

Material for preliminary exam of MEC-E5001 Mechatronic Machine Design course

Kari Tammi

The purpose of this document and the preliminary exam is to guarantee a certain basic knowhow while entering in the course. The course will be executed within six weeks without significant time for recap or absence. Please, be prepared to study and have your holidays before or after the course. A longer absence during the course easily leads to failing and possibly re-joining the course in forthcoming years. Completion of this course is strongly recommended before entering MEC-E5006 Vehicle Mechatronics course.

Please, familiarise yourself in the course content in Table 1. This document includes six chapters below following the course content. The text has been written in a compact form. Please, read the text carefully. *Significant terms have been formatted in italic font*. Take care you understand the terms. Search for the terms if the explanations are not sufficient. And be prepared to explain those terms in the preliminary exam. This document also includes multiple tasks carried out by Internet search and *Matlab* exercises. Maybe the best way to study for the preliminary exam is to keep Matlab and your favourite search engine open, test and study the problems while reading the text. Make sure you know Matlab basics before the course.

This document has references to the other documents; those documents are included in the scope of preliminary exam. Hence, the questions in the exam may be from the material references. Citations in the text are not included in the scope of the preliminary exam, but a good mechatronics engineer definitely knows them.

Table 1. The course content.

Wk	Lecture	Exercise	Other
1	Introduction to the course and background of mechatronics, Mechatronic design process	Learning / re-cap of Matlab	
2	Dynamic systems, frequency and time domain analysis	Laplace transform, Transfer function, Impulse and step responses, Basics of dynamic models	Preliminary exam deadline Release of project work
3	Electronics, control	Operational amplifier circuits, AD & DA conversion, Bode diagram	
4	Common control topologies, PID controller, Control applications	Common control topologies, PID controller, Control applications	Laboratory exercise 1
5	Mechatronic machine design with case example, Visiting lecturer	Mechatronic system simulation	Laboratory exercise 2
6	Summary of the course, Students' reflections: what we learnt, Mutual feedback	No exercises	Project work deadline
7	Project work wrap up /gala	Project work wrap up /gala	Course finished

1 Introduction to the course and background of mechatronics

What is *mechatronics*? Many answers can be found in literature. In general, mechatronics is explained as a design process that includes a combination of mechanical engineering, electrical engineering, telecommunications engineering, control engineering and computer engineering. In other words, mechatronics is a multidisciplinary field of engineering and it rejects splitting engineering into separate disciplines. ([wiki](#))

Maybe the simplest way is to define mechatronics as a multidisciplinary approach to design and study machines having *sensing, control, and actuation* functionalities. However, it is necessary to remember mechatronic machines have interfaces to other machines, various processes, and human beings. Hence, even behavioural sciences and usability research touch mechatronics via human-machine interaction.

Machine design process models have been developed in order to formalise the design process and increase the efficiency of the process. Commonly known process models include:

- [Incremental](#), classical development circle
- [Waterfall](#), classical definitions chain
- [V cycle](#), modern variation, emphasises validate
- [Spiral](#), sequences of waterfall chains
- [Scrum](#), modern, software engineering inspired

Assignments

1. Make yourself familiar with a company producing mechatronic products in Finland. Search and visit the company websites. Find answers to the following questions: What does the company produce? Where do they use mechatronics in their products? What type of *sensors, controls, and actuators* do they use in their products?

2. Browse the wiki links above and study the design process models: *Incremental, Waterfall, V cycle, Spiral, and Scrum*.

3. Open Matlab and test the following codes:

```
>> t=0:0.001:1;
```

```
>> w=2*pi*10; % f = 10 Hz
```

```
>> plot(t,sin(w*t),'k');
```

```
>> hold on;
```

```
>> phi=pi/2;
```

```
>> plot(t,sin(w*t+phi),'r');
```

AND

```
>> sys=tf(1,[1 0.1 1]); % requires control system toolbox
```

```
>> step(sys);
```

```
>> impulse(sys);
```

What do the codes execute? Explain the code row by row. What is the difference between *step()* and *impulse()*? Hint: command *doc* helps (just type *doc* or *doc step*, for instance).

