



## Subject card

Subject name and code	Real-time Operating Microsystems, PG_00047486						
Field of study	Electronics and Telecommunications						
Date of commencement of studies	October 2022		Academic year of realisation of subject		2022/2023		
Education level	second-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	1		Language of instruction		Polish		
Semester of study	2		ECTS credits		2.0		
Learning profile	general academic profile		Assessment form		exam		
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				
			mgr inż. Dariusz Palmowski				
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		4.0		16.0	50
Subject objectives	Getting familiar with application, construction, scalability and portability of real-time operating microsystems.						
Learning outcomes	Course outcome		Subject outcome		Method of verification		
	[K7_U07] can apply advanced methods of process and function support, specific to the field of study		Analyses time constraints and selects system kernel type and configuration.		[SU4] Assessment of ability to use methods and tools		
	[K7_W03] Knows and understands, to an increased extent, the construction and operating principles of components and systems related to the field of study, including theories, methods and complex relationships between them and selected specific issues - appropriate for the curriculum.		Student defines terminology: operating system, realtime system, system kernel, multitasking, task, process, thread . Identifies operating microsystems specificity (small hardware resources, application area, task severity, reliability). Explains mutual exclusion and intertask communication techniques.		[SW1] Assessment of factual knowledge		
	[K7_U03] can design, according to required specifications, and make a complex 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		Uses non-preemptive operating system for co-operative multitasking. Realizes preemptive operating system with intertsk communication based on messages and kernel services.		[SU1] Assessment of task fulfilment		

Subject contents	1. Introduction: course outline, course grading, references 2. Basic terminology: operating system, realtime system, system kernel, multitasking, task, process, thread 3. Operating microsystems specificity (small hardware resources, applica-tion area, task severity, reliability). 4. Simultaneous vs. concurrent processing. Getting operating microsystem requirements. 5. System resources (memory, CPU time, interrupts, DMA, I/O ports). Efficient memory management techniques. 6. Problems and methods of resource reservation. Shared resource. Shared resource exclusive access. 7. Task management and scheduling. Scheduler. Examples of realization. 8. Methods of inter-task communication and synchronization. 9. Message usage and servicing: mailboxes and queues. 10. Time dependencies realization: task-level response, calling of task periodically, delaying, external events synchronization, timeouts. 11. Configurability and aided debugging. 12. System scalability and resource usage. 13. Portability of operating microsystems. 14. Source code documenting and portability. 15. Example of simple operating microsystem: RTXtiny, FreeRTOS, eCOS. 16. Example of advanced operating microsystem: uC/OS-II, QNX embed-ded, uClinux.		
Prerequisites and co-requisites	No requirements		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	Exam	50.0%	60.0%
	Activity/homeworks	0.0%	10.0%
	Lab exercises	0.0%	30.0%
Recommended reading	Basic literature	1. J. J. Labrosse: MicroC OS II: The Real Time Kernel, Newnes 2002 2. J. J. Labrosse: Embedded Systems Building Blocks, Second Edition: Complete and Ready-to-Use Modules in C, CMP 1999	
	Supplementary literature	1. Ed Sutter: Embedded Systems Firmware Demystified, CMP 2002	
	eResources addresses	Adresy na platformie eNauczanie:	
Example issues/ example questions/ tasks being completed	Starting and testing of an example of application based on FreeRTOS  Scalling of real-time operating microsystem to application requirements		
Work placement	Not applicable		