Universitatea Națională de Știință și Tehnologie Politehnica București

Facultatea de Electronică, Telecomunicații și

Tehnologia Informației

COURSE DESCRIPTION

1. Program identification information

1.1 Higher education institution	**National University of Science and Technology Politehnica Bucharest**
1.2 Faculty	**Electronics, Telecommunications and Information Technology**
1.3 Department	**Electronic Devices, Circuits and Architectures**
1.4 Domain of studies	Electronic Engineering, Telecommunications and Information Technology
1.5 Cycle of studies	Bachelor/Undergraduate
1.6 Programme of studies	Microelectronics, Optoelectronics and Nanotechnologies

2. Date despre disciplină

2.1 Course name (ro) (en)				Arhitectura sistemelor de calcul Computer Systems Architecture			
2.2 Course Lecturer				S.l./Lect. Zoltan Hascsi, PhD			
2.3 Instructor for practical activities				S.l./Lect. Zoltan Hascsi, PhD			
2.4 Year of studies	4	2.5 Semester	I	2.6. Evaluation type	V	2.7 Course regime	Ob
2.8 Course type		S	2.9 Course code	04.S.07.O.406		2.10 Tipul de notare	Nota

3. Total estimated time (hours per semester for academic activities)

3.1 Number of hours per week	3	Out of which: 3.2 course	2.00	3.3 seminary/laboratory	1
3.4 Total hours in the curricula	42.00	Out of which: 3.5 course	28	3.6 seminary/laboratory	14
Distribution of time:					hours
Study according to the manual, course support, bibliography and hand notes Supplemental documentation (library, electronic access resources, in the field, etc) Preparation for practical activities, homework, essays, portfolios, etc.					32
Tutoring					1
Examinations					0
Other activities (if any):					0

3.7 Total hours of individual study	33.00
3.8 Total hours per semester	75
3.9 Number of ECTS credit points	3

4. Prerequisites (if applicable) (where applicable)

4.1 Curriculum	- Digital Integrated Circuits - Microprocessor Architecture
4.2 Results of learning	- basic-level knowledge of digital circuits and microprocessors - digital design methodology (Verilog HDL language and an integrated development environment - IDE)

5. Necessary conditions for the optimal development of teaching activities (where applicable)

5.1 Course	- Room equipped with a video projector
5.2 Seminary/ Laboratory/Project	- Room with computers/wotkstations - Xilinx Vivado IDE - Compulsory presence at laboratory classes, according to current PUB regulations

6. General objective *(Reffering to the teachers' intentions for students and to what the students will be thought during the course. It offers an idea on the position of course in the scientific domain, as well as the role it has for the study programme. The course topics, the justification of including the course in the currcula of the study programme, etc. will be described in a general manner)*

A detailed review of the current state-of-the-art processor architectures and of the main paradigms used to increase computer system performance. Design of basic computer systems with a CISC/RISC processor using a hardware description language - verilog, and simulation of the execution of short program sequences, written at machine code level.

7. Competences *(Proven capacity to use knowledge, aptitudes and personal, social and/or methodological abilities in work or study situations and for personal and proffesional growth. They refflect the empolyers requirements.)*

Specific Competences	- Understanding advanced architectural concepts used in computing systems as well as their limits. - Evaluation of the performance of computer system/subsystem from several perspectives: speed, cost, reliability, scalability. - Designing a mixed system, with hardware (processor) and software (program) components.
Transversal (General) Competences	- Honorable, responsible and ethical behavior to ensure the reputation of the profession. - Awareness of the need for continuous training; efficient use of resources and learning techniques for personal and professional development.

8. Learning outcomes *(Synthetic descriptions for what a student will be capable of doing or showing at the completion of a course. The learning outcomes reflect the student's acomplishments and to a lesser extent the teachers' intentions. The learning outcomes inform the students of what is expected from them with respect to performance and to obtain the desired grades and ECTS points. They are defined in concise terms, using verbs similar to the examples below and indicate what will be required for evaluation. The learning outcomes will be formulated so that the correlation with the competences defined in section 7 is highlighted.)*

Knowledge	*The result of knowledge aquisition through learning. The knowledge represents the totality of facts, priciples, theories and practices for a given work or study field. They can be theoretical and/or factual.* • **Describe** various types of computer/processor/memory architectures/structures. • **Classifiy** the computer/processor/memory architectures/structures using specific taxonomies. • **List** the characteristics, advantages and disadvantages of various types of computer/processor/memory architectures/structures. • **Explain** the operation of computer/processor/memory architectures/structures using block diagrams and/or diagrams/flow charts/graphs. • **Define** performance metrics for computer/processor/memory architectures/structures and **exemplify** methods for measuring/evaluating them. • **List** and **describe** the instruction types and addressing modes. • **Define** and **exemplify** the structural, data, and control instruction dependencies. • **Classify** the techniques/structures for instruction dependecy management. • **Explain** various techniques/structures for instruction dependecy management using block diagrams and/or diagrams/flow charts/graphs.
Skills	*The capacity to apply the knowledge and use the know-how for completing tasks and solving problems. The skills are described as being cognitive (requiring the use of logical, intuitive and creative thinking) or practical (implying manual dexterity and the use of methods, materials, tools and intrumentation).* • **Estimate**, using appropriate metrics, the performance of a computing system. • **Identify** the component parts of a computer/processor/memory structure. • **Modify** a given architecture/structure to add a new instruction. • **Compare** computer, processor, memory architectures. • **Highlight** the advantages/disadvantages of an architecture/structure. • Functionally/structurally **describe** an architecture/structure using the verilog language. • **Simulate** an architecture/structure and **interpret** the simulation results. • **Analyze** a sequence of instructions/assembly code. • **Identifiy** structural, data, and control dependencies in a sequence of instructions. • **Write** a sequence of instructions in assembly code.

