GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

Subject name and code	Hardware and Software Integration, PG_00048101		
Field of study	Electronics and Telecommunications		
Date of commencement of studies	October 2024	Academic year of realisation of subject	2027/2028
Education level	first-cycle studies	Subject group	Optional subject group Subject group related to scientific research in the field of study
Mode of study	Full-time studies	Mode of delivery	at the university
Year of study	4	Language of instruction	Polish
Semester of study	7	ECTS credits	2.0
Learning profile	general academic profile	Assessment form	assessment
Conducting unit	Department of Metrology and Optoelectronics -> Faculty of Electronics, Telecommunications and Informatics		
Name and surname of lecturer (lecturers)	Subject supervisor	dr hab. inż. Grzegorz Lentka	
	Teachers	dr hab. inż. Grzegorz Lentka	

Lesson types and methods of instruction

Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
Number of study hours	15.0	0.0	15.0	0.0	0.0	30

E-learning hours included: 0.0

Learning activity and number of study hours

Learning activity	Participation in didactic classes included in study plan	Participation in consultation hours	Self-study	SUM
Number of study hours	30	2.0	18.0	50

Subject objectives

Getting familiar with rules and methods tasks for hardware and software on a microsystem level, techniques of effective usage of resources of programmable components, methods and tools for co-design, co-debugging and co-testing of hardware and software.

Learning outcomes

Course outcome	Subject outcome	Method of verification
[K6_W32] Knows the parameters, functions and methods of analysis, design and optimization of analogue and digital circuits and electronic systems	Chooses methods and tools for co-design, codebugging and co-testing of hardware and software. Presents and uses techniques of effective usage of resources of programmable components. Explains methods of software development.	[SW1] Assessment of factual knowledge
[K6_U03] can design, according to required specifications, and make a simple device, facility, system or carry out a process, specific to the field of study, using suitable methods, techniques, tools and materials, following engineering standards and norms, applying technologies specific to the field of study and experience gained in the professional engineering environment	Student assigns tasks for hardware and software on a microsystem level. Estimates requirements for memory, computing power , power consumption. Student develops and debugs software on selected hardware platforms using software and hardware debugging tools.	[SU4] Assessment of ability to use methods and tools [SU1] Assessment of task fulfilment

Subject contents

1. Introduction: course outline, course grading, references. 2. Assigning tasks for hardware and software on a microsystem level. 3. Requirements definition and selection of the realization technology. 4. The rules of determination of critical requirements 5. Multi-level interfaces (like USB and CAN) as an example of a hardware software co-design. 6. Redundant design: design for testability. 7. CAD software for hardware-software co-design 8. The use of CPLD, FPGA and ISP technology for hardware reconfiguration by software means. 9. Effective usage of the resource of embedded controllers: effective ad-dressing modes, multi-instructions, bit-instructions. 10. Hardware-software optimization of power consumption of micropower systems. 11. Soft-processors: an example of hardware-software co-design. 12. Optimal assigning of tasks for pSoC. 13. Development of a software: low level and high level software libraries 14. Multitasking in microsystems 15. Hardware-software testing and debugging methods. 16. Debugging tools for hardware-software: software simulators, debug-gers, hardware emulators. 17. The construction and the use of Logic State Analyzers (LSA) 18. The use of ICD technique for debugging software on target hardware. 19. Summary

<table>
<tr><td>Prerequisites and co-requisites</td><td colspan="3">No requirements</td></tr>
<tr><td>Assessment methods and criteria</td><td colspan="3"><table>
<tr><th>Subject passing criteria</th><th>Passing threshold</th><th>Percentage of the final grade</th></tr>
<tr><td>Activity/homeworks</td><td>0.0%</td><td>10.0%</td></tr>
<tr><td>Lab exercises</td><td>0.0%</td><td>30.0%</td></tr>
<tr><td>Test during semester</td><td>50.0%</td><td>60.0%</td></tr>
</table></td></tr>
<tr><td rowspan="3">Recommended reading</td><td>Basic literature</td><td colspan="2">1. Ed Sutter: Embedded Systems Firmware Demystified, CMP 2002 2. J. J. Labrosse: Embedded Systems Building Blocks, Second Edition: Complete and Ready-to-Use Modules in C, CMP 1999 3. J. Staunstrup, W. Wolf: Hardware/Software Co-Design: Principles and Practice, Springer US 2010</td></tr>
<tr><td>Supplementary literature</td><td colspan="2">1. M. Barr, A. Massa: Programming Embedded Systems: With C and GNU Development Tools, 2nd Edition, O"Reilly Media 2006</td></tr>
<tr><td>eResources addresses</td><td colspan="2">Adresy na platformie eNauczanie:</td></tr>
<tr><td>Example issues/ example questions/ tasks being completed</td><td colspan="3"></td></tr>
<tr><td>Work placement</td><td colspan="3">Not applicable</td></tr>
</table>