Curriculum for

Bachelor’s programme in Electronics and Computer Engineering

Faculty of Engineering and Science

September 2011
Preface:
Under Law 695 of 22 June 2011 on Universities (The University Act) as subsequently amended the following curriculum for the Bachelor’s degree in Elektronik og datateknik, which is offered in Esbjerg. The program also follows the Framework Provisions and the Examination Regulations at the Technical Faculty of Science.

AAU, August 2011

Uffe Kjærulff
Director of studies
# Indholdsfortegnelse

## CHAPTER 1: LEGAL BASIS OF THE CURRICULUM, ETC.

- 1.1 Basis in ministerial orders .......................................................... 4
- 1.2 Faculty affiliation ........................................................................ 4
- 1.3 Board of Studies affiliation ........................................................... 4

## CHAPTER 2: ADMISSION, DEGREE DESIGNATION, PROGRAMME DURATION AND COMPETENCE PROFILE

- 2.1 Admission ...................................................................................... 4
- 2.2 Degree designation in Danish and English ........................................ 4
- 2.3 The programme’s specification in ECTS credits ................................... 4
- 2.4 Competence profile on the diploma ................................................ 4

## CHAPTER 3: CONTENT AND ORGANISATION OF THE PROGRAMME

- 3.1 Overview of the programme .......................................................... 7
- 3.2 Descriptions of modules ............................................................... 8

## CHAPTER 4: ENTRY INTO FORCE, INTERIM PROVISIONS AND REVISION

## CHAPTER 5: OTHER PROVISIONS

- 5.1 Rules concerning written work, including the Bachelor’s project ........... 62
- 5.2 Rules concerning credit transfer (merit), including the possibility for choice of modules that are part of another programme at a university in Denmark or abroad .............................. 62
- 5.3 Rules concerning the progress of the Bachelor’s programme ................ 62
- 5.4 Rules concerning the completion of the Bachelor’s programme ............ 62
- 5.5 Special project process .................................................................... 63
- 5.6 Rules for examinations .................................................................... 63
- 5.8 Rules and requirements for the reading of texts .................................. 63
- 5.9 Additional information .................................................................... 63
Chapter 1: Legal Basis of the Curriculum, etc.

1.1 Basis in ministerial orders
The Bachelor’s programme in Electronics and Computer Science is organised in accordance with the Ministry of Science, Technology and Innovation’s Ministerial Order no. 814 of June 29, 2010 on Bachelor’s and Master’s Programmes at Universities (the Ministerial Order of the Study Programmes) and Ministerial Order no. 857 of July 1, 2010 on University Examinations (the Examination Order) with subsequent changes. Further reference is made to Ministerial Order no. 233 of March 24, 2011 (the Admission Order) and Ministerial Order no. 250 of March 15, 2007 (the Grading Scale Order) with subsequent changes.

1.2 Faculty affiliation
The Bachelor’s programme falls under the Faculty of Engineering and Science.

1.3 Board of Studies affiliation
The Bachelor’s program falls under the Board of Studies for Electronics and IT.

Chapter 2: Admission, Degree Designation, Program Duration and Competence Profile

2.1 Admission
Admission to the Bachelor’s programme in Electronics and Computer Science requires an upper secondary education.

The program’s specific requirements are: English B, Mathematics A and Physics B and either Chemistry or Biotechnology A according to admission notice. All subjects must be passed.

2.2 Degree designation in Danish and English
Bachelor of Science (BSc) in Engineering (Electronic and Computer Engineering). The Danish designation is Bachelor (BSc) i teknisk videnskab (elektronik og datateknik)

2.3 The programme’s specification in ECTS credits
The Bachelor’s programme is a 3-year, research-based, full-time study programme. The programme is set to 180 ECTS credits.

2.4 Competence profile on the diploma
The following will appear on the diploma:

A graduate of the Bachelor's programme has competencies acquired through an educational programme that has taken place in a research environment.

A graduate of the Bachelor's programme has fundamental knowledge of and insight into his/her subject's methods and scientific foundation. These properties qualify the graduate of the Bachelor’s programme for further education in a relevant Master’s programme as well as for employment on the basis of the educational programme.
## 2.5 Competence profile of the programme:

### The graduate of the Bachelor’s programme:

<table>
<thead>
<tr>
<th>Knowledge</th>
<th>Skills</th>
<th>Competencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Has knowledge of and insight into fundamental theories, methods and practical subjects within the fields of Electronics and Computer Engineering&lt;br&gt;- Is able to understand and reflect upon said theories, methods and practical subjects&lt;br&gt;- Has a firm grasp of the mathematical and programming technical foundations of the field&lt;br&gt;- Has knowledge of the interaction between electronic and physical systems, including feedback mechanisms, electronic circuits, electromechanical systems&lt;br&gt;- Has insight into techniques and methods for real-time acquisition, storage and processing of complex information&lt;br&gt;- Has knowledge of all aspects of low-level computing, from hardware platform to operating systems and software applications</td>
<td>- Can utilize up-to-date scientific methods, tools and techniques to analyze and solve complex problems the fields of Electronics and Computer Engineering&lt;br&gt;- Can evaluate and compare theoretical and practical problems, as well as describe and select relevant solution strategies&lt;br&gt;- Is able to implement such solution strategies and evaluate their success in a systematic manner&lt;br&gt;- Is able to present problems and solution strategies within the fields of Electronics and Computer Engineering, in writing as well as orally, to specialists as well as non-specialists in the fields, including external parties, users, etc.</td>
<td>- Is able to handle complex situations that arise in research and/or development-related environments, such as university studies and/or engineering workplaces.&lt;br&gt;- Is able to develop and test hardware and software for embedded systems in a broad systems-oriented context&lt;br&gt;- Can work independently as well as in collaboration with others, both within and across technical fields, in an efficient and professional manner&lt;br&gt;- Is able to identify his/her own learning needs and structure his/her own learning in various learning environments</td>
</tr>
</tbody>
</table>
Chapter 3: Content and Organisation of the Programme

The programme is structured in modules and organised as a problem-based study. A module is a programme element or a group of programme elements, which aims to give students a set of professional skills within a fixed time frame specified in ECTS credits, and concluding with one or more examinations within specific exam periods. The examinations are defined in the curriculum.

The programme is based on a combination of academic, problem-oriented and interdisciplinary approaches and organised based on the following work and evaluation methods that combine skills and reflection:

- Lectures
- classroom instruction
- project work
- workshops
- exercises (individually and in groups)
- teacher feedback
- reflection
- portfolio work
3.1 Overview of the programme:
All modules are assessed through individual grading according to the 7-point scale or Pass/Fail. All modules are assessed by external examination (external grading) or internal examination (internal grading or assessment by the supervisor only).

<table>
<thead>
<tr>
<th>Semester</th>
<th>Module</th>
<th>ECTS</th>
<th>Assessment</th>
<th>Grading</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED1</td>
<td>Technological Project Work</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Monitoring and Programming</td>
<td>10</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Imperative Programming</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Problem-based Learning in Science, Technology and Society</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Linear Algebra</td>
<td>5</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td>ED2</td>
<td>Analog Instrumentation</td>
<td>15</td>
<td>7 grade scale</td>
<td>External</td>
</tr>
<tr>
<td></td>
<td>Calculus</td>
<td>5</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Introduction to Electrical Engineering</td>
<td>5</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Digital Design &amp; Sensors</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td>ED3</td>
<td>Mechatronics</td>
<td>15</td>
<td>7 grade scale</td>
<td>External</td>
</tr>
<tr>
<td></td>
<td>Algorithm &amp; Data Structures</td>
<td>5</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Mathematics</td>
<td>5</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Micro Processors and Programming</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td>ED4</td>
<td>Basic Control Engineering</td>
<td>15</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Object Oriented Software Engineering</td>
<td>5</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Fundamental Control Theory and Modelling</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Embedded Software Design</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td>ED5</td>
<td>Distributed Computing</td>
<td>15</td>
<td>7 grade scale</td>
<td>External</td>
</tr>
<tr>
<td></td>
<td>Numerical Methods</td>
<td>5</td>
<td>7 grade scale</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Signal Processing</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Operating systems and Network Data Communication</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td>ED6</td>
<td>BSc Project (Control Engineering)</td>
<td>15</td>
<td>7 grade scale</td>
<td>External</td>
</tr>
<tr>
<td></td>
<td>BSc project (Embedded Real-time Signal Processing)</td>
<td>15</td>
<td>7 grade scale</td>
<td>External</td>
</tr>
<tr>
<td></td>
<td>BSc Project (Distributed Systems)</td>
<td>15</td>
<td>7 grade scale</td>
<td>External</td>
</tr>
<tr>
<td></td>
<td>BSc Project (Programming Language)</td>
<td>15</td>
<td>7 grade scale</td>
<td>External</td>
</tr>
<tr>
<td></td>
<td>Introduction to Probability Theory and Statistics</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Matrix Computation and Convex Optimization</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Database, Language &amp; Compilers</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
<tr>
<td></td>
<td>Distributed Systems</td>
<td>5</td>
<td>Pass/Fail</td>
<td>Internal</td>
</tr>
</tbody>
</table>
3.2 Descriptions of modules

Technological Project Work
_Teknologisk projektarbejde_

Semester: 1st semester
Purpose: Through this module, the student shall acquire knowledge about problem oriented and problem based learning. Furthermore, he/she shall acquire first-hand knowledge about project-oriented group work as a learning method. Additionally, the student will be introduced to basic problems and concepts within the field of Electronics and IT.

