COURSE DESCRIPTION CARD - SYLLABUS

Course name

Tools for designing microsystems

Course

Field of study	Year/Semester
Computing	1/1
Area of study (specialization)	Profile of study
Edge Computing	general academic
Level of study	Course offered in
Second-cycle studies	Polish
Form of study	Requirements
full-time	elective

Number of hours

Lecture	Laboratory classes	Other (e.g. online)
15	30	0
Tutorials	Projects/seminars	
0	0	

Number of credit points

4

Lecturers

Responsible for the course/lecturer:

Szymon Szczęsny, PhD. Eng.

email: szymon.szczesny@put.poznan.pl

tel. 61 6652297

Faculty of Computing and Telecommunications

Piotrowo 3a st. 60-965 Poznań

Responsible for the course/lecturer:

Damian Huderek, MSc. Eng.

email: damian.huderek@put.poznan.pl

tel. 61 6652297

Faculty of Computing and Telecommunications

Piotrowo 3a st. 60-965 Poznań

Prerequisites

A student starting this course should have basic knowledge of digital and analog electronics, as well as basic knowledge of electrical engineering. An additional requirement is the ability to read catalog cards, understand the documentation of electronic components, subassemblies and microsystems. It is also necessary to be able to expand knowledge and work in a team. Concerning social competencies, the student should be aware that knowledge in computer science quickly becomes obsolete and requires constant expansion. The student should present an attitude of honesty, creativity, reliability and cognitive curiosity.

Course objective

1. Providing students with basic knowledge in the field of designing dedicated electronic devices

(intended for implementing such tasks as: measurement and recording, remote supervision, data transmission).

2. Discussing the methodology of comprehensive design of electronic printed circuits beginning with a concept, formulating a concept schematic diagram, to the visualization of the final product, taking into account issues concerning version management and design rules.

3. Acquainting with popular tools for designing printed circuits and methods for verifying correctness of a design.

4. Presenting case studies illustrating various implementations of typical functional blocks found in microsystems: power circuits, programmable units, communication interfaces.

5. Developing students' skills of practical use of knowledge in the field of electronics to implement given design tasks, the ability to use technical documentation to implement a defined design task.

6. Developing the ability to critically evaluate existing solutions in terms of meeting a given criterion (e.g. power consumption, physical dimensions of modules, sub-circuits response times).

Course-related learning outcomes

Knowledge

1. has advanced and in-depth knowledge of broadly understood IT systems, theoretical foundations of their construction, as well as methods, tools and programming environments used to implement them [K2st_W1]

2. has ordered and theoretically founded general knowledge related to key issues in the field of computer science [K2st_W2]

3. has advanced detailed knowledge of selected issues in the field of computer science [K2st_W3]

Skills

1. can - when formulating and solving engineering tasks - integrate knowledge from various areas of computer science (and, if necessary, also knowledge from other scientific disciplines) and apply a system approach, also taking into account non-technical aspects [K2st_U5]

2. can assess usefulness and possibility of using new achievements (methods and tools), as well as new IT products [K2st_U6]

3. is able to make a critical analysis of existing technical solutions and propose improvements [K2st_U8]

4. can assess usefulness of methods and tools for solving an engineering task of constructing or evaluating an IT system or its components, including limitations of these methods and tools [K2st_U9]

5. can - in accordance with a given specification taking into account non-technical aspects - design a complex device, IT system or process and implement the project - at least in part - using appropriate methods, techniques and tools, including adapting existing or developing new tools for the purpose [K2st_U11]

Social competences

1. understands that in computer science knowledge and skills become obsolete very quickly [K2st_K1]

2. understands the importance of using the latest knowledge in the field of computer science in solving research and practical problems [K2st_K2]

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment:

a) concerning lectures:

– based on answers to questions concerning the material discussed during previous lectures

b) concerning laboratories:

– based on assessment of current progress of implementation of tasks

Summative assessment:

a) concerning lectures, assumed learning outcomes are verified with:

– assessing knowledge and skills demonstrated in a problem-based written test

– discussing the results of the final test

b) concerning laboratories, assumed learning outcomes are verified with:

– evaluating the report prepared partly during the course and partly after its completion; this assessment also includes teamwork skills; concerns reproductive laboratory exercises (the student carries out the exercise according to the provided instructions)

– assessing the implementation of a complex task requiring integrating knowledge and skills acquired during laboratory classes; the assessment concerns technical aspects of implementation, ability to solve unconventional problems and proficiency in using available design tools

Getting extra points for activity during classes, especially for:

– discussing additional aspects of an issue

– effectiveness of applying the acquired knowledge for solving a given problem

– the ability to cooperate as part of a team practically carrying out a detailed task in the laboratory

– comments related to the improvement of teaching materials

– identifying students' perceptual difficulties enabling ongoing improvement of the teaching process

Programme content

The lecture program covers the following topics:

Features and functional requirements of electronic devices: degrees and classes of insulation, galvanic separation; electromagnetic compatibility and immunity (EMI, EMC); modularity, complementarity, substitutability in the implementation of devices for embedded systems; metaproducts and full-custom devices. Methodology of designing printed circuits: review of the production technology of printed circuits; rules for creating schematic diagrams, component hierarchy, using library components, creating a network of connections between components, using bundles, creating connection classes, managing component designators, applying ERC design rules; component database management, creating symbols for components with different levels of abstraction (ideological symbol, housing, 3D model, simulation model); rules for implementing a connection mosaic (layout); defining and maintaining differential pairs, techniques supporting paths, autorouting, control of DRC design rules; three-dimensional visualization of designed devices, rules of component description in Step models. Implementations of power and supply stages (impulse, continuous, monolithic, protections, monolithic, discrete bridges, unipolar and bipolar H-type, DC, BLDC, multiphase, stepper motor drivers, impulse voltage converters drivers). Control units (including digital and analog implementations of input/output interfaces, including their protections). Current and voltage measurements (systems dedicated to measuring voltages and currents, direct and differential method, using the Hall effect), construction and operation of selected sensors depending on environmental conditions (temperature, pressure, humidity, acceleration), pyroelectric effect, methods of gas concentration detection and dust, PIR detectors. Shaping the properties of the feedback path, filtration, Kalman filter, windup phenomenon. Communication interfaces - overview of integrated communication systems for MODBUS, CAN, Ethernet networks, basics of transmission protocols, methods of compiling the transmission in the microprocessor system. Creating models of peripheral devices using the VHDL-AMS language. Modeling complex digital and mixed systems using the AMS standard.

Laboratory classes are conducted in the form of thirty hours of laboratory exercises, preceded by a 2-hour instructional session at the beginning of the semester. All laboratory classes are conducted by 2-person teams of students. The laboratory program covers the following topics: advanced methods of designing a hardware layer of a microsystem based on the Altium Designer environment (selecting components, creating a schematic diagram, developing own library components, implementing a mosaic of connections, 3D visualization for the designed PCB). The summary of the laboratory classes is an implementation of two simple mini-projects of a printed circuit, taking into account methods and techniques learned during previous laboratory classes. Implementation of the project requires an independent selection of components based on documentation, taking into account design rules at each of the design stages and the efficient use of design tools in order to obtain a topography suitable for fabrication dedicated to the embedded system.

Teaching methods

1. Lecture: multimedia presentation supplemented with examples on a board

2. Laboratory exercises: problem solving, practical exercises, data analysis, simulation, discussion, team work, case studies, multimedia presentation.

Bibliography

Basic

1. The industrial electronics handbook Wilamowski B, Irwin D., Taylor & Francis, 2011

2. Silniki elektryczne w praktyce elektronika, Przepiórkowski J., BTC, Wa-wa, 2007

3. Komputerowe systemy automatyki przemysłowej, Kwiecień R., Helion, 2012

4. Metrologia elektryczna, Chwaleba A, Poniński M., Siedlecki A., WNT, 2007

5. Podstawy technologii montażu dla elektroników, Kisiel R., BTC, 2012

Additional

1. Programowalne moduły Ethernetowe w przykładach, Chruściel, M. BTC, Wa-wa 2013

2. Linux. Podstawy i aplikacje dla systemów embedded Skalski Ł., BTC, Wa-wa 2012

3. A. Handkiewicz, P. Katarzyński, S. Szczęsny, M. Naumowicz, M. Melosik, P. Śniatała, M. Kropidłowski, Design automation of a lossless multiport network and its application to image filtering, Expert Systems with Applications, vol. 41, Issue 5, 2014

4. A. Handkiewicz, S. Szczęsny, M. Kropidłowski, Over rail-to-rail fully differential voltage-to-current converters for nm scale CMOS technology, Analog Integrated Circuits and Signal Processing, vol. 18, Issue 1, 2018

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,0
Classes requiring direct contact with the teacher	45	2,0
Student's own work (literature studies, preparation for laboratory classes/tutorials, preparation for test, project preparation) [1]	55	2,0

[1] delete or add other activities as appropriate