2 Physics based design of mechatronic machines. Computational methods for machine design

Design of a mechatronic machine requires understanding of the prevailing physics of the machine and its surroundings. The design specification is usually derived from a required capacity, performance, consumption, *etc.* Various types of models of the machine itself and its surroundings are used to transfer the machine specification into the component specification and finally to design. Mechanical engineering design often includes *multibody systems (MBS)* and *finite element (FE)* tools. Electrical and control engineering tools include *circuit and system simulators* and also FE tools. Computer engineering tools include *software testing, debugging, and hardware emulator tools*. The above-mentioned list is not exhaustive and the tools can vary depending on the mechatronic machine designed and production volumes, for instance.

Basic functionalities of a mechatronic machine can be modelled and analysed with a set of analytical tools. A typical task is to derive the actuator specification according to the work process a machine is required to carry out.

Assignments

1. Search for *PID control* and find out the roles of P, I, and D coefficients, respectively.

2. Consider an electric forklift.

- a) The forklift lifts a 100-kg pallet from floor to up to 2-m shelf in 4 sec. What is the average power needed for lifting? Draw a free-body diagram of the situation.
- b) The forklift carries out the same lifting sequence 30 times an hour. How much energy is consumed?
- c) Hourly traction demand is the following: 0 W during 30 min, 1000 W during 15 min, 2000 W during 10 min, 3000 W during 5 min. How much energy is consumed?
- d) What battery power & capacity [Wh] are required? The forklift is charged once an hour. Power transmission efficiency is assumed to be 70 %.

3. Open Matlab, Simulink, and Simulink model file: *Fork_lift_traction_example_MMD.slx*. PID controller acts as a speed controller of the forklift speed. The PID controller is tuned by using Matlab tuning tool (can be opened in the PID block). The model is a simplistic representation of the forklift powertrain.

Find out the meaning for each block in the model. Simulate with default parameters and find out how the simulated speed exceeds the reference speed at 15 s, accelerates until about 25 s until starts to decelerate and reaches the reference speed at 35 s. Next, multiply integral gain by 0.01 and find out what happens: speed overshoot is nearly gone.

Find out answers what:

- is the role of integrator in a PID controller?
- means *integrator windup*?

3 Mechatronic devices and their dynamics modelling: static and dynamic models

Control systems have classically been based on analogue hardware. With digital computing becoming more and more prevalent, digital control hardware has taken over. The vast majority of

the control system realisations are digital today. Sensors and actuators can either be analogue (*e.g.* force, strain) or digital (*e.g.* on/off, step, backward/forward) in their nature.

In control theory and control engineering, the analysis can be carried out in continuous time (analogue control) or discrete time (digital, computer or sampled control). A discrete time controller may be analysed in continuous time; the analysis results are often similar. However, multiple exceptions exist, too. Terms *Laplace domain* and *Z domain* refer to continuous and discrete time systems, respectively.

At the inputs of a digital control system, the analogue sensor outputs are filtered and converted to digital format by an [analogue to digital converter](#) (*AD converter*). Various signal conditioning systems exist, the most common are amplification/attenuation devices and [low-pass](#), [high-pass](#), or [band-pass](#) filters. The behaviour of filters, and many other components including frequency dependent behaviour can be described with a [frequency response](#). Figure 1 shows a frequency response of a low-pass filter; the filter pass band is defined by -3 dB point where the gain has decreased by 3 dB compared to the gain at 0 frequency (the DC gain).

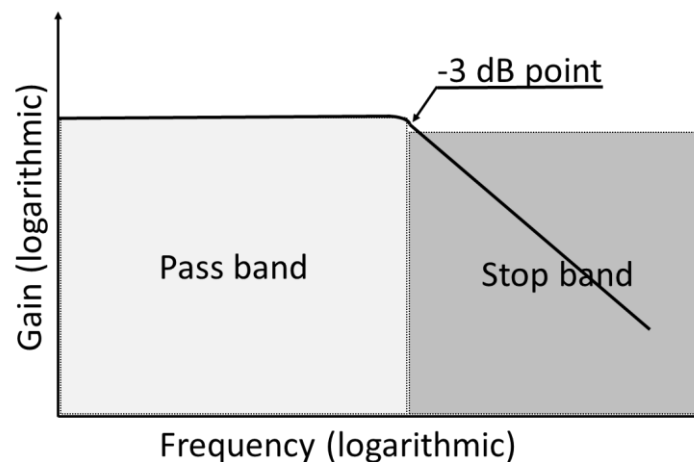


Figure 1. A schematic frequency response of a low-pass filter: the filter gain as a function of frequency.