Objectives: After completion of the module, the student:

* **Knowledge**
  - Shall have insight into elementary concepts related to project-oriented group work.
  - Shall be familiar with the processes involved in project work, knowledge acquisition and supervisor collaboration

* **Skills**
  - Shall be able to define project goals and work in a methodical manner toward achieving such goals
  - Shall be able to describe and analyze several approaches to project solutions
  - Shall be able to present results achieved within the project in writing, orally, and graphically in a comprehensive manner.

* **Competencies**
  - Shall be able to reflect upon the problem oriented and problem based learning approach taken throughout the study
  - Shall be able to document the results achieved during the project in a report
  - Shall be able to cooperate with other students during the project period and make a joint presentation of the results achieved in the project.
  - Shall be able to reflect upon different ways of presenting results achieved with the project in writing, orally, and graphically.

Content: The project group must prepare a report and process analysis, participate in a P0 collection of experience and attend a presentation seminar where the project group documents discussed.

Type of instruction: Project work with supervision
Exam format: Individual oral examination based on a written report
Monitoring and Programming
Overvågning og programmering

Semester: 1st semester
Prerequisites: Technical Project Work (P0)
Purpose: One of the most fundamental capabilities any Computer Engineer must possess is the ability to construct functionality that allows a computer to interact with its surroundings. Through the 1st semester project, the students shall acquire knowledge within Electronics and Computer Engineering through practical and theoretical work. The project takes its starting point in a problem of relevance to society or industry; the problem is then broken down into smaller, more manageable sub-problems and analyzed for the purpose of defining a relevant technical problem formulation, which can be solved via theories and methods related to micro-processor-based systems. The solution shall encompass an electronic system containing (at least) a programmable electronic computing device, which is able to measure signals from its surroundings via selected sensors and process them digitally in some form.

Objectives: Students who complete the module:

Knowledge
• Must have understanding of fundamental electronic systems and their interaction with their surroundings
• Must have knowledge about sampling of continuous signals
• Must have basic insight into concepts such as signals, sensors, actuators and micro-processors
• Shall have sufficient insight into technological and social issues to enable them to pinpoint relevant problems that can be solved by technical means
• Shall have knowledge about common processes in extensive, problem-oriented projects
• Shall be able to explain and clarify theories and methods used in the project

Skills
• Given a socially relevant problem, must be able to identify relevant requirements to a technical solution, product or similar
• Must be able to follow a relevant method for structured development in the project, including formulation and analysis of the problem, define a requirement specification and divide the problem into sub-problems that can be resolved separately
• Shall be able to select and utilize sensors and actuators for data collection and interaction with an electronic system's surroundings
• Shall be able to formulate and solve technical problems via
algorithms and be able to implement said algorithms in a microprocessor or similar programmable device

- Shall be able to analyze and evaluate their own utilization of said theories and methods
- Shall be able to document and present the knowledge and skills outlined above, using correct terminology, in writing as well as orally
- Shall be able to analyze and evaluate their own learning processes using relevant methods
- Shall be able to plan and carry out an extensive group project in collaboration with a supervisor

**Competencies**

- Shall understand the general concept of a system, in particular pertaining to electronic systems interacting with their surroundings
- Shall be able to assume responsibility for their own learning processes during an extensive group project, as well as generalize and interpret the experience acquired
- Shall be able to plan, structure, carry out, and reflect upon a project that starts from a socially or industrially relevant problem, in which electronic systems and information technology is an important element, individually as well as in groups.

**Type of instruction:** Project work with supervision

**Examination:** Individual oral examination based on a written report

**Evaluation criteria:** Stated in the Framework Provisions
Imperative Programming
*Imperativ programmering*

**Semester:** 1st semester  
**Purpose:** Students who complete the module enrich their background in working with computers and other digital devices in procedural ways to enable programming for different media platforms and working with analog and digital sensors.

**Objectives:** Students who complete the course module

*Knowledge*
- Shall have understanding of integrated development environments
- Shall have understanding of differences between run-time and compile-time computer programming languages
- Shall be able to explain the concepts of types, declarations, expressions and statements
- Shall be able to make use of libraries and understand the concept of linking
- Shall have insight into data structures, such as arrays
- Shall have insight into input/output in various forms
- Shall have understanding of procedures and functions, including function arguments
- Shall have understanding of pointers and references
- Shall have understanding of the complexity of a program

*Skills:*
- Shall be able to interpret and analyze a basic procedural program and elaborate its functionality
- Shall be able to design and implement algorithms for data structure manipulation using references and addresses where necessary
- Shall be able to estimate the complexity of a program
- Shall be able to explain how to use algorithms, functions and data for solving problems (understanding)

*Competencies:*
- Must be able, individually and in collaboration with others, to design and implement one or more programs to solve a previously specified problem

**Type of instruction:** As described in the introduction to Chapter 3.  
**Exam format:** Individual oral or written examination  
**Evaluation criteria:** Stated in the Framework Provisions
Problem Based Learning in Science, Technology and Society
Problembaseret læring i videnskab, teknologi og samfund

Semester: 1st semester
Purpose: To enable the student to approach real-life complex problems in a methodical manner, and to carry out project work, planning and documentation in a structured way.

Objectives: Students who complete the course module will obtain the following qualifications:

Knowledge
- Shall be able to explain basic learning theory
- Shall be able to explain techniques for planning and management of projects
- Shall be able to explain different approaches to problem-based learning (PBL), including the so-called Aalborg model based on problems that are part of a social and/or humanistic context
- Shall be able to explain different approaches to analysis and assessment of problems and solutions within engineering, natural and health sciences from a theoretical, ethical, and societal perspective
- Shall be able to explain how these methods can be applied within Electronics and Computer Science

Skills
- Must be able to plan and manage a problem-based study project
- Must be able to analyze the project group's organization and cooperation in order to identify strengths and weaknesses, and suggest how cooperation in future groups can be improved based on this analysis
- Must be able to reflect on the causes and devise possible solutions to any group conflicts
- Must be able to analyze and evaluate their own study work and learning, in order to identify strengths and weaknesses, and use these reflections to consider further study and group work
- Must be able to reflect upon the methods used from a theoretical perspective
- Must be able to identify relevant areas of focus, concepts and methods to assess and develop technical solutions under consideration of the social and humanistic contexts that solution must be a part of

Competencies
- Shall be able to take part in a team-based project
- Shall be able to document and present work carried out in a
project

- Shall be able to reflect upon and develop his/her own learning
- Shall be able to engage in and improve upon the collaborative learning processes
- Shall be able to reflect upon his/her professional activities in relation to the surrounding community

Type of instruction: As described in the introduction to Chapter 3.

