SYLLABUS OF A MODULE

Polish name of a module	**Programowanie systemów wbudowanych**
English name of a module	**Embedded systems programming**
ISCED classification - Code	0613
ISCED classification - Field of study	*Software and applications development and analysis*
Languages of instruction	*English*
Level of qualification:	*1 – BSc (EQF 6)*
Number of ECTS credit points	*6*
Examination:	*EW – exam written*

Number of hours per semester:

Lecture	Tutorial	Laboratory	Seminar	Project	Others
30EW	0	30	0	0	0

MODULE DESCRIPTION

MODULE OBJECTIVES

O1. Understanding advanced programming methods and properties of embedded systems.

O2. Acquiring practical skills in using selected integrated development environments as well as designing and debugging embedded systems software.

PRELIMINARY REQUIREMENTS FOR KNOWLEDGE, SKILLS AND OTHER COMPETENCES

1. Knowledge of mathematics, digital techniques and basic programming.
2. Ability to use different sources of information and technical documentation.
3. Ability to work independently and in a group.
4. Ability to correctly interpret and present their own activities.

LEARNING OUTCOMES

LO 1 – The student has theoretical knowledge in the field of architecture and basic properties of embedded systems and methods of their programming.

LO 2 – The student has the ability to use selected integrated programming environments and to design and implement embedded systems software.

MODULE CONTENT

Type of classes – Lectures		**Number of hours**
Lec 1 -	Basic definitions, architecture and components of microprocessor system: CPU, memory, GPIO, ADC. Real-time systems. Interrupts and software design models for embedded systems. Coding rules and conventions. CMSIS.	2
Lec 2 -	Computer arithmetic. Basic binary and logical operations in ANSI C. Programming of basic operations in C language.	2
Lec 3 -	Selected issues in the field of programming embedded systems in the C language: memory organization, pointers, data structures, bit fields and unions, modules, attribute modifiers, selected preprocessor and linker directives.	2
Lec 4 -	GPIO port controllers. Basic properties and programming methods.	2
Lec 5 -	Analog-to-digital and digital-to-analog converters.	2
Lec 6 -	Pulse width modulation technique and hardware timers. Analysis of sample programs.	2
Lec 7 -	Real-time operating systems.	2
Lec 8-9 -	Serial communication interfaces: UART, I2C, SPI, 1-wire. Basic properties, application area and software design methods.	4
Lec 10-11 -	Controller-area network (CAN) and real-time Ethernet communications solutions.	4
Lec 12 -	Fixed and floating-point numbers.	2
Lec 13 -	Embedded systems and programmable logic.	2
Lec 14-15-	Doxygen and MISRA standards. Summary of material.	4
Type of classes– Laboratories		**Number of hours**
Lab 1 -	Integrated development environment for microcontrollers. Running, analyzing and modifying of example programs.	**2**
Lab 2 -	Configuration of basic peripheral devices and control them with the CMSIS interface.	**2**
Lab 3 -	Basic arithmetic, binary and logical operations in C language. Measurement of the time needed for the microcontroller to perform selected arithmetic operations.	**2**
Lab 4-5 -	Interrupts, timers and PWM signal generation.	**4**
Lab 6-7 -	Design of multithreaded application based on RTOS.	**4**
Lab 8-12 -	Project implementation.	**10**
Lab 13-15 -	Project presentation. Discussion of implementation options.	**6**

TEACHING TOOLS

1. – Lectures using multimedia presentations, blackboard and chalk or whiteboards and pens.
2. – Computers with software, laboratory equipment and laboratory guides.
3. – Sample programs demonstrating the correct programming methods.

WAYS OF ASSESSMENT (F – FORMATIVE, S – SUMMATIVE)

F1. – Assessment of activity during classes.
S1. – Assessment of the ability to solve the problems posed and the manner of presentation obtained results - the final project for the evaluation of the laboratory. *.
S2. – Assessment of mastery of the teaching material being the subject of the lecture - exam.

*) The condition for obtaining credit is to receive positive grades from all laboratory exercises and to carry out the verification task.

STUDENT'S WORKLOAD

L.p.	Forms of activity	Average number of hours required for realization of activity
1. Contact hours with teacher		
1.1	Lectures	30
1.2	Tutorials	0
1.3	Laboratory	30
1.4	Seminar	0
1.5	Project	0
1.6	Consulting teacher during their duty hours	5
1.7	Examination	3
	Total number of contact hours with teacher:	68
2. Student's individual work		
2.1	Preparation for tutorials and tests	0
2.2	Preparation for laboratory exercises, writing reports on laboratories	22
2.3	Preparation of project	0
2.4	Preparation for final lecture assessment	0
2.5	Preparation for examination	20
2.6	Individual study of literature	40
	Total number of hours of student's individual work:	82
	Overall student's workload:	150
Overall number of ECTS credits for the module		6
Number of ECTS points that student receives in classes requiring teacher's supervision:		2,72
Number of **ECTS** credits acquired during practical classes including laboratory exercises and projects :		2,08

BASIC AND SUPPLEMENTARY RESOURCE MATERIALS

	BASIC
1.	Colin Walls: „Embedded Software: The Works", Elsevier, Boston, 2006.
2.	Geoffrey Brown, Discovering the STM32 Microcontroller, 2016 (free e-book).
3.	Donald Norris, Programming with STM32. Getting Started with Nucleo Board and C/C++, Mc Graw Hill Education, 2018.
4.	Trevor Martin, The Designer's Guide to the Cortex-M Processor Family. A Tutorial Approach, Elsevier, 2013.
5.	Herbert Schildt, C: The Complete Reference, Fourth Edition, McGraw-Hill, 2000.
	SUPPLEMENTARY
6.	Wayne Wolf: "Computers as Components: Principles of Embedded Computing System Design" Morgan & Kaufman 2000.
7.	Marwedel P.: "Embedded System Design" Kluwer Academic Publishers, Boston 2003.

MODULE COORDINATOR (NAME, SURNAME, INSTITUTE, E-MAIL ADDRESS)

dr hab. eng. Andrzej Przybył, Associate Professor, Institute of Computational Intelligence, andrzej.przybyl@pcz.pl