Responsability and autonomy	*The student's capacity to autonomously and responsably apply their knowledge and skills.* • Efficiently **use** the software tools and hardware resources for learning, analysis and design. • **Select** appropriate bibliographic sources and critically analyze them. • **Reuse** and **adapt** old structures/algorithms for new problems. • **Demonstrate** autonomy in planning and implementing solutions to given problems, as well as identifying and correcting errors/mistakes. • **Collaborate** with other colleagues and teaching staff in carrying out teaching activities. • Responsibly **apply** the principles, norms and values of professional ethics in completing homework and laboratory assignments. • **Self-evaluate** objectively, identifying gaps and needs, **provide** proactive feedback.

9. Teaching techniques *(Student centric techniques will be considered. The means for students to participate in defining their own study path, the identification of eventual fallbacks and the remedial measures that will be adopted in those cases will be described.)*

Teaching is based on the use of the videoprojector (covering communication and demonstration function). The oral communication methods are expository method and problem-solving method. Course materials are lecture notes and presentations. The lecture slides and course activities are available online on the faculty’s “Moodle” platform.

10. Contents

COURSE		
Chapter	**Content**	**No. hours**
1	Introduction. Sequential computer. Von Neumann model. Microprogramming. RISC versus CISC.	2
2	Parallelism. Main concepts. Granurality. Amdahl’s law.	2
3	Limits of parallelism. Data dependencies: RAW, WAR, WAW. Control dependencies. Structural dependencies.	2
4	Resolving Dependencies. Static and dynamic instruction reordering. Register renaming. Branch prediction and predication.	2
5	Pipelining. Introduction. Design. Performance and limitations. Data forwarding.	4
6	WLIW processors.	2
7	Superscalar processors. Instruction dispatch and issue. Execution order. Reorder buffer. Centralized (Thornton) and distributed (Tomasulo) algorithms. Exceptions and interrupts handling.	4
8	Multithreading and multicore processors. Multithreaded architectures. SMT. Multicore. Multithreading versus multicore.	2
9	Memory organization and management. Memory hierarchy. Cache memory. Types of cache. Virtual memory. Paging. TLB. Cache + TLB architectures	4

10	Branch prediction. Static prediction. Dynamic prediction. Adaptive predictors. Multiple prediction. Trace memory.	2
11	Embedded systems. Constraints and peculiarities.	2
	Total:	28

Bibliography:

1. https://curs.upb.ro/
2. John L. Hennessy, David A. Patterson. Computer Organization and Design
3. William Stallings. Computer Organization and Architecture

LABORATORY

Crt. no.	**Content**	**No. hours**
1	HW description and simulation of a RALU (ALU + register set + top + testbench)	2
2	HW description and simulation of a CISC sequential computer (processor with RALU and UCP + memory + top + testbench)	4
3	HW description and simulation of a RISC computer without dependencies (3 stages pipeline processor + program memory + data memory + top + testbench)	4
4	Data and control dependencies management in pipeline RISC (changes of the ALU, pipeline, processor and program memory)	2
5	Colloquium	2
	Total:	14

Bibliography:
https://wiki.dcae.pub.ro/index.php/Arhitectura_Sistemelor_de_Calcul

11. Evaluation

Activity type	11.1 Evaluation criteria	11.2 Evaluation methods	11.3 Percentage of final grade
11.4 Course	knowing how to apply the theory to specific problems	Mid-semester evaluation: written test.	30
	knowledge of the fundamental theoretical concepts	Final quiz: a multiple choice test with 20 questions uniformly covering the entire course.	20
11.5 Seminary/laboratory/project	Verilog implementation/adaptation of a given structure; Program design at machine code level; Circuit and program validation through simulation;	Laboratory Colloquium during the last meeting	50

11.6 Passing conditions
• At least 50% of the total marks. • At least 50% of the marks alloted to laboratory colloquim.

12. Corroborate the content of the course with the expectations of representatives of employers and representative professional associations in the field of the program, as well as with the current state of knowledge in the scientific field approached and practices in higher education institutions in the European Higher Education Area (EHEA)

- Digital Design is a key sector of electronics industry, with a rapid pace of growth. Recent entries in the Romanian industry of several renowned world producers of digital circuits and systems (Infineon, Microchip, IXIA) and the expansion in the digital design of some companies already operating in the telecommunications and applications sectors, substantially increased the demand for qualified engineers with a know-how in the analysis, design and operation of digital systems and programmable digital circuits.
- This course provides graduates the appropriate skills sought for by the employers from the above mentioned domains, with a modern scientific and technical know-how enabling their rapid employment after graduation, and is in line with the policy of "Politehnica" University of Bucharest, both in terms of content and structure, and in terms of skills and international openness for students.

Date	Course lecturer	Instructor(s) for practical activities
01.10.2025	S.l./Lect. Zoltan Hascsi, PhD	S.l./Lect. Zoltan Hascsi, PhD
		S.l./Lect. Mihai Antonescu, PhD
		S.l./Lect. George-Vlăduț Popescu, PhD
Date of department approval	Head of department	
	Prof. Claudius DAN, PhD	

Date of approval in the Faculty Council	Dean
	Prof. Radu Mihnea UDREA, PhD