Exam format: Individual oral or written examination

Linear Algebra

Purpose
Linear algebra is a fundamental tool for virtually all engineering mathematics

Objectives:
Students who complete the module:

Knowledge
- Shall have knowledge about definitions, results and techniques within the theory of systems of linear equations
- Shall be able to demonstrate insight into linear transformations and their connection with matrices
- Shall have obtained knowledge about the computer tool MATLAB and how it can be used to solve various problems in linear algebra
- Shall have acquired knowledge of simple matrix operations
- Shall know about invertible matrices and invertible linear mappings
- Shall have knowledge of the vector space $\mathbb{R}^n$ and various subspaces
- Must have knowledge of linear dependence and independence of vectors and the dimension and bases of subspace
- Must have knowledge of the determinant of matrices
- Must have knowledge of Eigen values and eigenvectors of matrices and their use
- Must have knowledge of projections and orthonormal bases
- Must have knowledge of first order differential equations, and on systems of linear differential equations

Skills
- Must be able to apply theory and calculation techniques for systems of linear equations to determine solvability and to provide complete solutions and their structure
- Must be able to represent systems of linear equations using matrix equations, and vice versa
- Must be able to determine and apply the reduced Echelon form of a matrix
- Must be able to use elementary matrices for Gaussian elimination and inversion of matrices
- Must be able to determine linear dependence or linear independence of small sets of vectors
- Must be able to determine the dimension of and basis for small
subspaces

- Must be able to determine the matrix for a given linear transformation, and vice versa
- Must be able to solve simple matrix equations
- Must be able to calculate the inverse of small matrices
- Must be able to determine the dimension of and basis for kernel and column spaces
- Must be able to compute determinants and could use the result of calculation
- Must be able to calculate Eigen values and eigenvectors for simple matrices
- Must be able to determine whether a matrix is diagonalizable, and if so, implement a diagonalization for simple matrices
- Must be able to compute the orthogonal projection onto a subspace of $\mathbb{R}^n$
- Must be able to solve separable and linear first order differential equations, in general, and with initial conditions

**Competencies**

- Shall demonstrate development of his/her knowledge of, understanding of, and ability to make use of, mathematical theories and methods within relevant technical fields
- Shall, given certain pre-conditions, be able to make mathematical deductions and arguments based on concepts from linear algebra

Type of instruction: As described in the introduction to Chapter 3.

Exam format: Individual oral or written examination

Analog Instrumentation

Semester: 2nd semester
Prerequisites: Monitoring and programming (P1)
Purpose: Through theoretical and practical work on a selected problem, the students acquire knowledge in the Electronics and IT-related engineering discipline, as well as use appropriate methods to document that the problem has a relevant social context. The problem is analyzed by decomposition into sub problems in order to formulate a technical problem that can be solved by using analog electronic systems that interact with the environment in one way or another. The complete solution is assessed with respect to the relevant social context. Compared to the first semester, this semester focuses more on the continuous/analog aspects of electronic systems as well as interaction with the surroundings in greater detail.

Objectives: Students who complete the module:

Knowledge
- Shall have gained experience with theories and methods of calculation and simulation of linear electronic circuits, linear electro-mechanical systems, and/or other linear systems
- Shall have acquired knowledge of methods for analysis of linear dynamic systems, including electronic circuits, described by differential equations
- Shall have gained insight into basic feedback theory and its applications in electronic and dynamical systems
- Must master calculations with complex numbers, as used within the field of Electronics
- Shall have knowledge of recognized standards for documentation of electronic circuits, including electrical diagrams, PCB layout, etc.
- Shall be able to demonstrate knowledge of theory and method to the extent of being able to explain and justify the project’s theory and methods, including both selection and deselection.
- Shall master the relevant terminology

Skills
- Shall have understanding of basic theories behind simple electronic components such as resistors, capacitors, operational amplifiers, etc., including calculation of these components
- Shall be able to identify, analyze and formulate issues within the discipline through the use of contextual and technical analysis methods
Shall, based on the above, be able to create requirements and test specifications that enable the completed system to be tested rigorously

Shall be able to use mathematical theories and methods to analyze problems involving linear dynamic components

Shall be able to simulate and design simple analog circuits, allowing specific, desired properties to be achieved, for example, specific transfer functions or amplitude / phase characteristics

Shall be able to design and implement basic analog circuits and demonstrate that these work as intended

Shall be able to document and disseminate knowledge and skills with proper use of terminology, orally and in writing through a project report

Shall be able to analyze and reflect upon his/her own learning process using appropriate methods of analysis and experience from P0 and P1

Shall be able to analyze a technical-scientific problem under consideration of technological and societal contexts, and assess the technological and social consequences of proposed solutions.

Competences

Must be able to demonstrate, independently and in groups, the ability to plan, organize, implement and reflect upon a project that is based on a problem of relevance to society or industry, in which analog electronic devices play a central role

Must have acquired, independently and in groups, the ability to obtain the necessary knowledge of a contextual as well as of technical nature, and be able to formulate models of limited parts of reality to such a level of abstraction that the models can be used in the design, implementation and test of a comprehensive system to meet given requirements

Must be able to evaluate and take responsibility for science and technical solutions in a societal perspective.

Must be able to generalize and reflect upon the experience with project planning and cooperation for the further study acquired during the project work

Type of instruction: Project work with supervision
Exam format: Individual oral examination based on a written report
Semester: 2nd semester  
Prerequisites: Linear algebra  
Purpose: Calculus is the branch of mathematics that studies differential equations and operations such as integration. Differential equations, in turn, describe (among other things) how signals in electric circuits behave  
Objectives: Students who complete the module:

**Knowledge**
- Must have knowledge of definitions, results and techniques within the theory of differentiation and integration of functions of two or more variables
- Must have knowledge of the trigonometric functions and their inverse functions
- Must have knowledge of complex numbers, including rules for computation and their representations
- Must have knowledge of factorization of polynomials over the complex numbers
- Must have knowledge of the complex exponential function, its characteristics and its connection with trigonometric functions
- Must have knowledge of curves in the plane (both rectangular and polar coordinates) and spatial parameterizations, tangent vectors and curvatures of such curves
- Must have knowledge of the theory of second order linear differential equations with constant coefficients

**Skills**
- Must be able to visualize functions of two and three variables using graphs, level curves and level surfaces
- Must be able to determine local and global extrema for functions of two and three variables
- Must be able to determine area, volume, moment of inertia etc. using integration theory
- Must be able to approximate functions of one variable using Taylor's formula, and use linear approximations for functions of two or more variables
- Must be able to perform arithmetic computations with complex numbers
- Must be able to find the roots in the complex quadratic equation and perform factorization of simple polynomials
- Must be able to solve linear second order differential equations with constant coefficients, in general, and with initial conditions
- Must be able to use the concepts, findings and theories introduced in the course to make mathematical deductions in the
context of simple and concrete abstract problems

**Competencies**
- Shall demonstrate development of his/her knowledge of, understanding of, and ability to make use of, mathematical theories and methods within relevant technical fields
- Shall, given certain pre-conditions, be able to make mathematical deductions and arguments based on concepts from multi-variable calculus

Type of instruction: As described in the introduction to Chapter 3.

Exam format: Individual oral or written examination

Introduction to Electrical Engineering
Introduktion til elektriske grundfag

Semester: 2nd semester
Prerequisites: Monitoring and programming (P1) or 1st semester of Electronics and IT or Energy studies

Purpose: To give the students theoretical and practical insight into analog electronic devices, their models, and how to use them in the design of electronic systems

Objectives: Students who complete the module:

Knowledge
- Must have knowledge and understanding of resistive electrical circuits
- Must have knowledge and understanding of operational amplifiers
- Must have knowledge and understanding of analog electronics, including diodes and transistors
- Must have knowledge and understanding of electrical measurement techniques
- Must have knowledge and understanding of laboratory safety of electro technical laboratory experiments
- Must have acquired understanding of the following key concepts:
  - Basic DC Circuit Theory (without energilagrende components), Ohm's law, units, Kirchhoff's laws, circuit reductions (serial and parallel), star-triangle dependent and independent sources, the focal point and mask method, basic operational amplifier couplings, the ideal operational amplifier, Thévenin and Norton theorems, the superposition principle, and the maximum power transfer principle.
  - PN transitions, the diode and its stationary large / small-signal model diode as rectifier, transistor and its use as a linear amplifier, the transistor as a switch, and semiconductor models in simulation software
  - Measurement of current, voltage, power and energy, use of common electrical measuring instruments such as voltmeter, ammeter, and wattmeters, as well as oscilloscopes.
  - Accuracy, complex measurement errors and uncertainty calculations
  - Relevant rules and regulations

Skills
- Must be able to analyze simple and complex electrical DC circuits
- Must be able to use circuit analysis techniques to calculate
currents, voltages, energy and power in DC circuits

- Must be able to use circuit reduction methods to simplify electrical circuit models
- Must be able to apply and analyze electrical DC circuits with diodes and transistors
- Must be able to use analysis methods to design the operational amplifier couplings
- Must be able to plan and execute well-designed, successful electro technical laboratory experiments in a safe and appropriate manner

**Competencies**

- Must be able to handle simple development-oriented situations related to electric circuits and laboratory setups in study- or work-related contexts
- Must be able to independently engage in professional and interdisciplinary collaboration with a professional approach within the context of basic electrical engineering
  
  Must be able to identify his/her own learning needs within basic circuit theory and the electro technical laboratory experiments, and structure such learning accordingly.

