SYLLABUS

1. Program Information

1.1 Higher education institution	Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Department of Automation
1.4 Field of study	Automation, Applied Informatics and Intelligent Systems
1.5 Cycle of studies	Bachelor
1.6 Study Programme/Qualification	Intelligent Automation Systems (dual, in English language)
1.7 Form of education	IF – full-time education
1.8 Course code	56.20

2. Course information

2.1 Course title	**Real-Time Simulation and Validation of Control Systems**				
2.2 Course lecturer	*Conf dr. ing. Mirela Dobra – mirela.trusca@aut.utcluj.ro*				
2.3 Seminar / Laboratory / Project Lecturer	*Ing. Ciprian Iakab (Emerson)*				
2.4 Year of study	4	2.5 Semester	2	2.6 Type of assessment	E
2.7 Course status	Formative category *(DF, DS, DC)*				DS
	Optionality *(DOB, DOP, DFac)*				DOP

3.Total estimated time

3.1 Number of hours per week	4	of which:	HEI	Lecture	2	Seminar	0	Laboratory	0	Project	0
			CO		0		0		2		0
3.2 Number of hours per semester	56	of which:	HEI	Lecture	28	Seminar	0	Laboratory	0	Project	0
			CO		0		0		28		0

3.3 Distribution of time allocation (hours per semester) for:	HEI	CO
(a) Study based on textbook, course support, bibliography, and notes	20	
(b) Additional documentation in library, specialized electronic platforms, and fieldwork	20	10
(c) Preparation of seminars/laboratories, assignments, papers, portfolios and essays		10
(d) Tutoring	4	
(e) Examinations	3	2
(f) Other activities:		
3.4 Total individual study hours (sum (3.3(a)... 3.3(f)))	47	22
3.5 Total hours per semester (3.2+3.4)	75	50
3.6 Number of credits per semester	3	2

(HEI = Higher Education Institution, CO = Company)

4. Prerequisites (where applicable)

4.1 Curriculum Prerequisites	• Control Engineering • Embedded Systems • Computer Programming and Algorithm Design • Process Modeling
4.2 Competency Prerequisites	• Basic analog/digital electronics • Understanding of feedback control systems • Proficiency in MATLAB scripting • Basic C/C++ for embedded code development

5. Conditions (where applicable)

5.1. Course Organization Conditions	• Course attendance > 50%

5.2. Seminar / Laboratory / Project organization conditions	• Laboratory is mandatory

6. Specific Competencies Acquired

Professional Competencies	• PC02 Analyse test data • PC05 Conduct quality control analysis • PC07 Demonstrate disciplinary expertise • PC10 Develop electronic test procedures • PC11 Develop mechatronic test procedures • PC16 Monitor manufacturing quality standards • PC21 Report analysis results • PC26 Use information technology tools • PC28 Use testing equipment
Transversal Competencie	• TC01 Apply knowledge of science, technology and engineering • TC02 Think analytically • TC03 Demonstrate responsibility

7. Learning outcomes

Knowledge:	• Understand the differences and relationships between: • Model-in-the-Loop (MIL) / Software-in-the-Loop (SIL) / Processor-in-the-Loop (PIL) / Hardware-in-the-Loop (HIL) • Comprehend the full model-based design (MBD) workflow for embedded systems • Understand real-time systems constraints, timing analysis, and execution scheduling
Skills:	• Develop and deploy C/C++ code for microcontrollers (e.g., TI C2000); Use toolchains like Code Composer Studio and Embedded Coder • Generate real-time code from Simulink models; deploy to embedded targets or HIL platforms (dSPACE, Speedgoat, etc.)
Responsibility and autonomy:	• Independently carry out modeling, simulation, and code development for control systems. • Apply engineering judgment when choosing between simulation, SIL, and HIL approaches.

8. Course Objectives

8.1 General objective of the course	• Understand the principles and workflows of SIL and HIL. • Design and implement SIL and HIL simulations.
8.2 Specific objectives	• Integrate control systems with virtual and real-time environments. • Use commercial tools like MATLAB/Simulink, dSPACE, NI VeriStand, or Speedgoat.

