COMPUTER ORGANIZATION AND ARCHITECTURE

Faculty of Computer Science				
Study programme:	**Computer Science**		Degree level:	**Engineer's degree full-time programme**
Specialization	**---**		Diploma path:	**2026/2027W - 2026/2027S**
Module name:	**Computer Organization and Architecture** (Architektura komputerów)			
Module type:	**obligatory**	**Semester: 1**	ECTS:**5**	Module ID:**FCS-00011**
No. of hrs in semester:	Lecture (L) - **30** Classes(C) - **0** Specialization workshop (SW) - **30** Project (P) - **0** Laboratory classes (LC) - **0** Seminar (S) - **0**			
Prerequisites	Synthesis of Digital Systems (FCS-00106),			

Aims and objectives:	Course delivers the basic understanding of computer organization: roles of processors, main memory, and input/output devices. Understanding the concept of programs as sequences of machine instructions. Understanding the relationship between assembly language and machine language; development of skill in assembly language programming. Understanding simple data path and control designs for processors. Understanding memory organization, including cache structures and virtual memory schemes. Course include basic machine architecture and design, central processing unit, pipelining, machine representation of instructions and data, addressing techniques, I/O and interrupts, memory organization, and execution of instructions at machine level. Several laboratory assignments will provide hands-on experience on some of the above topics.		
Forms of teching activities::	lecture, specialization workshop,	Assessment:	Evaluation must be relevant to the intended learning outcomes: The objectives of this course is to teach critical thinking and how to communicate technical concepts in the area of computer systems. These objectives will be met through lectures with final examination and challenging assignments at laboratory and the project classes. Methods for assessing: 1: Traditional writing exam. 2: Team-based assignments (in lab exercises) in which an effective student interaction and effective work-load sharing is required .
Module content:	Lectures History of Calculation and Computer Architecture. Harvard-Princeton Architecture. Complex Instruction Set: programming model and Stack, GPR Architectures. CISC, RISC and x86 family. Simple Instruction Pipelining. Pipeline Hazards Advanced Superscalar Architectures Multilevel Memories - Technology Cache (Memory) Performance Optimization, Cache Coherence and (MPP, UMA, NUMA, MP, SMP). Virtual Memory. I/O and interrupts. VLIW/EPIC: Intel IA-64 Hands-on exercises Principles of using watchdog functionality, programing of alphanumeric LCD display, programing of timers (timers), programing of uart serial communication, programing of ADC, programing of 1-wire serial communication		
Teaching methods:	project method, programming, lecture problem,		

Learning outcomes

Symbol	Specify min. 4, max. 8 learning outcomes in the following order: knowledge – skills – competence. Each learning outcome must be verifiable	Reference to the programme learning outcomes of education
LO1	should be able to define the concept of computer system architecture and organization, espesially in term of such components as CPU micro-architecture, memory, I/O.	
LO2	should be able to jude the performance of computer architectuer affected by a very wide range of design choices	
LO3	should be able to specify and design the simple organization and architecture of the computer system.	
LO4	should be able to implement the simple organization and architecture of the computer system with used of the microcontolres and its starting kits board	
LO5	should be able to write assemble assembly language programs, assemble into machine a cross assembler utility and download and run their program on the training boards that will provide solutions real-world control problems.	
LO6	should be able to use Technical Document	

No. of learning outcome	Methods of assessing the learning outcome	Type of teaching activities (if more than one) during which the outcome is assessed
LO1	Writing quizzes will be held in class with exact dates determined during the semester.	L
LO2	Writing quizzes will be held in class with exact dates determined during the semester.	L
LO3	Preparing, attending lab and demonstrating to your lab instructor that you have completed the work	Lab

LO4	Preparing, attending lab and demonstrating to your lab instructor that you have completed the work	Lab	
LO5	Preparing, attending lab and demonstrating to your lab instructor that you have completed the work. Project report, results and discussion	Lab	
LO6	Preparing, attending lab and demonstrating to your lab instructor that you have completed the work. Project report, results and discussion.	Lab	

Student's workload (in hours)	**1** - Attendance at lectures	None	30
	2 - Attendance at laboratories	None	30
	3 - Preparation for hands on activities	None	15
	4 - Performance of projects task	None	30
	5 - Participation in student-teacher sessions	None	8
	6 - Preparation for the assessment	None	10
	7 - Presence during assessment		2
		TOTAL:	**125**
Quantitative indicators	Student's workload - activities that require direct teacher participation: (5)+(2)+(1)+(7)	70	**ECTS**
			2.8
	Student's workload connected with practical classes (4)+(3)+(2)+(6)	85	3.4

Basic references:	1. William Stallings, Computer Organization and Architecture: Designing for Performance, Prentice Hall Eighth Edition 2. HENNESSY J.L., PATTERSON D.A., Computer Architecture. A Quantitative Approach, San Mateo CA, Morgan Kaufmann. 3. PATTERSON D.A., HENNESSY J.L., Computer Architecture. Hardware-Software Interface, San Mateo CA, Morgan Kaufmann, 1998.
Further reading	1. Computer Systems: A Programmer’s Perspective, Randal E. Bryant and David O’Hallaron, 2nd Ed, Prentice Hall, 2011

Unit:	Department of Digital Media and Computer Graphics	Lecturer/ instructor	
Date of issuing the programme:	31st March 2026	Author of the programme:	dr inż. Mirosław Omieljanowicz

L - lecture, C - classes, LC - laboratory classes, P-project, SW - specialization workshop, S - seminar