**Type of instruction:**

As described in the introduction to Chapter 3. Notice that attendance at lab exercises is mandatory.

**Exam format:**

Individual oral or written examination

**Evaluation criteria:**

Stated in the Framework Provisions
Digital Design & Sensors
Digital design og sensorer

Semester: 2nd semester
Prerequisites: Monitoring and programming (P1)
Purpose: To help the students to acquire knowledge and skills which enable the analysis and designing of basic digital circuits, they acquire knowledge about various sensors and develop skills to use and monitor the signals. To teach the operating principles of various types of sensors and to introduce the concepts & designs for the measurement of electrical and non-electrical quantities.

Objectives: Students who complete the module:

Knowledge:
- Must have knowledge and understanding of basic digital circuits
- Must have knowledge of Boolean algebra and minimali methods
- Must be able to explain the difference between CMOS and TTL circuits
- Must be able to describe the basic principles of bi-stable circuit
- Must have knowledge of Multi-vibrators & Sequential circuits.
- Must have knowledge of Bi-stable Circuits, structure of mono-stable and a-stable circuits.
- Must have knowledge of Mealy and Moore State Machines
- Must have knowledge of Counters and Shift Registers
- Must have knowledge of different sensors
- Must have knowledge of how signal is obtained from different sensors
- Must have knowledge of internal working principal of various sensors.

Skills:
- Must be able to analyze simple and complex digital circuits
- Must be able to design digital circuits which is a central feature of data or electrical engineering
- Must be able to understand the analysis, design and the realization of digital circuits
- Must be able to demonstrate an understanding of relevant concepts, theories and methods of analysis and synthesis of combinational and sequential networks.
- Must be able to apply concepts, theories and methods to describe and analyze a specific problem and explain the theoretical and practical implementation considerations.
- Must be able to outline the main electrical characteristics of logic building blocks.
- Must be able to demonstrate knowledge of different logical networks, including both combinational and sequential
- Must be able to model and synthesize digital circuits.
- Must be able to use Karnaugh Map to simplify circuit design
- Must be able to use measurements terminologies including resolution, sensitivity, accuracy, and uncertainty
- Must be able to use sensors for example for the measurement of temperature, displacement and position, digital encoders, shaft encoders, absolute and relative encoders, linear encoders.
- Must be able to plan and execute well-designed, successful electro technical laboratory experiments in a safe and appropriate manner

**Competencies:**
- Must be able to handle simple development-oriented situations related to digital circuits design, sensors and laboratory setups in study- or work-related contexts
- Must be able to independently engage in professional and interdisciplinary collaboration with a professional approach within the context of digital electronics and sensor measurements
- Must be able to identify his/her own learning needs within digital electronics and sensor technology theory and the electro technical laboratory experiments, and structure such learning accordingly.

**Type of instruction:** As described in the introduction to Chapter 3.

**Exam format:** Individual oral or written examination

**Evaluation criteria:** Stated in the Framework Provisions
Mechatronics

Semester: 3. semester

Prerequisites: Knowledge of Electronics corresponding to Analog Instrumentation (2nd semester)

Purpose: One of the important goals of this Bachelor's Programme is to be able to analyze, design and construct so-called mechatronic systems, i.e., systems that involve both electronic and mechanical components, such as robots, precision machinery, etc. In this project unit, the students investigate the transition from electronic signals to actual, physical motion in greater detail.

Objectives: Students who complete the module:

Knowledge:
- Have knowledge of basic mechanical and electromechanical theories and methods and their uses and limitations
- Have understanding of the function of typical sub-components in a mechatronic system
- Have knowledge of and experience with laboratory work with mechatronic systems.

Skills:
- Must be able to use the project unit's theories and methods to derive models of sub-components in a mechatronic system
- Must be able to use relevant to tools such as frequency characteristics, Laplace transform etc. to analyze the dynamic behaviour of components in a mechatronic system
- Shall be able to analyze results from simulations and laboratory work under the project's theme
- Shall be able to implement computer algorithms to achieve desired behaviour in a computer connected with appropriate hardware (sensors and actuators)

Competencies:
- Must have gained the ability to translate academic knowledge and skills within the fields of basic mechanics and electromechanics to a practical problem, which can be formulated and solved
- Are able to design a mechatronic system, including selecting appropriate actuators and sensors, such that the system can be used to solve the problem formulated above
- Possesses the ability to design and implement algorithms to
control said mechatronic system.

<table>
<thead>
<tr>
<th>Type of instruction:</th>
<th>Project work with supervision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exam format:</td>
<td>Individual oral examination based on a written report</td>
</tr>
</tbody>
</table>
Algorithms and Data Structures

Semester: 3rd semester
Prerequisites: Qualifications corresponding to 2th semester on the Electronics and Computer Engineering Bachelor's Degree study
Purpose: To learn about the fundamental principles of algorithm design, its complexity, and some common data structures employed in a diversity of algorithms.

Objectives: Students who complete the course module

Knowledge
- Must have an understanding of the basic principles of algorithm design and the data structures used.
- Must have an understanding of how to calculate the complexity of algorithms in time and space.
- Must have an understanding of some common algorithms recursive and non-recursive used in sorting, and to processing queues, lists, graphs and trees.
- Shall have an insight into the use and design of basic data structures such as hash tables, stacks, queues, lists, graphs and trees.

Skills:
- Shall be able to implement simple data structures and algorithms
- Shall be able to decide which data structure could be used more efficiently for a specific application

Competencies:
- Shall demonstrate the ability to assess whether or not a computational problem can be efficiently solved using a known algorithm, and if so, be able to design or implement algorithms and data structures to support such a solution
- Shall be able to discuss features and advantages and disadvantages of specific data structures

Type of instruction: As described in the introduction to Chapter 3.
Exam format: Individual oral or written examination
Mathematics
*Matematik*

Semester: 3rd semester  
Prerequisites: Linear Algebra, Calculus  
Purpose: The dynamical behaviour of mechatronic systems is typically described by differential equations. This course supports the semester theme by providing mathematical tools for analyzing such systems in detail.

Objectives: Students who complete the module:

- **Knowledge:**
  - Must have knowledge of important results within vector analysis in 2 and 3 dimensions
  - Shall be able to understand Laplace transformation and use it to solve differential equations.
  - Must have knowledge of complex analytic functions
  - Must have an understanding of power series and Taylor series
  - Must have knowledge of Laurent series and the method of residues integration

- **Skills:**
  - Must be able to use vector analysis, including inner product, vector product, vector functions, scalar functions and fields, as well as elements of vector differential and integral calculus
  - Must have understanding of Fourier series, including concepts such as trigonometric series, periodic functions, even and odd functions, complex Fourier series and forced oscillations resulting from non-sinusoidal input
  - Shall be able to understand and utilize the Laplace transform for analysis of differential equations; specific subjects include:
    - The definition of the Laplace transform.
    - Inverse transformation.
    - Linearity and s-shift.
    - Transformation of common functions, including regular, impulse and step functions.
    - Transformation of derivatives and integrals.
    - Solving Differential Equations
    - Folding and integral equations
    - Differentiation and integration of transformed systems of ordinary differential equations
    - Using Tables
  - Shall be able to use complex analytical functions within the contexts of conformal mappings and complex integrals; specific subjects include:
    - Complex numbers and complex plane
    - Polar form of complex numbers
    - Exponential, trigonometric and hyperbolic functions
    - Logarithmic functions and general power functions
    - Complex Integration: Line integrals in the complex plane
    - The Cauchy integral theorem
**Competencies**

- Shall demonstrate development of his/her knowledge of, understanding of, and ability to make use of, mathematical theories and methods within relevant technical fields
- Shall be able to identify their own learning requirements and structure their own learning within the context of fundamental mathematics.

**Type of instruction:** As described in the introduction to Chapter 3.

**Exam format:** Individual oral or written examination

**Evaluation criteria:** Stated in the Framework Provisions
Micro Processors and Programming

Mikroprocessorer og programmering

Semester: 3rd semester
Prerequisites: Imperative Programming; Sensors, Sampling and Signals
Purpose: Most mechatronic systems include dedicated computers that handle the "intelligent" tasks of guidance, monitoring and control. Typically, such a dedicated computer is connected to/equipped with sensors that allow it to measure important information about current system status and (in some cases) its surroundings. Using these measurements, the dedicated computer executes various algorithms that enables it to determine how to operate the mechatronic system's actuators in response to the immediate situation. Building on the knowledge gained in the 2nd semester course on Sensors, Sampling and Signals, this course aims to provide the students with theories and methods that enable them to design and implement programs for such dedicated computers and use them in a practical mechatronic context.

