COURSE DESCRIPTION

Analysis and synthesis of circuits

Academic year 2026-2027

1. Programme-related data

1.1. Higher Education Institution	Babeş-Bolyai University
1.2. Faculty	Faculty of Mathematics and Computer Science
1.3. Department	Department of Computer Science
1.4. Field	Computer Science
1.5. Level of study	Bachelor
1.6. Degree programme / Qualification	Computer Science (in English)
1.7. Form of education	Full time

2. Course-related data

2.1. Course title	**Analysis and synthesis of circuits**			Course code	**MLE5185**
2.2. Course coordinator	Prof. dr. ing. Octavian Creţ				
2.3. Seminar coordinator	Prof. dr. ing. Octavian Creţ				
2.4. Year of study	3	2.5. Semester	6	2.6. Type of assessment	Viva voce
2.7. Course status	Optional			2.8. Course type	Specialisation subject

3. Total estimated time (hours per semester of teaching activities)

3.1. Number of hours per week	4	of which: 3.2. course	2	3.3. ~~seminar~~/ laboratory/ ~~project~~	1
3.4. Total of hours in the curriculum	42	of which: 3.5. course	28	3.6. ~~seminar~~/ laboratory/ ~~project~~	14
Time allocation for individual study (IS) and self-taught activities (ST)					**hours**
Learning from textbooks, course materials, bibliography, and notes (IS)					26
Additional research in the library, on subject-specific electronic platforms, and on-site					14
Preparing seminars/ laboratories/ projects, assignments, reports, portfolios, and essays					22
Tutoring (professional guidance)					18
Examinations					3
Other activities					
3.7. Total hours of individual study (IS) and self-taught activities (ST)				83	
3.8. Total hours per semester				125	
3.9. Number of credits				5	

4. Prerequisites (where applicable)

4.1. curriculum-related	
4.2 skills-related	

5. Specific conditions (where applicable)

5.1. course-related	Class room with projector
5.2. seminar/laboratory-related	Laboratory with computers

6.1. Competencies resulting from the completion of the degree programme ***(as referred to in the curriculum)***

Professional competencies	
Competency code	**Competency**
PC3	Use of software tools in an interdisciplinary context
PC4	Use of theoretical foundations of computer science as well as of formal models

Transversal competencies	
Competency code	**Competency**
TC1	Application of organized and efficient work rules, of responsible attitudes towards the didactic-scientific field, to bring creative value to own potential, with respect for professional ethics principles and norms
TC2	Efficient development of organized activities in an interdisciplinary group and the development of empathetic abilities for interpersonal communications, to relate to and cooperate with various groups

6.2. Learning outcomes relevant to the degree programme (as referred to in the curriculum)

Learning outcomes targeted by the subject		
Competency code	**Knowledge and comprehension**	**Specific academic skills**
PC1	• The graduate is able to identify complex problems and examine related issues to develop solving options and implement solutions. • The graduate has the ability to apply general rules to specific problems and produce relevant solutions.	• Note-Taking: Summarizing and recording key information from lectures or texts. • Critical Thinking: Analyzing, evaluating, and synthesizing information rather than just • Reading Comprehension: Active reading, scanning, skimming, and understanding complex texts. • Digital Literacy: Using databases, software, and online tools effectively for research. • Communication & Presentation: Public speaking, presenting arguments, and group work. • Collaboration: Working effectively in teams and providing constructive critique.
PC2	• The graduate is able to combine diverse information to formulate solutions and generate ideas for developing new products and applications. • The graduate has knowledge related to programming, mathematics, engineering and technology and has the skills to use them to create complex information technology systems.	• Time Management: Prioritizing tasks, meeting deadlines, and organization. • Academic Writing: Constructing clear, structured, and evidence-based arguments. • Study Skills: Revision techniques, test-taking strategies, and memory aids. • Problem-Solving: Applying logical reasoning to solve academic problems.

