CEIE UNDERGRADUATE PROGRAMME

OBLIGATORY COURSES – SUBJECT SYLLABI
Subject: Algebra and Analytic Geometry

<table>
<thead>
<tr>
<th>Level of studies: BSc</th>
<th>Code: CEIE_S1_11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teacher: dr. Iwona Nowak</td>
<td></td>
</tr>
</tbody>
</table>

**Pre-requisite qualification:**

The knowledge and skills (on average level) in mathematics on the secondary school level are required.

**Course objectives:**

The aim of education is efficient use of the basic mathematical apparatus (related to the complex numbers, matrix and vector calculus, vector spaces and linear transformations) necessary for the further study, the ability to formulate problems and their description in the language of algebra and interpretation of results.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

45/0/30/0/0

**Syllabus description:**

**Lectures:**

Complex Numbers (def., operations in C, polar form, roots of complex numbers, exponential form); Polynomials; Matrices and Determinants; System of Linear Equations (Gaussian Elimination, Cramer Rule, homogenous systems, rank, Kronecker-Capelli Theorem,); Analytical Geometry (calculus of vectors, Line and plane In 3D space, curves of third degree); Vector Spaces (def.+examples, subspaces of vector space, spanning sets, linear dependence and independence, basis and dimension); Inner Product Spaces (def., length and distance in inner product spaces, complex vector space, complex inner product space, angle and orthogonality, orthogonal basis, the Gram-Schmith orthonormalization process); Linear Transformations (transformation defined by matrix, kernel and range, standard matrix, composition of transformations, inverse transformation, matrix of nonstandard basis, transition matrix and similarity); Eigenvalues and Eigenvectors (def., eigenspaces, diagonalization, orthogonal diagonalization, Jordan canonical form)

**Classes:**

During classes students solve practical tasks related to the topic introduced in the lecture.

**References:**

Number of ECTS credits: \(5 + 4 = 9\)

Subject: Calculus and Differential Equations  
Code: CEIE_S1_12

Level of studies: BSc  
Semester(s): 1,2

Teacher: dr. Ewa Łobos

Pre-requisite qualification:
Mathematics from the secondary school (average knowledge is sufficient).

Course objectives:
The objective of the course is to give theoretical bases of mathematical analysis and its applications. Students should be able to formulate problems, describe them in the language of mathematics, and interpret obtained results.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
60/0/60/0/0

Syllabus description:

Lectures
The real number system – the real line – the absolute value.
The concept of a function; examples. Metric on a set – metric spaces. One-to-one onto functions; composite functions, inverse functions. Review of elementary functions; hyperbolic and inverse trigonometric functions.
Sequences (of real numbers, complex numbers, plane points). Convergence in a metric space. Properties of convergent sequences. Limits of some numerical sequences.
Undergraduate obligatory courses - subject syllabi

Antiderivatives and indefinite integrals. Techniques of integrations – integration by parts, the method of substitution. Integration of rational functions by partial fractions. Rationalizing substitutions.

The definite integral – definition, properties. Fundamental Theorems of Calculus. Applications (area between two curves, arc length). Improper integrals.


Double integral – definitions, properties, basic theorems and applications. Change of variables (polar coordinates).


Classes: practical problems, the interpretation of obtained results

References:

S. R. Lay, Analysis with an Introduction to Proof
B. Sikora, E. Łobos, A First Course in Calculus
E. Łobos, B. Sikora, Advanced Calculus – Selected Topics
E. Łobos, B. Sikora, Calculus and Differential Equations in Exercises
R. A. Adams, Calculus: a Complete Course

Number of ECTS credits: 5 + 5 =10

<table>
<thead>
<tr>
<th>Subject: Circuit Theory</th>
<th>Code: CEIE_S1_13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 1,2</td>
</tr>
</tbody>
</table>

Teacher: dr. Damian Grzechca

Pre-requisite qualification:

Course attendants are supposed to have general knowledge concerning mathematics (including the ability to solve algebraic equations, operations on complex numbers, differentiation and
integration of basic functions), physics (elementary concepts and laws such as the electrostatic field, familiarity with the basic electrical units).

Course objectives:

**CT1:** The main objective of the course is to provide the students with basic and advanced knowledge concerning linear and nonlinear direct current (DC) and time domain circuits. During the course the students should develop the skills concerning the analysis methods of DC circuits as well as time domain circuits.

**CT2:** The main objective of the course is to provide the students with basic and advanced knowledge concerning alternating current (AC) circuits, time domain circuits, transmission line and 3-phase circuits. The students should understand differences between these circuits, know how to analyze a circuit and possess skills of dealing with these circuits.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

60/0/45/0/15

Syllabus description:

Lectures:

**CT1:**

1. Introduction to circuit theory, circuit variables - basic terms and definitions, classification of circuit theory problems, dc analysis, circuit elements, classification.
3. Analysis of complex circuits, generalized Kirchhoff's analysis, node voltage (nodal) analysis.
Undergraduate obligatory courses - subject syllabi

12. 1st order circuit – s-domain method. 1st order circuit – boundary values based method.

