Subject programme

1. Subject name / subject module: **Embedded Systems Design**
2. Lecture language: **English**
3. The location of the subject in study plans:
   - Area or areas of the studies: **Computer Control Systems Engineering**
   - Degree of the studies: **2nd degree studies**
   - Field or fields (implementation of effects standard): Mechatronics
4. Supervision of subject implementation:
   - The Institute / Another unit: **Institute of Informatics and Mechatronics**
   - The person responsible for the subject: **Ocetkiewicz Tomasz, mgr inż.**
   - People cooperating in the development of the programme of the subject:
5. The number of hours and forms of teaching for individual study system and the evaluation method:

Mode of study	Teaching activities with the tutor – Form of classes												Total
	Lecture	SOW	ECTS	Laboratory work	SOW	ECTS	...	SOW	ECTS	...	SOW	ECTS	ECTS
Full-time studies	24	51	3	45	55	4							**7**
Part-time studies													
Credit rigor	Graded assigment			Graded assigment									

6. Student workload – ECTS credits balance:
   1 ECTS credit corresponds to 25-30 hours of student work needed to achieve the expected learning outcomes including the student's own work

Activity (please specify relevant work for the subject)	**Hourly student workload (full-time studies/part-time studies)**
Participation in lectures	24
Participation in laboratory classes	45
Preparation to laboratory classes	55
Independent study of the subject	49
Participation in an exam / graded assignment	2
Total student workload (TSW)	175
ECTS credits	7
* Student's workload related to practical forms	130
Student's workload in classes requiring direct participation of academic teachers	69

7. Implementation notes: recommended duration (semesters), recommended admission requirements, relations between the forms of classes:
   - Recommended admission requirements – none.
   - Recommended duration of the subject is taken from the course plan.
8. Specific learning outcomes – knowledge, skills and social competence:

Specific learning outcomes for the subject		Form	Teaching method	Methods for testing of (checking, assessing) learning outcomes
Outcome symbol	Outcome description			
Knowledge				
K_W02	**A student possesses sufficient knowledge of automation, electronics, and electrical engineering, necessary to understand the structure of the embedded system and to apply this knowledge in practice through the use of appropriate methods, tools, and components to design and prototype an embedded system and program a microcontroller.**	**Lecutre, Laboratory work**	**Inquiry methods**	**Student learning activities**
K_W05	**A student knows and understands selected facts and phenomena in embedded systems, is able to explain the complex relationships between them, which constitute the advanced general knowledge in the field of automation, electronics, and electrical engineering, sufficient to design, prototype, implement an embedded system.**			
K_W08	**A student has an in-depth knowledge of the life cycle of an embedded system.**			
Skills				

K_U01	**Student is able to obtain information through the selection of sources like on-line databases, application notes, product datasheets, integrate them, make their interpretation, critical analysis, and synthesis, as well as draw conclusions and formulate opinions to creatively solve complex and unusual problems.**	**Laboratory work**	**Inquiry methods**	**Student learning activities**

9. Assessment rules / criteria for each form of education and individual grades:

Activity	Grades	Calculation	To final
Laboratory tasks	db; bdb; bdb; db; bdb (4; 5; 5; 4; 5)	4 * 18% + 5 * 18% + 5 * 18% + 4 * 18% + 5 * 18% = 4.14	4.14
Classes attendence	> 75%	(80% * 5) * 10%	0.4
Final result			4.64

0 – 3.00	ndst	**4.01 – 4.50**	db
3.01 – 3.50	dst	**4.51 – 4.7**	db+
3.51 – 4.00	dst+	**4.71 – 5.0**	bdb

10. The learning contents with the form of the class activities on which they are carried out

a. (lecture) Design: Introduction to Embedded System embedded system overview, classification of embedded systems, hardware and software in an embedded system.
b. (lecture) Hardware design issue: core of the embedded system, memory, sensors, acutators, power-supply(battery,solar,energy harvesting), PCB design for embedded system(EMC).
c. (lecture, laboratory work) Memory: memory write ability and storage permanence, types of memory, memory hierarchy and cache.
d. (lecture, laboratory work) Interfacing: I/O addressing, interrupts, DMA, arbitration, multilevel bus architecture, communication protocols: SPI, I2C, I2S, CAN, UART.
e. (lecture, laboratory work) Embedded software: low-level programming, optimizing for speed/memory, interrupt service routines, data types, functions, multithreading programming.
f. (lecture, laboratory work) Real Time Operating System (RTOS): operating system basics, task, process and thread, multiprocessing and multitasking, task scheduling, task synchronization.
g. (lecture, laboratory work) Design examples: closed loop control system, PID controller implementation, user interface implementation(LED display, LCD, TFT, analog gauge,keyboards), interfacing to sensors and actuators.

11. Required teaching aids

a. Lecture - multimedia projector.
b. Laboratory classes - specialist laboratory.

12. Literature:

a. Basic literature:

1. Tianhong PanYi Zhu; Designing Embedded Systems with Arduino; ISBN 978-981-10-4418-2; Springer 2018

2. Peter Marwedel; Embedded System Design; ISBN 978-3-319-56045-8; Springer 2018

b. Supplementary literature:

1. Cesare Alippi; Intelligence for Embedded Systems; ISBN 978-3-319-05278-6; Springer 2014

2. Alberto Sangiovanni-Vincentelli, Haibo Zeng, Marco Di Natale, Peter Marwedel; Embedded Systems Development; ISBN 978-1-4614-3879-3; Springer 2014

3. Alan Holt, Chi-Yu Huang; Embedded Operating Systems; ISBN 978-3-319-72977-0; Springer 2018

4. MicroPython for the Internet of Things; Charles Bell; ISBN 978-1-4842-3123-4; Apress, Berkeley; 2017

c. Internet sources:

1. Getting started with the Arduino NANO 33 BLE Sense - https://www.arduino.cc/en/Guide/NANO33BLESense
2. Arduino Tutorials - https://www.arduino.cc/en/Tutorial/HomePage
3. Mbed OS - os.mbed.com
4. Introducing the Arm architecture - https://developer.arm.com/architectures/learn-the-architecture/introducing-the-arm-architecture/single-page
5. GCC-AVR Inline Assembler Cookbook Version 1.6 - http://www.ethernut.de/pdf/GCCAVRInlAsmCB.pdf
6. CooCox CoIDE - https://www.st.com/en/development-tools/coide.html

13. Available educational materials divided into forms of class activities (Author's compilation of didactic materials, e-learning materials, etc.)

14. Teachers implementing particular forms of education:

Form of education	Name and surname
1. Lecture	Ocetkiewicz Tomasz, mgr inż.
2. Laboratory classes	Ocetkiewicz Tomasz, mgr inż.
3. Training	
4. Project classes	
5. Workshop classes	
6. Simulation game	
7. Language classes	