Objectives: Students who complete the module:

Knowledge:
- Shall have understanding of basic real-time aspects of single-processor system operation, including clock frequency, sampling rate, algorithm processing time etc., as well as how these aspects affect each other
- Must have insight into common micro-processor architecture elements, such as RAM, ALU, registers, buses, etc., as well as how these components interact
- Shall have insight into number representation on digital computers
- Must have basic insight into simple digital filters
- Must be able to use relevant tools such as the complex Z-transform to find a digital implementation of a continuous-time differential equation

Skills:
- Must be able to design algorithms for a chosen micro-processor that satisfy specified timing constraints
- Must be able to use a relevant programming language, along with relevant compilers and linkers, to implement and test said algorithms on said micro-processor
- Must be able to design and implement relevant circuitry to enable a micro-processor to become an integrated part of a mechatronic system

Competencies:
- Are able to design and implement simple, micro-processor-based systems that can be integrated in mechatronic systems and handle fundamental monitoring and control tasks.

Type of instruction: As described in the introduction to Chapter 3.
Exam format: Individual oral or written examination.

Basic Control Engineering
Grundlæggende regulering

Semester: 4th semester
Prerequisites: Qualifications corresponding to 3rd semester on the Electronics and Computer Engineering Bachelor’s Degree study

Purpose: This semester is a direct extension of the third semester, where the theories and methods acquired on the previous semesters are put to more advanced use in the context of a control system. Where the systems treated on the third semester were required to be mechanical, the systems that the students can choose to deal with on this semester can be of any type, for instance a set of sewer pumps, a floor heating system, a medicinal micro-dosage system, a ship attitude control system, a Segway or a diesel engine to name but a few. Control Engineering is basically concerned with making such systems behave according to plan, even if there are external disturbances that would otherwise throw them off course. In order to provide effective control solutions, the students are required to make models of the systems as well as consider the effects of feedback (the control) and noise (the disturbances) in a more rigorous manner than before.

Objectives: Students who complete the module:

Knowledge:
- Must have an understanding of digital and analog transfer functions described via the Laplace and Z-operators, respectively, including features such as poles, zeros, analog and digital implementation, transfer matrices etc.
- Must be able to linearize non-linear system models in order to approximate them by linear models
- Must have insight into real-time aspects in relation to digital systems communicating with other analog and/or digital systems
- Shall have insight into the object-oriented paradigm and its application to modelling of control systems

Skills:
- Must be able to use methods for modelling of physical systems, including electric, electro-mechanical, thermal and fluid dynamical systems, at a level where the resulting models can be utilized in a control system design
- Must have an understanding of exchange and processing of analog signals and discrete data between (sub-)systems, including frequency responses, phase and gain characteristics, sampling, analog and digital filtering etc.
- Must be able to design control laws to achieve specific system requirements, for instance specific phase and gain margin, limited overshoot etc.

Competencies:
- Must be able to construct systems comprising one or several computer-based systems embedded in physical surroundings, involving transformation between analog and digital signal (and corresponding data representations), such that an a priori specified behaviour is achieved by the overall system
- Shall be able to use feedback to reduce effects of disturbances, uncertainties and so forth, as well as be able to specify requirements to and achieve desired system responses for linear systems

Type of instruction: Project work with supervision

Exam format: Individual oral examination based on a written report

Semester: 4th semester
Prerequisites: Linear algebra, calculus and mathematics
Introduction to Electrical Engineering
Purpose: Feedback can be understood as the ability to use measurements of a system's state to correct errors and thereby achieve a desired dynamic responses to given stimuli, such as references and noise. By designing an appropriate feedback - a controller - a system may be forced to take a specific, desirable behaviour. The purpose of this course is to enable the students to use and configure controllers using frequency domain methods, and to configure various controllers and systems described by transfer functions.
Objectives: Students who complete the module:

Knowledge
- Must have knowledge of the modelling of physical systems, provision of operating points and linearization
- Must have an understanding of a system's dynamic and stationary behaviour, including the impact of system type and order, as well as poles and zeros and their influence on the system response
- Must have an understanding of analysis and design using root locus methods
- Must have an understanding of a system's frequency response (open-loop and closed-loop)
- Must have an understanding of relative stability
- Must have insight into design using frequency response techniques
- Must have insight into analogue implementation of controllers
- Must have knowledge of the operation of classical sensors operation (pressure, temperature, position, velocity, acceleration, flow)

Skills
- Shall be able to model and analyze simple dynamical systems, including electrical, mechanical and thermo dynamical systems
- Must be able to develop models of dynamic systems in the form of block diagrams and transfer function features
- Must be able to apply control theory to specify performance criteria
- Shall be able to analyze a system's response and stability using classical methods
- Shall be able to select appropriate controls and predict / evaluate their influence
- Must be able to design and tune basic controllers

Competencies
- Must have gained the ability to translate academic knowledge and skills within the fields of basic modelling and control
engineering to a practical problem, which can be formulated and solved

- Are able to design a control system, such that the system can be used to solve the problem formulated above
- Possesses the ability to design and implement algorithms for said control system.

<table>
<thead>
<tr>
<th>Type of instruction</th>
<th>As described in the introduction to Chapter 3.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exam format</td>
<td>Individual oral or written examination</td>
</tr>
</tbody>
</table>
Object Oriented Software Engineering
Objektorienteret software udvikling

Semester: 4th semester
Prerequisites: Imperative Programming
Purpose: To learn, practice and perfect the art and science of software engineering from design and implementation to delivery; to learn about communication between digital devices; to learn how to develop graphical user interfaces.

Objectives: Students who complete the course module will obtain the following qualifications:

Knowledge
- Understanding OOSE principles: objects, classes and instances; encapsulation; inheritance; virtual functions and polymorphism; overriding and overloading; private vs. public vs. protected class members; templates
- Understanding OOSE elements: object and class definitions; instances, data members and functions; pointers and references; UML
- Understanding OOSE advantages: code comprehension; debugging; modular development and reusability; abstraction
- Knowledge of the development of large projects using an integrated development environment (IDE)
- Understanding concepts of programming for the Internet and its various applications, such as e-mail, WWW, peer-to-peer and file transfer
- Understanding the top three layers in the 5-layer Internet protocol stack: Application, Transport and Network
- Understanding concepts of programming for networks: clients and servers; sockets; establishing a connection; ports; IP address; Ethernet address
- Knowledge on the concepts behind graphical user interfaces (GUIs): common GUI elements; message passing; issues with cross-platform development; multi-threaded programming
- Knowledge on the concepts behind multi-threaded programming: processes and threads; scheduling; bottle necks and deadlock; shared data; mutex locks; the race condition

Skills
- Design and implement an existing solution to a problem using OOSE principles: objects, classes and instances; encapsulation; inheritance; virtual functions and polymorphism; overriding and overloading; private vs. public vs. protected class members; templates
- Design and implement a new solution to an existing problem using OOSE principles that is modular, well-documented and comprehensible

35
• Analyze, interpret and explain pre-existing code and UML
• Analyze and interpret an application programming interface (API)
• Analyze and work out computational complexity of an algorithm
• Understand digital application and communication protocols: HTTP; FTP; SMTP; TCP and UDP
• Implement a program that communicates between two digital devices at the Application and Transport layers (application)
• Explain how to use a common GUI API (understanding)
• Implement a useful GUI with: common GUI elements; message passing; issues with cross-platform development; multi-threaded programming

**Competencies**

• Evaluate and combine adapt pre-existing code, such as libraries and application programming interfaces (APIs)
• Develop OO software to achieve specified goals

**Type of instruction:**
As described in the introduction to Chapter 3.

**Exam format:**
Individual oral or written examination

**Evaluation criteria:**
Stated in the Framework Provisions
**Embedded Software Design**

*Indlejret software design*

**Semester:** 4th semester  
**Prerequisites:** Micro Processors and Programming  
**Purpose:** This course aims to advance the students’ knowledge of software design and implementation for dedicated computers integrated within an inhomogeneous system, so-called *embedded* computers. Such computers are often subject to very specialized requirements, such as minimal system resources, low power consumption and high speed. The course provides tools for understanding and designing software for such devices, including how to interface between them and more general-purpose computers such as PCs.