9. Contents

9.1 Lectures	No. of hours	Teaching methods	Obs.
1. Introduction to SIL and HIL. Definition and comparison of MIL, SIL, PIL, and HIL. Importance in control system development. Overview of real-time systems	2	Comprehensive slides	

2. Control System Design Process. Model-based design. Verification and validation. From simulation to real-time testing	2	Blackboard annotations Oriented discussions on the subject	
3. System Modeling. Modeling in MATLAB/Simulink. Creating plant and controller models. Linear vs. nonlinear systems	2		
4. Software-in-the-Loop (SIL) Basics. SIL architecture. SIL implementation using Simulink. Code generation and verification	2		
5. SIL Testing and Automation. Unit testing and integration testing. MATLAB scripting for automated testing. Coverage and analysis	2		
6. Hardware-in-the-Loop (HIL) Basics. HIL architecture and components. Real-time simulators (dSPACE, Speedgoat, NI PXI). Communication interfaces (CAN, UART, Ethernet)	2		
7. Real-Time Simulation Platforms. Introduction to real-time operating systems (RTOS). Overview of HIL toolchains. Data acquisition and I/O handling.	2		
8. Code Generation and Deployment. Embedded Coder / TargetLink overview. From Simulink to embedded code. Deployment on HIL platforms	2		
9. HIL System Integration. Connecting physical hardware (ECU, sensors). Safety and signal conditioning. Debugging and validation	2		
10. Case Study – Aerospace or Robotics. UAV control with SIL/HIL.	2		
11. Sensor fusion testing. Integration with IMUs, GPS	2		
12. Advanced Topics. Co-simulation with FMI/FMU. FPGA-based HIL.	2		
13. Digital twin concept	2		
14. Live demo of HIL/SIL system. Explain test strategy, results, and learning outcomes	2		

Bibliography

1. Nise, Norman S. Control systems engineering. John Wiley & Sons, 2020.
2. Nicolescu, Gabriela, and Pieter J. Mosterman. Model-based design for embedded systems. Crc Press, 2018.
3. Torsten Gedenk, Hardware-in-the-Loop Simulation. dSPACE GmbH (White Papers & User Guides)
4. Valvano, Jonathan W. Embedded Systems. Eigenverl. des Verf., 2014.
5. Marwedel, Peter. Embedded system design: embedded systems foundations of cyber-physical systems, and the internet of things. Springer Nature, 2021.

9.2 Laboratory	Hours HEI	Hours CO	Teaching methods	Obs.
1. Introduction to SIL and HIL Environments; available hardware (e.g., Speedgoat, Arduino, dSPACE); overview of software: MATLAB/Simulink, Real-Time Workshop; setting up simulation environments		4	Short tutorials	
2. Code Generation for SIL; use Embedded Coder to auto-generate code from Simulink; run and debug code in a host-based simulation; analyze performance and accuracy		4		
3. Real-Time Execution Using Simulink Real-Time; compile model for real-time execution; run model on Speedgoat/dSPACE hardware; real-time parameter tuning and signal monitoring		4		

4. Hardware Setup and I/O Configuration; connect hardware (Raspberry Pi, Arduino, or dSPACE); configure analog/digital I/O, PWM, encoder inputs; deploy simple blinking LED or PWM control		4		
5. HIL Simulation – DC Motor Control; replace physical plant with Simulink model; interact with real controller through analog/digital signals; inject faults (sensor noise, delay)		4		
6. CAN Communication and Signal Monitoring; set up CAN communication between hardware and PC; use Vector CANoe or NI tools; read/write CAN messages from Simulink		4		
7. Case Study – Automotive ECU HIL Testing; HIL setup for ECU (virtual engine control or ADAS); Simulate sensors and actuators; Log performance data and faults		4		

Bibliography

1. Nise, Norman S. Control systems engineering. John Wiley & Sons, 2020.
2. Nicolescu, Gabriela, and Pieter J. Mosterman. Model-based design for embedded systems. Crc Press, 2018.
3. Torsten Gedenk, Hardware-in-the-Loop Simulation. dSPACE GmbH (White Papers & User Guides)
4. Valvano, Jonathan W. Embedded Systems. Eigenverl. des Verf., 2014.
5. Marwedel, Peter. Embedded system design: embedded systems foundations of cyber-physical systems, and the internet of things. Springer Nature, 2021.

10. Correlation of course content with the expectations of the epistemic community representatives, professional associations, and major employers in the field related to the program

Control system communities' expectations meet rigorous validation and verification processes for safety-critical systems (e.g., ISO 26262 in automotive, DO-178C in aerospace).

11. Evaluation

Activity Type	**Evaluation criteria**	**Evaluation methods**	**Weight in final grade**
11.1 Lecture	Quizzes & Assignments: Final Exam:	Written evaluation	20% 40%
11.2 Seminar/ Laboratory/Project	Individual Project	Project presentation	40%
11.3 Minimum Performance Standard			
Quizzes & Assignments *0.2 + Final Exam*0.4>5			

Date of completion: 15.09.2025	Lecturers		Signature
	Course	Conf. Dr. ing Mirela Dobra	

Date of approval by the Department of Automation Council 24.11.2025	Director of the Department of Automation Prof.dr.ing. Honoriu VĂLEAN
Date of approval by the Faculty of Automation and Computer Science Council 28.11.2025	Dean Prof.dr.ing. Vlad MUREȘAN