Mechatronics is an engineering discipline to study the synergistic combination of mechanical engineering, electronics engineering, control engineering, and computer engineering.

This course covers the fundamental areas of science and technology on which a mechatronics design is based. This includes mathematical modeling of complex dynamical systems, analysis of mathematical models using computer simulations, measurement systems (sensors and signal conditioners), actuators, continuous-time controller design and its real-time digital implementation, and networked control systems. The focus is on the role of each of these areas in the overall design process and how these key areas are integrated to form a successful mechatronics system design.

The instructional objectives are:
- To enable students understanding the modern mechatronics components.
- To present the underlying principles and alternatives for mechatronics systems design.
- To provide students with hands-on experience of mechatronics technology for diverse applications.
- To develop the student’s ability to evaluate appropriate technology and devise realistic industrial systems.
295902 - ISCA - Implementation of Automatic Control System
## Content

<table>
<thead>
<tr>
<th>Section</th>
<th>Learning time:</th>
<th>Description:</th>
</tr>
</thead>
</table>
| **1. Course Presentation** | 10h | **1.1 Course contents and syllabus.**  
**1.2 Involved projects - Description.**  
**1.3 Rules and timetable.** |
| **2. Introduction to Mechatronics Systems Design.** | 20h | **2.1 Components of mechatronics systems.**  
**2.2 Motion control systems.**  
**2.3 Servomotors, Stepper Motors, and Actuators for Motion Control.**  
**2.4 Stationary and Mobile robots.**  
**2.5 Linkages: Drives and Mechanisms.**  
**2.6 System integration.** |
| **3. Dynamic systems Modelling.** | 30h | **3.1 Example of Models.**  
**3.2 Principles of Physical modelling.**  
**3.2 Parameters identification.**  
**3.3 Model simulation.** |

### Description:
- 4.1 Controllers types.
- 4.2 Design in Time Domain.
- 4.3 Design in frequency.

### Learning time:
- Theory classes: 8h
- Laboratory classes: 8h
- Self study: 24h

## 5. ARM-based Microcontrollers.

### Description:
- 5.1 ARM Cortex-M0+ Processors.
- 5.2 Interrupts and Low Power Features.
- 5.3 CMSIS and peripherals.

### Learning time:
- Theory classes: 6h
- Laboratory classes: 6h
- Self study: 18h

## 6. From the System to the microcontroller.

### Description:
- 6.1 Code generation for embedded applications.
- 6.2 Code generation from MATLAB/SIMULINK to C/C++.
- 6.3 Workflow for code generation.
- 6.4 Optimization Strategies
- 6.5 Controlling C Code Style.
- 6.6 Deploy and Test Executable Program.

### Learning time:
- Theory classes: 4h
- Laboratory classes: 4h
- Self study: 12h
Qualification system

The final course mark is based on four evaluations:

1. Description and scope of the work (20%).
2. Development and evolution of the work during the course (25%).
3. Project Presentation (25%).
4. Technical report (30%).

According to the specific EEBE academic regulations, sections 2.2.b and 2.2.c, this subject is considered as continuous assessment methodology and, therefore, is not subject to reevaluation.

Bibliography

Basic:


MATLAB Coder. User's guide.