**Objectives:** Students who complete the module:

*Knowledge:*
- Shall be aware of basic low-level programming primitives such as moving data, adding and multiplying, conditional branches etc., as well as logical operations such as AND, OR, NOT and so forth.
- Shall be aware of how these primitives are mapped into machine-language and assembler code.
- Have understanding of state machines and how they can be used to specify system behaviour.
- Have knowledge of the tasks of an operating system, including inter-process communication, accessing peripheral devices, active process shifts etc.
- Have insight into concepts of hardware/software interaction such as device drivers, use of hardware timers, etc.
- Must have understanding of the basics of algorithms, i.e., fundamental principles such as iteration, induction, recursion etc.
- Must be able to implement simple data communication routines between computers using specified protocols.

*Skills:*
- Shall be able to utilize different types of scheduling principles, e.g., round-robin, fixed priorities and EDF, as well as various schedulability criteria.
- Must be able to design and implement various types of inter-process communication.

*Competencies:*
- Shall be able to design tasks and programs that can operate in an optimal fashion on a specific embedded operating system.
- Are able to understand, consider and utilize more advanced concepts relevant to embedded systems, such as memory consumption, context change timing, pipelining, interrupt handling etc.

**Type of instruction:** As described in the introduction to Chapter 3.
Exam format: Individual oral or written examination

Distributed Computing
Distribueret databehandling

Semester: 5. semester
Prerequisites: Qualifications corresponding to 4th semester on the Electronics and Computer Engineering Bachelor's Degree study

Purpose: Although people are often unaware of it, many computing activities today and in the future are, in fact, parallel and distributed. Indeed, the mass marketing of multicore computers and general-purpose graphics processing units in home and office PCs and laptops means that even common users have the option of performing parallel computing. Furthermore, as almost all but the most dedicated computers are becoming connected to networks, data communication is becoming similarly important. As a consequence, it is no longer sufficient for even basic programmers to acquire only the conventional programming skills for single-processor systems; they must understand the notions of several machines working together on solving problems.

Objectives: Students who complete the project module:

Knowledge:
- Must have an understanding of the basic concepts of data communication, such as medium access, data link, protocols and routing
- Must have insight into the notion of data distribution, including distributed storage
- Shall have an insight into loosely and tightly coupled computer systems
- Must have insight into the basic notions of parallel computing

Skills:
- Shall be able to implement digital filters to solve specific signal processing problems on multi-processor systems
- Shall be able to design and implement software to support communication on a distributed system, for instance using Object-oriented Software Engineering

Competencies:
- Shall demonstrate the ability to assess whether or not a computational problem can be efficiently solved using parallel and distributed computing, and if so, be able to design simple algorithms and architectures to support such a solution
- Shall be able to demonstrate how to perform distributed data storage and retrieval in a distributed system, and discuss advantages and disadvantages thereof

Type of instruction: Project work with supervision
Exam format: Individual oral examination based on a written report

Numerical Methods
Numeriske metoder

Semester: 5. semester
Prerequisites: Mathematics
Purpose: Not all mathematical and engineering problems are simple enough to solve analytically. The purpose of this course is to provide the students with tools and methodologies to approach those problems that cannot be solved with 'pen and paper', but requires numerical approximations.

Objectives: Students who complete the module:

Knowledge:
- Must have understanding of how to solve partial differential equations with analytic methods
- Must have understanding of different numerical methods
- Must have an understanding of finite difference, finite volume and finite element method

Skills:
- Shall be able to use analytical methods for solving partial differential equations, in particular the method of Separation of Variables and D’Alembert’s Principle.
- Shall be able to utilize numerical methods to solve mathematical problems, including:
  - Systems of linear equations, Gauss elimination, and factorization-based method.
  - Iterative solution of systems of linear equations, e.g., Gauss-Seidel.
  - Ill-conditioned systems of linear equations.
  - Matrix Eigen value problems.
  - Solving systems of non-linear equations.
  - Interpolation and splines.
  - Numerical solution of definite integrals.
  - Numerical solution of first- and second-order differential equations.
- Must be able to utilize finite difference methods for solution of partial differential equations, including:
  - Approximation by finite differences.
  - Elliptical equations.
  - Dirichlet and Neumann boundary value problems.
  - Parabolic equations.
  - Explicit and implicit method, the Theta-method.
  - Hyperbolic equations.
  - Relationship with finite volume methods.
- Shall have understanding of finite element methods for solving partial differential equations.

Competencies:
- Shall demonstrate development of his/her knowledge of, understanding of, and ability to make use of, mathematical theories and methods related to solving technical problems using numerical methods.
- Shall be able to identify their own learning requirements and
structure their own learning within the context of numerical mathematics.

Type of instruction: As described in the introduction to Chapter 3.

Exam format: Individual oral or written examination

Signal Processing
Signalbehandling

Semester: 5th semester
Prerequisites: Mathematics, Micro Processors and Programming, Fundamental Control Theory and Modelling
Purpose: Analysis and filtering of signals is one of the fundamental disciplines for all specializations within electronic systems; it is used within Control Engineering, Communications, Multi-media technology, and so forth. This course aims to support students in understanding key concepts, theories and methods of analysis and filtering of analog and digital signals, and apply theories and methods of analysis and filtering of analog and digital signals.

Objectives: Students who complete the course:
Knowledge:
- Must have knowledge about theories and methods for analysis and processing of physical signals on a computer
- Must have knowledge about theories and methods for spectral estimation
- Must have knowledge about theories and methods for design of digital filters (IIR/FIR)
- Must have an understanding of the limitations of said theories and methods
- Must have knowledge about the interplay between analysis of signals in the time and frequency domains
- Must have knowledge about theories and methods for transformation between different domains

Skills:
- Shall be able to utilize tools for analysis, design and simulation of digital signal processing systems
- Must be able to apply theories and methods for spectral estimation including DFT / FFT
- Must be able to demonstrate the correlation between frequency resolution, window functions and zero-padding
- Must be able to apply theories and methods for design of digital filters
- Must be able to implement IIR filters using bilinear transforms and impulse invariant methods
- Must be able to explain the importance of phase linearity and group runtime
- Must be able to design FIR filters using windowing methods
- Must be able to explain the relationship between the pole/zero plots and frequency responses of digital filters
- Must be able to implement filters in practice, making use of appropriate filter structures, quantization, and scaling.

Competencies:
- Shall be able to discuss fundamental theories and methods for analysis and processing of digital signals, using correct
terminology
• Shall be able to assess opportunities and limitations in connection with practical application of said theories and methods

Type of instruction: As described in the introduction to Chapter 3.
Exam format: Individual oral or written examination
Operating Systems and Network Data Communication
Styresystemer, netværk og datakommunikation

Semester: 5th semester
Prerequisites: Knowledge, skills and competencies equivalent to having passed the fourth semester

Purpose: The course will assist the students in acquiring competencies within operating systems, network and data communication. All Bachelor Programs within Data & Electronics require competencies within operating systems, network and data communication

Objectives: Students who complete the course:

Knowledge:
- To give students an understanding of principles for and handling of systems characterized by numerous cooperating and communicating processes
- To give students a comprehension of principles and techniques of modern data network systems and their communications

Skills:
- Must be able to understand basic process concepts
- Must be able to comprehend mutual exclusion and synchronization including internal process/thread communication
- Must be able to understand and apply scheduling techniques
- Must be able to understand memory management
- Must be able to understand OSI models and protocol concepts
- Must be able to understand Layer 1 and 2 including basic data-transmission, MAC, LLC, HDLC
- Must be able to understand network protocols and their programming, including IP, UDP, TCP, Sockets, and RPC.
- Must be able to use concepts from the OSI model, including the MAC, network, transport and application layers.
- Must be able to use TCP / IP protocol stack and be able to assess functions in the network, transport and application layers, including Quality of Service mechanisms.

Competencies:
- Shall be able to analyze an operating system and describe/design a communication network for operating systems as well as for other systems.

Type of instruction: As described in the introduction to Chapter 3.

Exam format: Individual oral or written examination.

BSc Project (Control Engineering)
Bachelorprojekt (Reguleringssteknik)

Semester: 6th semester
Prerequisites: Knowledge, skills and competencies equivalent to having passed the fifth semester

Purpose: The project must be based on a physical process. The process can be mechanical, thermal, electrical, biologic or chemical. A dynamic model of the process has to be developed. The model has to be adjusted and verified through measurements. Demands as well in the time as in the frequency domain has to listed. Using the dynamic model a classic and a state space controller are designed and implemented on the process. The controllers have to be evaluated and compared to the demands.

