COURSE DESCRIPTION CARD - SYLLABUS

Course name
Intelligent control engineering design [S2AiR2-ISAiR>TIS]

Course

Field of study Automatic Control and Robotics	Year/Semester 2/3
Area of study (specialization) Intelligent Control and Robotic Systems	Profile of study general academic
Level of study second-cycle	Course offered in Polish
Form of study full-time	Requirements compulsory

Number of hours

Lecture 30	Laboratory classes 30	Other 0
Tutorials 0	Projects/seminars 0	

Number of credit points

4,00

Coordinators	Lecturers
dr hab. inż. Konrad Urbański konrad.urbanski@put.poznan.pl	

Prerequisites

A student beginning this course should have knowledge of automation and robotics corresponding to level 6 of the Polish Qualification Framework, in particular, knowledge of the basics of automation, matrix operations and programming skills. They should also understand the need to broaden their competences and be ready to cooperate in a team.

Course objective

To familiarize students with the methods of programming, simulation and analysis of selected control methods and structures in selected operating systems and programming environments. To familiarize students with configuration methods and basic functions and capabilities of the system and programming environments used.

Course-related learning outcomes

Knowledge
has specialised knowledge of remote, distributed, real-time systems and network techniques; understands the design methodology of specialised analogue and digital electronic systems; has a structured and in-depth knowledge of specialised microprocessor systems for control and measurement systems;

Skills
is able to select and integrate elements of a specialised measurement and control system including: control unit, execution system, measurement system and peripheral and communication modules; is able to construct an algorithm to solve a complex and non-typical engineering task and a simple research problem and implement, test and run it in a selected programming environment for selected operating systems; is able to construct an algorithm to solve a complex, non-typical measurement and computational-control task and implement, test and run it in a selected programming environment on a microprocessor platform;
Social competences
is aware of the need to approach technical matters in a professional manner, to study the documentation meticulously and to understand the environmental conditions in which the equipment and its components may operate;

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:
Lecture: exam
Laboratory: checking the ability to create control structures, selection and determination of parameters of control modules and analysis of their operation

Programme content

Preparation and operation of the environment for the research tools used
Methods of selecting controller parameters and analyzing the quality of operation of selected control structures

Course topics

Preparation of programming tools: installation and configuration of a selected Linux-based system, installation and configuration of the programming environment for the Python language (auxiliary programs, software modules, libraries: communication, arithmetic, control, visualization, computational intelligence, including TensorFlow, etc.). Modelling and launching of the selected controller structures provided in the modules, creating models of control objects. Start-up of control structures, analysis of their correctness.

Issues:
selection of the regulator settings according to specific criteria
IMC - internal model control
SP - Smith predictor
MPC - model predictive control
KF - Kalman filter, implementation in control systems
SSN - artificial neural networks - implementation
TF, Keras and deep networks - training and implementation
impact of delays in the control loop
control of devices using Python

Teaching methods

Lectures with multimedia presentation (including: drawings, photos, animations, sound, films) supplemented by examples given on the board
Lectures conducted in an interactive way with formulation of questions to a group of students
Presentation of a new topic preceded by a reminder of related content known to students from other subjects

Laboratories:
- working in teams
- computational experiments

Bibliography

Basic:
1. Online tutorials for the current version of Python 3.x

2. Documentation of selected Python packages for version 3.x
3. Documentation of Keras and TensorFlow libraries
4. PID Controllers : Theory, Design, and Tuning, 2nd Edition, K.J. Astrom, T. Hagglund, 1995
5. Control system design guide, G. Ellis, Elsevier 2004

Additional:
1. Automate the Boring Stuff with Python, A. Sweigart, latest edition
2. Python: wprowadzenie, M. Lutz, Helion, wydanie jak najnowsze
3. Python dla każdego. Podstawy programowania, M. Dawson, wydanie jak najnowsze
4. Deep Learning with Python , F. Chollet, Manning Pub. Co. 2018

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,00
Classes requiring direct contact with the teacher	60	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	40	2,00