SYLLABUS

Name: ***Microprocessors (EiTAu>SI5-Microp-17)***
Name in Polish:
Name in English: ***Microprocessors***

Information on course:

Course offered by department: Faculty of Automatic Control, Electronics and Computer Science
Course for department: Silesian University of Technology

Default type of course examination report:

EGZ

Language:

English

Course homepage:

https://platforma.polsl.pl/rau3/course/view.php?id=30

Short description:

Microprocessor architecture and its operation, instruction execution, bus architecture, interrupt system,
Microprocessor system architecture, interaction with memory and peripheral components, interrupt system concept and implementation, programming with use of assembly language and high level languages.
Basics of serial interfaces, serial data transfer concepts and implementation,
Practical aspects of design and implementation of embedded systems, programming and debugging, on chip debug systems, design of custom peripherals, system integration, simulation and modelling of systems

Description:

Lecture (30 hours):

Introduction to microprocessors
- Evolution from finite state machine through the microprogrammable device to microprocessor
- Basic functional blocks of microprocessor and their properties
- Architecture influence to behavior and performance of computing system
- Instruction execution cycle. Harvard and Princeton (von Neumann) architectures.
- Sequential and pipelined execution concepts

Computer system - general properties and requirements

Arithmeticd of computer systems
- Numeral systems
- Signed numbers - efficient representation
- Fixed point numbers
- Floating point numbers
- Arithmetic operations implementation

Multiprocessor systems
- Architecture classification
- Performance of multiprocessor systems and application areas
- Data flow graphs - representation of calculations
- Basics of scheduling of operations and tasks

Instruction set and argument addressing
- Memory model and addressing modes
- Memory management, protection and virtualization
- Implemented operations
- Conditional program flow implementation
- Subprograms

Bus architecture of computer system
- Bus interface cycles
- Controlling interface cycle timing
- Connecting memories and peripheral devices to microprocessor
- Taking control over a system bus - direct memory access

Interrupt system
- Event retrieval - pooling vs. interrupts
- Implementation concept
- Finding the interrupt source. Concept of vectorized interrupt system.
- Classification of interrupt systems

Multiprogram operation
- Idea and benefits
- Methods of task switching - cooperative and preemptive systems
- System resources management
- Protection - operation modes of CPU

Advanced processors
- RISC and CISC architectures
- Pipelined architecture - properties, benefits and limitations

- Memory protection system

Serial communication and selected interfaces
- Principles of asynchronous serial transmission
- Principles of synchronous data transmission
- Principles of data transmission and reception
- Synchronization of receiver in asynchronous and synchronous modes - Data integrity check methods
- Interfaces and protocols: RS232, 1-Wire, I2C, USB

Classes (15 hours):
- Introduction to programming of microprocessor with use of assembly language and high level languages
- Introduction to assembler
- Implementing basic computation tasks using assembly language
- Implementing a conditionally executed blocks. Computing logic and arithmetic conditions
- Design of subprograms
- Argument passing to subprograms.
- Local variables handling.
- Implementing interrupt service routines
- Implementing multitasking - cooperative and preemptive cases
- Designing memory system
- Designing and interfacing peripheral devices

Laboratory (30 hours):
- Programming embedded systems using C/C++
- Running On Chip Debug with CAD/EDA programming tools
- Utylizing inspection tools like a DataVizualizer software
- Controlling resources of microcontroller
- Servicing interrupts
- Interfacing real hardware. Handling: displays LED, LCD and graphic LCD, keyboards, sensors
- Concurrent operation of the program
- Optimization of program for embedded systems
- Implementation of peripheral devices using HDL and FPGA technology.
- Binding the custom peripheral device with high level programming language
- Creating and using bus functional models in verification

ECTS:6
sum of hours: 150h (75h contact + 75h student's own work)
lecture: 30h
classes:15h
laboratory:30h
student's own work:75h (preparation for laboratory classes, familiarization with the technical documentation of peripheral systems used in the laboratory)

Bibliography:

1. M. Morris Mano: Computer System Architecture
2. M. Morris Mano, C.R. Kime: Logic and Computer Design Fundamentals.
3. W. Wolf: Computers as Components: Principles of Embedded Computing Systems Design
4. S. Furber ARM System-on-chip Architecture
5. N. Wirth: Algorithms + Data Structures = Programs
6. B.W. Kernighan, D.M. Ritchie: The C Programming Language

Learning outcomes:

Knowledge: knows and understands
1. Computer architecture issues, in particular the hardware layer (K1A_W06)
2. issues in the field of methodology and programming techniques (K1A_W07)
3. issues in the field of architecture and software of microprocessor systems (high and low level languages) (K1A_W08)
Skills: able
1. formulate a specification of simple electronic systems at the level of implemented functions, also using hardware description languages (K1A_U14)
2. design simple electronic circuits and systems intended for various applications, including simple digital signal processing systems (K1A_U16)
3. use catalog cards and application notes in order to select appropriate components of the designed electronic system or system (K1A_U17)
4. formulate an algorithm, uses high and low level programming languages and appropriate IT tools to develop computer programs that control the electronic system and to program microcontrollers or control microprocessors in the electronic system (K1A_U22)

Social Competence: is ready to
1. critical evaluation of knowledge and perceived content and recognition of the importance of knowledge in solving cognitive and practical problems as well as consulting experts in the event of difficulties in solving the problem on its own (K1A_K01)
2. submit to the rules of team work and take responsibility for jointly performed tasks (K1A_K04)

Assessment methods and assessment criteria:

Lecture: Exam - 1/2 of final grade
Laboratory: Completing all assigned laboratory tasks - 1/2 of final grade
Final course grade (FCG) is calculated as follows:
FCG = 0.5 *Lecture grade + 0.5 * laboratory grade,
rounded to the university grading scale.

The syllabus is valid from academic year 2024/25 and its content cannot be changed during the semester.

Element of course groups in various terms:

Course group description	First term	Last term
missing group description in English (EiTAu>SI_5)	2020/2021-Z	

Course credits in various terms:

Electronics and Telecommunication, full-time first degree engineering studies 7 sem. (EiTAu-SI7)

Type of credits	Number	First term	Last term
European Credit Transfer System (ECTS)	6	2020/2021-Z	