Objectives: Students who complete the project module:

Knowledge:
- Must have knowledge of how to design control systems
- Must be able to understand and implement dynamic modelling, classic controller design and model based control.

Skills:
- Must be able to analyze dynamic systems in time and frequency domain
- Must be able to analyze and apply model based controller design methods as state space control and pole placement
- Must be able to apply mechanical, thermodynamic, biological or chemical equations to develop a dynamic model
- Must be able to analyze and apply methods for simulation of dynamic systems
- Must be able to synthesize, i.e., implement and test dynamic models and controllers
- Must be able to evaluate industrial control and supervision methods.
- Must be able to communicate the above knowledge and skills (using proper terminology of the field), both orally and in a written report

Competencies:
- Must be able to select and extract relevant features and apply these in a new context
- Must be able to plan, structure and execute a project, within the subject-field of this project module

Type of instruction: Project work with supervision
Exam format: Individual oral examination based on a written report
BSc Project (Embedded Real-Time Signal Processing)

Bachelorprojekt (Indlejret realtidssignalbehandling)

Semester: 6. semester
Prerequisites: Knowledge, skills and competencies equivalent to having passed the fifth semester
Purpose: An embedded system is defined as an electronic system which is based on a computer, but the system is not in itself a computer, e.g., like a PC. According to this definition, an average person is interacting with hundreds of embedded systems on a daily basis, typically in terms of audio/video applications, wireless/mobile communication, gaming consoles, household machines, automotive and medical devices, as well as avionic and satellite based systems. In most cases, the computer embedded in such devices is conducting some kind of signal processing, i.e., an analogue signal is registered by a sensor and sampled, and next the signal is either analyzed or modified digitally by software executing on the computer. Eventually the resulting signal is finally re-converted back to the analogue domain. An interesting feature of this overall process is that in most cases it must be conducted in hard real-time, i.e., the processing must be completed within a predefined and fixed time interval. Otherwise, the system will fail, potentially leading to hazardous situations. Taking the outset in a real-life problem/application, the purpose of this project module is to specify, design, simulate, implement, test and document (part of) an embedded real-time signal processing system. In this context, the algorithm(s) which are to perform the signal processing have to be developed, simulated/evaluated (preferably using C or Matlab) and optimized, and next compiled into an executable code which can run in real-time on a programmable digital signal processor. The overall design parameters may include, but are not limited to execution time, code size, numerical robustness, and eventually energy consumption. Primarily, the project will focus on the signal processing theories and algorithms, as well as the development of optimal source- and object codes using commercially available development boards/tools, thus excluding the design and implementation of user-specific hardware.

Objectives: Students who complete the project module:

Knowledge:
- Must have knowledge about the building blocks used in a generic embedded real-time digital signal processing (DSP) system, their mutual interaction and interfaces, as well as relevant performance parameters.
- Must have knowledge about theories and methods used to design numerically robust and resource optimal DSP algorithms suitable for being executed real-time on programmable digital signal processors.

Skills:
- Must be able to analyze a technical problem which naturally finds its solution in terms of real-time digital signal processing. Secondly, to formulate a set of specifications for the algorithms to be developed, and possibly also for the hardware/software
platform to be used.

- Must be able to apply various methods to design, simulate, and evaluate DSP algorithms according to the specifications for functionality and numerical properties. C or Matlab are candidates for executable specifications and for simulation purposes.
- Must be able to analyze DSP algorithms from a computational complexity, structural, and data flow oriented point of view in order to specify architectural requirements for a software programmable target platform.
- Must be able to apply design tools, such as C compilers (possibly using in-line assembly language), in order to synthesize and optimize real-time executable code for DSP algorithms.
- Must be able to evaluate 1) an overall system solution, and 2) the design methods applied to derive the solution. This must be done in terms of relevant metrics such as execution time, memory usage, numerical robustness, and energy consumption. Secondly, from a micro-computer architectural point of view, the students must be able to evaluate the match between algorithms and architectures.
- Must be able to communicate the above mentioned knowledge and skills (using the terminology of the domain), both orally and in a written report.

**Competencies:**

- Must be able to identify, design, implement, and evaluate a viable solution for an embedded real-time signal processing system in a real-life context.
- Must be able to plan, structure, and conduct a project within the scientific subject of this project module.

**Type of instruction:** Project work with supervision

**Exam format:** Individual oral examination based on a written report

**Evaluation criteria:** Stated in the Framework Provisions
BSc Project (Distributed Systems)
Bachelorprojekt (Distribuerede systemer)

Prerequisites: Qualifications corresponding to 5th semester on the Electronics and Computer Engineering Bachelor’s Degree study

Purpose: The development of the internet in the 1990s has made feasible to interconnect computing systems that are designed to provide some services or perform some processing on demand. These geographically distributed systems are of widespread use today. Therefore, it is necessary to understand the principles used to design these systems, how they work and how they can be used.

Objectives: Students who complete the project module:

Knowledge:
- Must have an understanding of the basic principles of distributed computing
- Must have an understanding of the main architectures used in distributed systems
- Shall have an insight into the use of distributed methods and objects
- Must have insight into the basic notions of performance improving techniques for distributed systems
- Must have an understanding of the main paradigms used to design distributed systems
- Must have an understanding on how to use different computer languages to design distributed systems

Skills:
- Shall be able to implement a distributed system using middleware
- Shall be able to design and implement simple multitier client server systems
- Shall be able to use different computer languages to design a distributed system

Competencies:
- Shall demonstrate the ability to assess whether or not a computational problem can be efficiently solved using distributed computing, and if so, be able to design simple algorithms and architectures to support such a solution
- Shall be able to demonstrate how to perform distributed data processing and services using sockets or middleware and discuss advantages and disadvantages thereof

Type of instruction: Project work with supervision
Exam format: Individual oral examination based on a written report
BSc Project (Programming Languages)
Bachelorprojekt (Programmeringssprog)

Semester: 6. semester
Prerequisites: Qualifications corresponding to 5th semester on the Electronics and Computer Engineering Bachelor's Degree study
Purpose: Modern computer languages are multiparadigm. They employ techniques and concepts used in pure object oriented, functional and logical languages. Therefore it is necessary for programmers to understand the concepts employed in these languages to fully utilize these features. Additionally, given that current architectures of multicore processors are underutilized by current single threaded or even multithreaded languages, it is necessary for programmers to understand how to exploit these architectures more efficiently.

Objectives: Students who complete the project module:
Knowledge:
- Must have an understanding of the basic principles used in pure object oriented, functional or logical languages.
- Must have an understanding of the basic principles of parallel programming.
- Shall have an insight into the techniques used in modern languages to implement features from functional or logical languages.

Skills:
- Shall be able to write applications using multiparadigm computer languages with features from pure object oriented, functional, logical or parallel processing languages and libraries.
- Shall be able to identify which language is more suitable for a specific application.

Competencies:
- Shall demonstrate the ability to use object oriented, declarative or logical languages to solve a problem.
- Shall demonstrate the ability to use parallel programming languages and libraries to solve more efficiently a problem.

Type of instruction: Project work with supervision
Exam format: Individual oral examination based on a written report
Introduction to Probability Theory and Statistics

Semester: 6. semester
Prerequisites: Fundamentals in Linear Algebra, Calculus, and Fourier Theory
Purpose: After attending the course the students have developed the engineering intuition of the fundamental concepts and results of Probability, Statistics, and Stochastic Processes. They are able to apply the taught material to model and solve simple engineering problems involving randomness.

Objectives: Students who complete the course:

Knowledge:
- Must have knowledge about the concept of probability spaces
- Must have knowledge about the conceptual models of estimation and hypothesis testing
- Must be able to understand the basic concepts of probability theory, i.e., probability of events, random variables, etc.
- Must be able to understand basic concepts of statistics such as binary hypothesis testing.

Skills:
- Must be able to apply/compute
  - Bayes rule in simple contexts
  - The probability that Binomial, Poisson, and Gaussian random variables take values in a specified interval
  - The mean and variance of Binomial, Poisson, and Gaussian random variables
  - The marginal distributions of multi-variate Gaussian variables
- Must be able to apply and interpret
  - ML-estimation in simple contexts involving the Binomial, Poisson, and Gaussian distribution
  - Binary-hypothesis tests in simple contexts involving the Binomial, Poisson, and Gaussian distribution

Competencies:
- Must be able to apply the general concepts of Probability Theory and Statistics in a new, simple context. This includes choosing suitable methods, evaluating outcomes, and drawing the appropriate conclusions.

