POZNAN UNIVERSITY OF TECHNOLOGY

COURSE DESCRIPTION CARD - SYLLABUS

Course name
Custom Computing [S2Inf1-PB>SUOBL]

Course

Field of study	Year/Semester
Computing	1/1
Area of study (specialization)	Profile of study
Edge Computing	general academic
Level of study	Course offered in
second-cycle	Polish
Form of study	Requirements
full-time	compulsory

Number of hours

Lecture	Laboratory classes	Other
20	30	0
Tutorials	Projects/seminars	
0	0	

Number of credit points

4,00

Coordinators

dr inż. Marek Kropidłowski
marek.kropidlowski@put.poznan.pl

Lecturers

Prerequisites

Knowledge: Student starting this module should have a basic knowledge in the field of digital electronics, structured programming and scripting languages. Skills: The student should be able to obtain information from the indicated sources, as well as understand the need to expand his competences and be ready to cooperate in a team. Social Competences: The student should show such features as: honesty, responsibility, perseverance, cognitive curiosity, creativity, personal culture, respect for other people.

Course objective

1. To provide students with knowledge on the construction and operation of high-performance FPGA devices used in modern digital systems and edge computing. 2. Present students a set of development technologies for modeling digital devices, designing reusable components, FPGA-based prototyping. 3. Developing students" skills in solving technical problems in the field of complex digital system design. 4. Shaping teamwork skills in students - the ability to cooperate in the design teams and in the preparation of final research reports.

Course-related learning outcomes

Knowledge:

1. has advanced and detailed knowledge related to selected areas of computer science, developing digital systems, prototyping system-on-chip for hardware verification,
2. has knowledge about new technologies in the area of hardware-software development and embedded systems
3. has advanced and detailed knowledge regarding hardware life cycle which involves developing a programmable hardware system and testing it.

Skills:
1. is able to acquire, combine, interpret and evaluate information from literature, databases and other information sources (in mother tongue and english); draw conclusions, and formulate opinions based on it,
2. is able to combine knowledge from different areas of computer science (and if necessary from other scientific disciplines) to formulate and solve engineering tasks related to hardware-software development,
3. is able to design and develop a hardware layer of complex digital system,
5. is able to design (according to a provided specification which includes also non-technical aspects) a digital system using technologies learned during the course,
6. is able to work in a group, performing a role of programmable hardware designer.

Social competences:
1. understands that knowledge and skills related to computer science quickly become obsolete,
2. knows how new development technologies and tools could be helpful to solve practical problems like developing a digital system.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:
Formative assessment:
a) lectures: based on the answers to the questions which test understanding of material presented on the lectures
b) laboratory classes: based on the assessment of the tasks done during classes and as a homework
Summative assessment:
a) verification of assumed learning objectives related to lectures within an online written test. The final grade is determined using the following scale: (90%, 100%] -> 5.0, (80%, 90%] -> 4.5, (70%, 80%] -> 4.0, (60%, 70%] -> 3.5, (50%, 60%] -> 3.0, (0%, 50%] -> 2.0.
b) verification of assumed learning objectives related to laboratories is based on:
- design contest and verification of the laboratory tasks. The final grade is determined using the following scale: (90%, 100%] -> 5.0, (80%, 90%] -> 4.5, (70%, 80%] -> 4.0, (60%, 70%] -> 3.5, (50%, 60%] -> 3.0, (0%, 50%]-> 2.0.
Getting extra points for activity during classes, especially for:
- proposing to discuss additional aspects of the issue,
- effectiveness of applying the acquired knowledge while solving a given problem,
- ability to work within a team that practically performs a specific task in a laboratory,
- comments related to the improvement of teaching materials.

Programme content

The program covers the design of reprogrammable systems using hardware description languages.

Course topics

The lecture program includes the following topics:
Synthesis and implementation of digital devices on FPGA programmable platforms. Dependence of the HDL description on the implementation platform (FPGA vs. ASIC). Construction of modern reconfigurable systems such as FPGA/CPLD/FPAA/FPOA/3D-PLD/PSoC. The role of IP components (Intellectual Property) in the design of complex digital systems. Using remote (IRL), partial and dynamic reconfiguration (PDR) in digital edge systems.
Systems with dynamic reconfiguration: configuration memory models, design methods. Softprocessors and reconfigurable platforms for embedded systems: overview of solutions, implementation methods and tools,

hybrid systems, AP SoC systems.

Laboratory classes are conducted in the form of 2-hour meetings held in the laboratory, preceded by an instructional session at the beginning of the semester. The exercises are carried out by two-person teams of students.
The laboratory program covers the following topics:
Studying the influence of the description style used on the synthesis results. Constructing advanced testing environments (testbench). Design and implementation
digital devices using software from Mentor Graphics, AMD-Xilinx, Altera-Intel.

Some of the above-mentioned program content is carried out as part of the student's own work.

Teaching methods

1. Lecture with multimedia presentation (diagrams, formulas, definitions, etc.) supplemented by the content of the board.
2. Laboratory exercises: multimedia presentation, presentation illustrated with examples given on the board and performance of tasks given by the teacher - practical exercises.

Bibliography

Basic
1. Mark Zwoliński, Projektowanie układów cyfrowych z wykorzystaniem języka VHDL, WKŁ2007, ISBN: 9788320616354.
2. Andrew Rushton, VHDL for Logic Synthesis, Third Edition, John Wiley & Sons, 2011, ISBN: 978-0-470-68847-2
3. Scott Hauck, Andre DeHon, Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation / Edition 1, Elsevier Science, November 2007, ISBN: 0123705223
Additional
1. Michael Keating, Pierre Bricaud, Reuse Methodology Manual for System-on-a-Chip Designs / Edition 3, Springer-Verlag New York, August 2007, ISBN: 0387740988
2. Peter J. Ashenden, The Designer"s Guide to VHDL / Edition 3, Elsevier Science, June 2008, ISBN:0120887851.
3. Steve Kilts, Advanced FPGA Design: Architecture, Implementation, and Optimization, John Wiley & Sons, June 2007, ISBN: 0470054379.

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,00
Classes requiring direct contact with the teacher	50	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	50	2,00