(pieczęć wydziału)

COURSE DESCRIPTION

1. Course title: MICROPROCESSOR END EMBEDDED SYSTEMS	**2. Course code:** MES
3. Validity of course description: 2018/2019	
4. Level of studies: first degree	
5. Model of studies: stationary	
6. Field of study: INFORMATICS	
7. Profile of studies: general academic	
8. Programme: ALL	
9. Semester: 4, 5	
10. Faculty teaching the course: Faculty of Automatic Control, Electronics and Computer Science, Institute of Informatics	
11. Course instructor: DSc. Eng. Bartłomiej Zieliński	
12. Course classification: general	
13. Course status: obligatory	
14. Language: English	
15. Pre-requisite qualifications: Electronics and Measurements, Digital Circuits Theory, Arithmetic of Digital Systems, Digital Systems Design, Fundamentals of Computer Programming	

16. Course objectives:

The subject concerns hardware issues related to the construction and operation of functional modules of microprocessor and embedded systems. Its purpose is to present all the hardware and software aspects that will ensure understanding not only logical structures and operating principles of functional modules of microprocessors and embedded systems but also the methods of their design, construction and launching.

17. Description of learning outcomes:[1]

Nr	Learning outcomes description	Method of assessment	Teaching methods	Reference code
1	Student has an ordered, theoretically founded general knowledge in the field of microprocessor and embedded systems.	Exam	Lecture, classes	K1A_W07, K1A_W10

[1] należy wskazać ok. 5 – 8 efektów kształcenia

2	Student has basic knowledge about the life cycle of microprocessor systems. Student knows the basic methods, techniques and tools (hardware and software) used in the design, implementation and testing of microprocessor and embedded systems for typical applications.	Test, exam, laboratory report, project report	Lecture, classes, laboratory, project	K1A_W16, K1A_W22
3	Student has the ability to program using low-level languages and to use operating systems at the API level. Student can systematically test both hardware and software. Student has the ability to design hardware and software for simple microprocessor and embedded systems.	Test, exam, laboratory report, project report	Classes, laboratory, project	K1A_U08, K1A_U15, K1A_U23, K1A_U28
4	Student is able to determine the technical and functional specification of simple microprocessor systems and built-in with respect to software hardware and functional features. Student can, according to the given specification, design, implement and launch a simple microprocessor and embedded system including a hardware and software part using appropriate methods, techniques and tools.	Test, exam, laboratory report, project report	Classes, laboratory, project	K1A_U20, K1A_U28, K1A_U29
5	Student is able to plan and organize individual work, interact and work in a group taking on different roles.	Laboratory report, project report	laboratory, project	K1A_U31 K1A_K02

18. Teaching modes and hours

Lecture / BA /MA Seminar / Class / Project / Laboratory:

30 / 0 / 0 / 30 / 0 / 0 (4. Sem.)

30 / 0 / 0 / 0 / 16 / 30 (5. Sem.)

19 Syllabus description:

<u>Lecture</u>
Basic concepts. Directions of development and classification of microprocessors: microprocessors with a fixed instruction set of CISC and RISC type, microprogram microprocessors. Structure and organization of CISC microprocessors: functional elements of the microprocessor, the arithmetic and logic unit, shifting systems, general purpose registers, dedicated registers, control system, internal and external buses. Structure and organization of RISC microprocessors.
Microprocessor system (microcomputer): output and input signals (external bus) of the microprocessor, memory blocks, input and output circuits, unified and separated address space of I/O, address decoders. Timing of signals during the write and read cycle. Adaptation of processor speed to response time of the memory and peripherals. Microprocessor communication with the environment, transmission with and without checking the readiness status,