Type of instruction: As described in the introduction to Chapter 3.
Exam format: Individual oral or written examination
Matrix Computation and Convex Optimization
*Matriksberegninger og convex optimering*

Semester: 6th semester
Prerequisites: Linear Algebra, Calculus
Purpose: Engineering systems and design problems can often be compactly described, analyzed and manipulated using matrices and vectors. Moreover, tractable solutions to design problems can be obtained by casting the design problems as optimization problems. For the class of linear and quadratic problems, the solutions can be obtained by solving systems of equations. In computer programs, this is achieved via matrix factorizations. For the larger class of convex problems, no closed-form solution may exist and numerical methods must be applied. This course aims at teaching numerically robust methods for solving systems of equations and, more generally, convex optimization problems, including also standard constrained problems.

Objectives: Students who complete the course:

*Knowledge:*
- Must have knowledge about convex functions and sets, norms, special matrices
- Must have understanding of how to classify and solve systems of equations and convex optimization problems
- Must have understanding of numerical aspects of solving systems of equations and convex optimization problems
- Must have knowledge about Lagrange multipliers
- Must have understanding of matrix factorizations and their properties

*Skills:*
- Must be able to identify optimization problems and cast them into standard form
- Must be able to identify types of extrema (minima, maxima, local, global, etc.)
- Must be able to apply Eigen value and singular value decomposition to relevant matrix problems
- Must have understanding of state space descriptions of systems of linear differential equations
- Shall be able to apply numerically robust methods to solve systems of equations
- Shall be able to apply and implement the following numerical optimization methods to unconstrained optimization problems: Steepest Descent, Newton's method, Gauss-Newton method
- Shall be able to apply and interpret least-squares solutions when solving over-determined systems of equations
- Shall be able to apply the Lagrange multiplier method to constrained convex optimization problems

*Competencies:*

51
- Are able to apply linear algebra theory to analyse engineering systems in their field
- Can state and analyze engineering design problems in their field as systems of equations or standard optimization problems
- Are able to select appropriate matrix factorization or numerical optimization methods to solve engineering design problems in their field

Type of instruction: As described in the introduction to Chapter 3.
Exam format: Individual oral or written examination
Database Systems, Languages and Compilers

*Database systemer, sprog og compilere*

**Semester:** 6. semester

**Prerequisites:** Algorithm & Data Structures, Object Oriented Software Engineering

**Purpose:** All Bachelor Programs within Data & Electronics require competencies within database design. To help the students in acquiring competencies within database design, formal languages, automata theory and compiler design. To help the students to acquire knowledge about basic principles of programming languages and understanding of techniques for description and translation of languages in general.

**Objectives:** Students who complete the course:

**Knowledge:**

- Must have knowledge of basic concepts and principles of database systems
- Must have knowledge of ER-models
- Must have knowledge of converting to relational models
- Must have knowledge of SQL
- Must understand design, characterization and design of programming languages, translators, interpreters and similar tools
- Must understanding fundamental issues relating to programming languages and their applications.
- Must have knowledge and basic concepts about formal languages, interpreters and compilers
- Must have knowledge of Finite Automata and Machines
- Must have knowledge of Scanners & Parsers
- Must have knowledge of Semantic analyzer
- Must have knowledge of Run time environment
- Must have knowledge of Top Down & Bottom Up Parsers

**Skills:**

- Must be able to apply principles for design of simple relational database systems.
- Must be able understanding the basic theoretical aspects of programming languages: Machines, language theory, compilers etc.
- Must be able to demonstrate learning at the application level, tools for construction of dedicated formal languages.
- Must be able to demonstrate knowledge and overview of the relevant techniques and concepts in language design and translator construction
- Must be able to describe, analyze and implement programming, describe the individual phases and the relationship between the phases of a translator
- Must be able use correct terminology reasoning about data logically and with the relevant concepts and techniques
Type of instruction: As described in the introduction to Chapter 3.

Exam format: Individual oral or written examination

**Distributed Systems**  
*Distribueret systemer*

**Semester:** 6th semester  
**Prerequisites:** Qualifications corresponding to 5th semester on the Electronics and Computer Engineering Bachelor's Degree study  
**Purpose:** Given that distributed systems are widespread it is necessary for programmers to understand how distributed systems may be designed using high level abstractions of data communications, objects and methods.

**Objectives:**

Students who complete the course module

*Knowledge*
- Must have an understanding of the basic principles of distributed computing
- Must have an understanding of the main architectures used in distributed systems
- Shall have an insight into the use of distributed methods and objects
- Must have insight into the basic notions of performance improving techniques for distributed systems

*Skills:*
- Shall be able to implement a simple distributed system using middleware
- Shall be able to design and implement simple multitier client server systems

*Competencies:*
- Shall demonstrate the ability to assess whether or not a computational problem can be efficiently solved using distributed computing, and if so, be able to design simple algorithms and architectures to support such a solution
- Shall be able to demonstrate how to perform distributed data processing and services using sockets or middleware and discuss advantages and disadvantages thereof

**Type of instruction:** As described in the introduction to Chapter 3.  
**Exam format:** Individual oral or written examination  
**Evaluation criteria:** Stated in the Framework Provisions
Chapter 4: Entry into Force, Interim Provisions and Revision

The curriculum is approved by the Dean of the Faculty of Engineering and Science and enters into force as of September 2011.

In accordance with the Framework Provisions and the Handbook on Quality Management for the Faculty of Engineering and Science and The Faculty of Medicine at Aalborg University, the curriculum must be revised no later than 5 years after its entry into force.

Chapter 5: Other Provisions

5.1 Rules concerning written work, including the Bachelor’s project

In the assessment of all written work, regardless of the language it is written in, weight is also given to the student's formulation and spelling ability, in addition to the academic content. Orthographic and grammatical correctness as well as stylistic proficiency are taken as a basis for the evaluation of language performance. Language performance must always be included as an independent dimension of the total evaluation. However, no examination can be assessed as ’Pass’ on the basis of good language performance alone; similarly, an examination normally cannot be assessed as ’Fail’ on the basis of poor language performance alone.

The Board of Studies can grant exemption from this in special cases (e.g., dyslexia or a native language other than Danish).

The Bachelor’s project must include an English summary.\(^1\) If the project is written in English, the summary must be in Danish.\(^2\) The summary must be at least 1 page and not more than 2 pages (this is not included in any fixed minimum and maximum number of pages per student). The summary is included in the evaluation of the project as a whole.

5.2 Rules concerning credit transfer (merit), including the possibility for choice of modules that are part of another programme at a university in Denmark or abroad

In the individual case, the Board of Studies can approve successfully completed (passed) programme elements from other Master’s programmes in lieu of programme elements in this programme (credit transfer). The Board of Studies can also approve successfully completed (passed) programme elements from another Danish programme or a programme outside of Denmark at the same level in lieu of programme elements within this curriculum. Decisions on credit transfer are made by the Board of Studies based on an academic assessment. See the Framework Provisions for the rules on credit transfer.

5.3 Rules concerning the progress of the Bachelor’s programme

The student must participate in all first year examinations by the end of the first year of study in the Bachelor’s programme, in order to be able to continue the programme. The first year of study must be passed by the end of the second year of study, in order that the student can continue his/her Bachelor's programme.

In special cases, however, there may be exemption from the above if the student has been on a leave of absence. Leave is granted during first year of study only in the event of maternity, adoption, military service, UN service or where there are exceptional circumstances.

---

\(^1\) Or another foreign language (French, Spanish or German) upon approval by the Board of Studies.

\(^2\) The Board of Studies can grant exemption from this.
5.4 Rules concerning the completion of the Bachelor’s programme
The Bachelor’s programme must be completed no later than six years after it was begun.

5.5 Special project process
In the 3rd, 4th and 5th semesters, the student can upon application, design an educational programme where the project work is replaced by other study activities; cf. the Framework Provisions section 9.3.1.

5.6 Rules for examinations
The rules for examinations are stated in the Examination Policies and Procedures published by the Faculty of Engineering and Science on their website.

5.7 Exemption
In exceptional circumstances, the Board of Studies study can grant exemption from those parts of the curriculum that are not stipulated by law or ministerial order. Exemption regarding an examination applies to the immediate examination.

5.8 Rules and requirements for the reading of texts
It is assumed that the student can read academic texts in his or her native language as well as in English and use reference works etc. in other European languages.

5.9 Additional information
The current version of the curriculum is published on the Board of Studies’ website, including more detailed information about the programme, including exams.