In the case of analogue actuators, a digital control system converts the control command into an analogue signal by a [digital to analogue converter](#) (*DA converter*).

Assignments

1. Make yourself familiar with the *Shannon's sampling theorem and Nyquist frequency*. Test in Matlab the following code (variable $w = 62.8319$ rad/s as in the earlier assignment):

```
>> w = 62.8319; % angular frequency [rad/s]
>> t=0:0.001:1; % sampling 1/0.001 Hz
>> plot(t,sin(w*t),'k. ');
>> hold on;
>> t=0:0.01:1; % sampling 1/0.01 Hz
>> plot(t,sin(w*t),'r+');
>> t=0:0.1:1; % sampling 1/0.1 Hz
```

```
>> plot(t,sin(w*t),'m*');
```

What do you see? What happens with growing sampling time (lowering sampling frequency)?

2. Make yourself familiar with *Laplace* transformation. Solve a differential equation using Laplace transformation and derive the transfer function $X(s)/F(s)$. The initial values can be assumed zeros.

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = f(t)$$

3. Simulate step and impulse responses for the transfer function $X(s)/F(s)$ above, $m = 1$, $c = 0.01$, $k = 1$. You can use *tf*, *step*, *impz* commands or build a Simulink model to simulate the responses.

4 Mechatronic control, hardware, theory

Mechatronic machines regularly require a set of control systems. Control systems can be cascaded on different levels. High level controls take care of machine functions, *e.g.* accelerating a machine/vehicle, lifting a component, opening a gate *etc.* Lower-level controls take care of machine components or subcomponents *e.g.* open a valve, increase a current, decrease a pressure *etc.* *Open-loop control* refers to a control system that is not explicitly using an output measurement to correct the control action. *Closed-loop control* uses a feedback measurement to adjust the control actions in such a way that the control error is minimised (Figure 2).

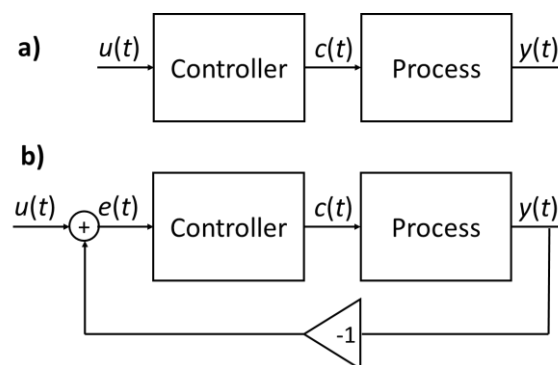


Figure 2. Open-loop a) and closed-loop b) control topologies.

Control theory and control engineering study control systems, their performance, robustness, and stability. In this course, we focus on commonly-known controllers such as *proportional-integral-derivative (PID)* control. It is worth knowing that multiple approaches for classical controller synthesis exist such as *optimal, robust control methods*. *Soft-computing methods* include for example *fuzzy control, neural networks, and evolutionary algorithms*.

The control algorithms developed are implemented in a selected control hardware, called a *target hardware*. The hardware realisations can for instance be *microcontrollers (μC), digital signal processors (DSP), field-programmable gate array (FPGA)*, or an ordinary computer. Programming a target hardware can be carried out in various ways. Low-level programming languages such as Assembler are utilised to produce an efficient code that can be executed with minimal computational resources. The time used for the code development is typically longer for low-level languages. Higher-level languages and graphical programming are used more today as data processing has become faster. However, low-level programming is relevant when efficiency and reliability requirements for the code are high. Figure 3 shows an example workflow of a hardware and software development project and target programming process.

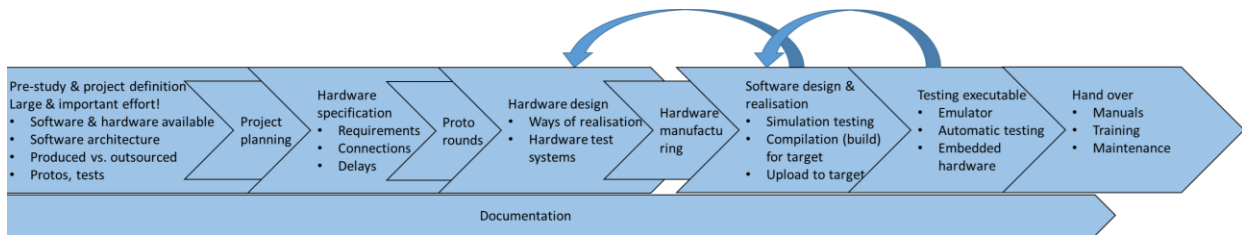


Figure 3. An example of hardware and code development process for an embedded system.