7. Subject-specific learning outcomes

Knowledge and comprehension
1. The main objective of this course is to provide students with the fundamentals of Logic Design, enabling them to analyze, design, and implement any digital system. To achieve this goal, students will learn to:
2. Analyze and synthesize combinational logic systems;
3. Analyze and synthesize synchronous and asynchronous sequential circuits;
4. Apply digital system design principles and descriptive techniques;
5. Use programmable devices, such as FPGAs and PLDs, to implement logic/digital systems;
6. Understand synchronization issues in logic/digital systems and study them through simulation and physical testing.
Specific academic skills
1. The graduate is able to diagnose and troubleshoot logic/digital circuits.
2. The graduate is able to test and evaluate the functional and non-functional characteristics of logic/digital circuits based on specific criteria.

8. Contents

8.1. Course	Teaching and learning methods	Remarks
1. Introduction. Number systems and codes, errors	Presentations, discussions	N/A
2. Number systems. Binary arithmetic		
3. Boolean algebra. Boolean functions. Logic gates. Digital systems and the representation of Boolean functions		
4. Methods for minimizing Boolean functions		
5. Analysis and design (synthesis) of combinational logic circuits (CLC). SSI and MSI CLC circuits		
6. Methods for designing digital systems using SSI, MSI, LSI, and VLSI circuits. Combinational hazard analysis		
7. Sequential logic circuits. Flip-Flops and Latches		
8. Applications of Flip-Flops : frequency dividers, counters		
9. Applications of Flip-Flops: data registers, converters, memories		
10. Methods for designing digital systems using Flip-Flops		
11. Methods for designing digital systems using memories, multiplexers, decoders, and counters		
12. Methods for designing synchronous sequential systems		
13. Methods for designing digital systems using programmable devices (I)		
14. Methods for designing digital systems using programmable devices (II)		
Bibliography 1. Contemporary Logic Design, Randy H. Katz, Benjamin Cunnings / Addison Wesley Publishing Co., 1993. 2. Digital Design Principles and Practices, John F. Wakerly, Prentice-Hall, 2000. 3. FPGA-based System Design, Wayne Wolf, PRENTICE HALL Professional Technical Reference Upper Saddle River, NJ 07458 www.phptr.com ISBN: 0-13-142461-0.		
8.2. Seminar/ laboratory	**Teaching and learning methods**	**Remarks**
1. Introduction to laboratory topics. Initial experiments	Hands-on exercises using educational test boards, FPGA boards, specialized software (simulators), blackboard presentations, additional explanations, and discussions	N/A
2. Basic logic circuits		
3. The schematic editor and Logisim Evolution simulator (I)		
4. The schematic editor and Digital simulator		
5. Combinational logic circuits (I)		
6. Combinational logic circuits (II) – MSI circuits		
7. Combinational logic circuits (III) – Complex circuits		
8. Synthesis of combinational logic circuits using programmable logic devices		
9. Flip-Flops		
10. Counters (I)		
11. Counters (II)		
12. Registers and shift registers		
13. Complex sequential logic circuits		
14. Laboratory test		
Bibliography 1. Analiza şi sinteza dispozitivelor numerice, Îndrumător de laborator, Ediţia a-3-a, L. Văcariu, O. Creţ, A. Neţin, Ed. U.T. Press, Cluj-Napoca, 2009.		

9. Evaluation

Type of activity	9.1 Evaluation criteria	9.2 Evaluation methods	9.3 Percentage in the final grade
9.4. Course	Problem-solving skills. Presence, (inter)activity	Written exam	70%
9.5. Seminar/ laboratory	Problem-solving skills.	In-person and/or written exam, or via the TEAMS platform, if necessary	30%
9.6 Minimum standard for passing			
• Requirements for taking the final written exam: grade on practical assignments ≥ 5 • Requirements for passing the exam: grade on practical assignments ≥ 5 AND grade on the written exam ≥ 5; • Formulating and solving typical logic design problems using the formal tools specific to the field.			

10. SDG labels (Sustainable Development Goals)

	○	Sustainable Development Generic Label

								
○	○	○	○	○	○	○	○	○
								No label applies
○	○	○	○	○	○	○	○	◉

Date of entry:
May 9, 2026

Signature of course coordinator

.....

Signature of seminar coordinator

.....

Date of approval in the department:
...

Signature of the head of department

.....