transmissions with interrupts, direct memory access (memory – I/O and memory – memory). Interrupt system. Interrupts: external and internal, maskable and non-maskable, single and multi-level, simple and vector. Programmable, universal and specialized I/O systems. Parallel and serial transmission. Examples of the construction and organization of typical programmable I/O systems: parallel and serial ports, programmable timers-counters, programmable interrupt controllers and DMA controllers.
One-chip microcomputers: logical structure, organization and operation of single-chip microcomputers with CISC and RISC processor cores. Examples of solutions and areas of application of selected one-chip microcomputers. Techniques of programming internal memories of single-chip microcomputers.
Dedicated microprocessors and coprocessors: I/O microcontrollers, signal microprocessors, numerical coprocessors (floating point processing units).
Embedded systems. Definition, application features. Methods of implementing the embedded system: microprocessor system, programmable system, hybrid system. Embedded system software, work without an operating system. Real-time operating system, system structure, task scheduling algorithms. Hardware dependent layer, device drivers. Methods of designing the embedded system.
Directions of microprocessor development and methods of increasing their efficiency on the example of selected modern 16, 32 and 64 bit CISC and RISC microprocessors. Parallelization and optimization of execution of individual phases of instructions: pipelining (single and multi-core microprocessors) increasing the number of fixed and floating point processing units, out of order execution, increasing the number of general purpose registers. Increasing the operating memory address space: real and virtual memory, paging. Memory management. Reducing the average access time to the operating memory: Cache memory, writing and reading methods, examples of Cache memory construction. Serialized transmissions to and from Cache memory.
Protection mechanisms used in microprocessors designed for multitasking. Task switching.
Built-in arbitrage mechanisms for creating multiprocessor systems.
Examples of solutions and comparison of selected modern microprocessors. System buses ISA, EISA, PCI, PCI Express. Scalable synchronous links QPI, HT. Architecture of PC-type microcomputers.
Design, construction and launching of microprocessor and embedded systems. Development tools: simulators, system emulators, logic state analysers. Areas of application of microprocessors and microprocessor systems.

Classes

Auditory (table) classes consist in solving tasks illustrating the synthesis of various modules (memory and I/O systems) of microprocessor and embedded systems and examples of their applications exercises based on lecture topics.

Laboratory

The aim of the laboratory is to familiarize students with the basic techniques of design, construction, software and launching of microprocessor and embedded systems. Laboratory is performed basing on different microcomputer platforms: AVR, ARM, DSP.

Project

The aim of the project is to develop of a microprocessor system prototype according to the proper phases: (assumptions, tasks, division of tasks into hardware and software, system concept, logical design, construction design, software, integration of hardware and software, system execution, launching and testing, creating the documentation). Each student (or two students in the case of more complex systems) is required to design and present individually a working device along with its documentation.

20. Exam: yes (sem. 5)

21. Primary sources:

1. P. Misiurewicz „Podstawy techniki mikroprocesorowej"; WNT 1991 r.;
2. H. Małysiak, B. Pochopień, P. Podsiadło, E. Wróbel „Modułowe systemy mikrokomputerowe" WNT 1990 r.
3. Berry B. Brey „The Intel Microprocessors 8086/8088 ... Pentium4 Architekture, Programming and Interfacing";Prentice Hall 2006r.
4. R. Baranowski „Mikrokontrolery AVR ATmega w praktyce"; BTC 2005r..

22. Secondary sources:

1. T. Shanley "Protected Mode Software Architecture"; Mind Share, Inc, 1996 r.
2. H. Małysiak "Mikrokomputery jednoukładowe rodziny MCS 48, 51, 96"; Wyd. PKJS, 1992r.
3. D. Patterson, J. Hennesy „Computer Organization and Design"; Elsevier 2005r.
4. A. Pawlaczyk Sztuka programowania mikrokontrolerów AVR – przykłady"; BTC 2007r.

23. Total workload required to achieve learning outcomes

Lp.	Teaching mode	Contact hours / Student workload hours
1	Lecture	60/40
2	Classes	30/20
3	Laboratory	30/20
4	Project	15/45
5	Seminar	/
6	Other	10/15
	Total number of hours	145/140

24. Total hours: 285

25. Numbers of ECTS: 8

26. Number of ECTS credits allocated for contact hours: 5

27. Number of ECTS credits allocated for in-practice hours (laboratory classes, projects): 4

26. Comments:

Approved:

……………………………..... ……………………………………………………………
(date, Instructor's signature) (date, the Director of the Faculty Unit signature)