Assignments

1. In Figure 2 b), the controller is realised as a PID controller. The controller outputs the control signal $c(t)$ based on the error signal $e(t)$ magnitude (proportional control action), its accumulation (integral action) and its instantaneous change (derivative action). Analyse a PID controller in Matlab/Simulink:

```
>> simulink
```

Open file `PID_example_MMD.slx`

The *rise time*, *overshoot* and *settling time* describe the performance of a controller. What happens if you increase coefficients: K_p , K_i , or K_d by 50 %? Try out the changes by varying one coefficient at a time. Hint: you can multiply the original coefficient by 1.5, and then cut and paste 1.5* to the next coefficient.

2. Search for an FPGA based controller and find an example where it has been used.

3. Study Figure 3. How you would fit it in the V cycle development process?

5 Mechatronic machine design case example

The 5th lecture features variable content. A visiting lecturer / lecturers will speak about a case example related to a mechatronic machine and/or its design process.

Active magnetic bearings (AMBs) are described as a case example of a mechatronic machine. AMBs typically appear in high-speed rotating machines. A rotor is levitated in an air gap by actively controlled magnetic forces. Levitating a rotor provides advantages such as low friction and almost maintenance-free operation. AMBs have been used in a variety of high-speed applications, from magnetically levitating trains to small electrical machines. AMBs have changed rotor applications remarkably by pushing up the maximum speeds of rotating machines. The fastest industrial rotors operate at a speed of 6 kHz (360000 rpm). The maximum speed is limited by the rotor's material strength rather than by a resonance behaviour (in rotors supported by conventional bearings, resonances, lubrication, and friction losses often limit the speed).

As another example of a mechatronic machine, a passenger car has multiple mechatronic systems. *Anti-lock brake system (ABS)* is one of the mechatronic systems in vehicles. ABS prevents a vehicle's wheels from locking under emergency braking, maintaining stability, steerability, and saving vehicle tyres.

Another mechatronic system in vehicles is the electronic stability control (ESC, also known as ESP, VSC). ESC controls vehicle stability by braking individual wheels, turning the vehicle into the direction aimed by the steering wheel. ESC has been found a useful safety system in emergency situations ([wiki](#)). In the '90s, the stability control was for luxury cars. The first generation A class Mercedes changed the story. Its behaviour in extreme lane changing conditions was found unstable and was

criticized by car magazines. The manufacturer decided to bring stability control for smaller and cheaper vehicles too. An interesting consequence was that acceleration and gyroscopic sensors were needed in large volumes. The consequences were good for us Finns producing those sensors (today the producer is a part of Murata company).

Assignments

1. Read the spinning rpm World record paper in ICEM 2016: *Rotor Losses in an Ultra-High Speed Spinning Ball Motor* by Marcel Schuck, Daniel Steinert, and Johann W. Kolar. Focus on general description on pages 2611-2612 and conclusions on p. 2616. Find out what sensors, actuators, and control systems were used in the study.
2. Read description of ABS system in the book *Mechatronics in Action* by David Bradley and David W. Russell (editors) on pages 97-101. Find out what sensors, actuators, and control systems are used in ABS systems.

6 Summary of the course

As claimed in the beginning, mechatronics rejects the boundaries of engineering disciplines and it assumes the machine design process is a union of machine design, electrical and control design, and software design. Reading this document shows the design process of a mechatronic machine involves multiple disciplines, includes several stages, and requires various skills. The design process contains the development of different types of modelling, testing, validation, analysis and implementation.

Machine design process models have been developed in order to formalise the design process and increase the efficiency of the process. Commonly known process models include: *Incremental*, *Waterfall*, *V cycle*, *Spiral*, and *Scrum* processes.

Assignments

Find answers to the questions:

1. Think and reflect: what have you learnt reading this document for the preliminary exam, what did you know before, what surprised you, what do you want to learn during the course?
2. What type of sensors, actuators, controls (hardware & software) you have used? And what you likely use in the future?
3. Study all the assignments in this document once again and think about the answers. Please provide feedback, suggestions to improve this document during the course.