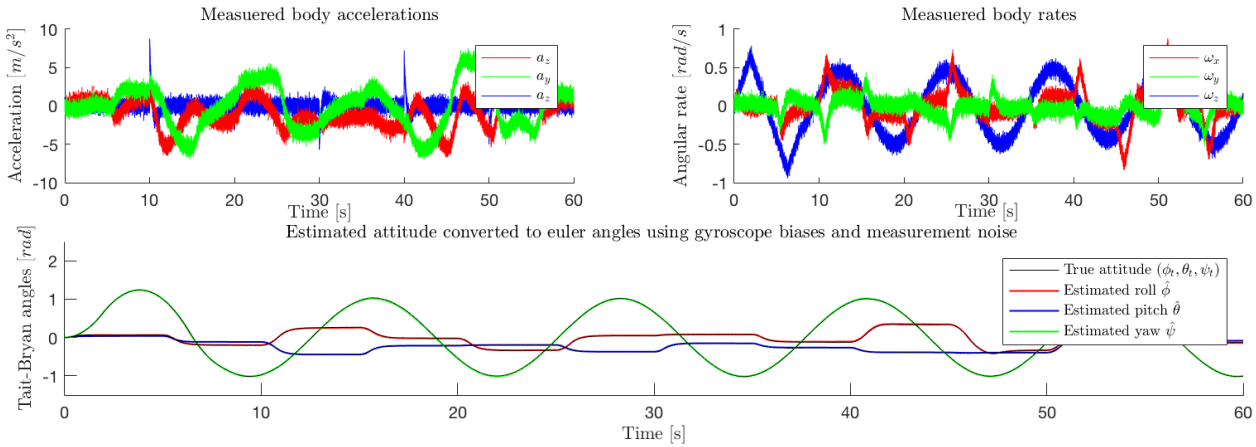
Project 10: UAV attitude estimation

Rough outline

- Analyze an estimator for AHRS*
- Implement filter in C
- Fly with the estimator

Selling points

- Interesting - quaternions and rotations!
- Useful - frequently found in robotics!



* Attitude Heading and Reference System (AHRS), could be Mahony-, Madgwick-, or nonlinear Kalman filter of your choice.

Project 11: XC05 Adaptive Controller

Try the MC XC05—an industrial adaptive controller from FirstControl

- Model and simulate a nonlinear process (the ETH helicopter) using the Modelica language
- Run the adaptive controller against the model and then on the real process
- Laptop with Windows required

Project 12: Batch Process in Continuous Mode

The batch process is a multivariable process with two pumps, heating, cooling, and mixing capabilities.

- Investigate operating the process in continuous mode (small constant flow)
- Model the system using system identification and/or mathematical modeling
- Design multivariable controller for regulating liquid level, temperature and (simulated) concentration