COURSE DESCRIPTION CARD - SYLLABUS

Course name
Hardware applications of microcontrollers and microcomputers [S1MiKC2>ASMiM]

Course

Field of study
Microelectronics and Digital Communication

Year/Semester
3/5

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
Polish

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
24

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

3,00

Coordinators

dr inż. Sławomir Michalak
slawomir.michalak@put.poznan.pl

dr inż. Krzysztof Arnold
krzysztof.arnold@put.poznan.pl

Lecturers

Prerequisites

A student has structured and mathematically based knowledge of the basics of circuit theory, necessary to understand the operation of electrical circuits. He has mastered basic knowledge about analog and digital circuits. Has knowledge of the basics of measurement techniques. Is able to obtain information from literature in Polish and English. It actually uses the application of basic electronic components and circuits. Is able to measure typical parameters of electrical signals. He understands the need to expand his own knowledge and is responsible. Behaves actively during classes and systematically solves problems in the team.

Course objective

Presentation of the basic applications of microcontrollers and microcomputers. Knowledge and understanding the mechanisms of cooperation of the microcontroller and microcomputer with the environment. Understanding the principles of microcontroller operation in the modes of viewing the status of I/O modules and external devices and handling interrupts. Mastering programming skills to operate basic systems cooperating with microcontrollers. Mastering the ability to create simple microprocessor systems and systems managed by a microcomputer.

Course-related learning outcomes

Knowledge:
Has structured and practically based knowledge of the architecture of technically advanced microcontrollers and microcomputers. Understands the effective mechanisms of operation of built-in resources. Knows the possibilities and principles of cooperation of microcontrollers and microcomputers with external systems. Has basic knowledge of the division of tasks hardware/software, programming microcontrollers in assembler language and their hardware applications. Knows about the progress of technology aimed at the development of hardware towards resource integration, increasing the speed of operation and reducing power consumption.

Skills:
Is able to use company descriptions of modern microcontrollers. Is able to use a programming environment and write simple programs in assembly language. He can analyze and compare typical systems built using microcontrollers and microcomputers.

Social competences:
Is able to implement team projects. Has a sense of responsibility for running microcontroller and microcomputer applications.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

The final written and/or oral test verifies knowledge and understanding of the lecture content. Contains open problem questions with different scores. Final grade from the colloquium: less than 50% of the possible points - 2.0; from 50% - 3.0; from 60% - 3.5; from 70% - 4.0; from 80% - 4.5; from 90% - 5.0. The final grade for the laboratory is the weighted arithmetic mean of the grades for the implementation of basic and additional tasks (preparation for subsequent tasks, behavior, commitment, consolidation of skills) and the grades for individual or team reports closing the tasks. Weight is determined during introductory classes. The mark for an individual or team report may also be verified in oral and/or written form. Additional tasks verify your skills when applying to pass a laboratory or increase your grade. They may include a written or oral test. Scale for final grades: as for passing a lecture. The laboratory resit examination includes a practical part and a written or oral test.

Programme content

Lecture: Architecture of standard microcontrollers and microcomputers. Principle of CPU operation. Principle of operation and application of built-in hardware modules. Development environment, programmers and target systems. ISP programming. Programming microcontrollers in assembly language. Stack organization and calling subroutines. The interrupt system and its applications. Principle of operation, initialization and operating modes of parallel ports. Applications and programming support for typical data input systems to the microcontroller. Basic applications of microcontrollers and microcomputers with support for output devices. Port load capacity. Applications with viewing the status of parallel ports. Cooperation of microcontrollers and microcomputers with a character terminal. Communication with a PC.
Laboratory: Introduction to the programming environment and basics of assembly language programming. Standard microcontroller instructions and their effective use. Working registers, functional registers and memories. Internal data transfer. Conditional and unconditional changes in the order of execution of commands. Running arithmetic and logic programs. Introducing software delays. Stack-based applications. Running the main program and subprograms using simulation tools. Programming the microcontroller in the target system. Working with the evaluation set. Running the parallel ports of the microprocessor and microcomputer. Support for binary inputs in viewing mode.

Button and keyboard support. Use of output mode ports to control external devices. Visualization of the status of parallel ports using LEDs and displays. Running built-in modules with the support of measuring equipment. Starting communication between the slave station and the PC.

Course topics

Architecture and properties of microcontrollers and microcomputers. Programming environment and microcontroller operation in the target system. Assembly language in applications. Tools supporting the launch of hardware and software. Data processing applications. Using the device status review mode and the interrupt system. Operating modes and basic applications of parallel ports. Operating modes and basic applications of serial ports. Hardware expansion of the system.

Teaching methods

Lecture with multimedia presentation, supported by problem-solving discussion and examples on the board, containing student-activating references to experimental results of laboratory tasks observed or predicted by them.
Laboratory: implementation of practical problem tasks given by the instructor and verification of results using a programming environment and runtime kits, comparative discussion of final solutions, possible implementation of tasks requiring cooperation of two or more teams. Interactive varied tasks taking into account student progress.

Bibliography

Basic:
1. Andrzej Pawluczuk: Sztuka programowania mikrokontrolerów AVR. Podstawy. Wyd. BTC, Warszawa 2006
2. Andrzej Pawluczuk: Sztuka programowania mikrokontrolerów AVR. Przykłady. Wyd. BTC, Warszawa 2007
3. Rafał Baranowski: Mikrokontrolery AVR ATmega w praktyce. Wyd. BTC, Warszawa 2005
4. ATmega16A. 8-bit AVR Microcontroller with 16K Bytes In-System Programmable Flash. Datasheet, Atmel Corporation 2014

Additional:
1. Paweł Hadam: Projektowanie systemów mikroprocesorowych. Wyd. BTC, Warszawa 2004
2. ATmega128A. 8-bit AVR Microcontroller Datasheet Complete. Atmel Corporation 2015
3. ATmega8A, mega AVR Data Sheet. 2020 Microchip Technology Inc.
4. Jacek Bogusz: Lokalne interfejsy szeregowe w systemach cyfrowych. Wyd. BTC, Warszawa 2004

Breakdown of average student's workload

	Hours	ECTS
Total workload	84	3,00
Classes requiring direct contact with the teacher	54	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	30	1,00