CT2:

2. Laplace transform, Heaviside formula, Laplace transform dictionary.
5. AC steady-state power. Measures of power. Instantaneous power. Average or real power. Apparent power. Reactive power. Complex power. Maximum power transfer.
9. Analysis of circuit response when one circuit constant varies.
13. Transient analysis in transmission line.
15. Summary and exemplary practical problems.
Undergraduate obligatory courses - subject syllabi

Classes:

CT1:

2. DC circuit with practical sources, ideal and real ammeter and voltmeter
3. Nodal analysis method
5. Multi-terminal elements.

CT2:

1. Transients in the first order circuits
2. Transients in the higher order circuits. Heaviside formula.
3. Transients in circuits with non-zero initial conditions.
5. AC domain circuits. Phasors.
8. Transformers. Mutual inductance.
10. AC steady state analysis of transmission line (standing wave)
11. Three phase systems.

Laboratory

CT2:

1. Introduction to laboratory and to work with oscilloscope
2. Transient in first order circuits with zero initial conditions switched on a DC source
3. Transient in higher order circuits with zero initial conditions switched on a DC source
4. Transient in circuits with non-zero initial conditions
5. Resonance and frequency response
6. Transmission lines

References:


Number of ECTS credits: 5 + 6 = 11
**Subject: Fundamentals of Computer Programming**  
**Code: CEIE_S1_14**

| Level of studies: BSc | Semester(s): 1 |

**Teacher:** dr. Piotr Fabian

**Pre-requisite qualification:**

It is assumed, that the student has an elementary knowledge of mathematics at the secondary level and logical thinking skills, including abstract thinking.

**Course objectives:**

The course provides the knowledge required to understand, design and write computer programs in the C language. The aim of the course is to lay a solid foundation of good software engineering and programming language practice. The program contains: introduction to imperative programming in C language (basic knowledge required to create and understand programs as well as skills essential for good software engineering and programming practice), basic algorithms and data structures and some advanced problems and techniques essential for programmers. Lectures are illustrated with slides with many sample programs. They are supported by laboratories, which give students an opportunity to create programs on their own.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

30/0/0/0/30

**Syllabus description:**

**Lectures:**

- Introduction. The first program.
- Development environments
- Variables, basic types.
- Operators and expressions.
- Instructions and program control.
- The structure of a program
- Functions
- Memory management.
- Arrays and pointers, memory allocation.
- Structures and unions
- Dynamic data structures.
- The preprocessor, separate compilation.
- Header files and libraries

**Laboratory:**

Small programming exercises and one individual programming assignment.
Undergraduate obligatory courses - subject syllabi

References:

- B. Stroustrup, The C++ Programming Language. Addison-Wesley, Reading, MA.

Number of ECTS credits: 5

<table>
<thead>
<tr>
<th>Subject: Theory of Logic Circuits</th>
<th>Code: CEIE_S1_15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 1,2</td>
</tr>
</tbody>
</table>

Teacher: dr. Piotr Czekalski

Pre-requisite qualification:

Course objectives:

Theory of Logic Circuits presents to the audience a complete course covering wide aspects of modern digital system design (combinational, sequential, microprogrammable, programmable), it’s analysis and review. Student are presented step-by-step course on general two-value logic, numeric systems, algebra and arithmetic of digital devices, various synthesis and analysis methods related to the digital circuits along with review of digital devices and it’s utility.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/30/0/30

Syllabus description:

Lectures:

- Numeric systems.
- Converting numbers between different numeric systems.
- Binary forms of numbers and it’s representation in digital systems.
- Fixed point arithmetic.
- Information and communication – digital vs analogue world.
- Digital devices, circuits and systems.
- Boolean algebra, gates and binary operators.
- System functionally complete.
- Digital systems classification.
- Combinational circuits design.
- Synthesis and analysis of combinational circuits.
- Iterative circuits.
Undergraduate obligatory courses - subject syllabi

- Sequential digital systems.
- Asynchronous sequential systems design.
- Synchronous sequential systems design.
- Dynamics of sequential systems.
- Microprogrammable circuits design.
- Programmable logic devices.

Classes:

Classroom exercises cover practice of the subjects that are closely related to the lecture, particularly insisting on real problem analysis and solution.

Laboratory:

Laboratory course covers systems design on digital systems and computer systems. Students are creating and analyzing real digital systems, build of various operators and medium scale integration devices (including sequential-related components and microprogrammable related memory components).

Number of ECTS credits: 5 + 3 = 8

<table>
<thead>
<tr>
<th>Subject: Social Sciences I</th>
<th>Code: CEIE_S1_16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 1</td>
</tr>
<tr>
<td>Teacher: prof. Waldemar Czajkowski</td>
<td></td>
</tr>
<tr>
<td>Pre-requisite qualification:</td>
<td>none</td>
</tr>
</tbody>
</table>

Course objectives:

There are two main goals to be achieved in this course. First - to discuss the importance of social knowledge/science for the citizens' participation in democracy. Second - to analyze the most important global social (technological, economic, cultural etc.) processes and to demonstrate their challenging character.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/15/0

Syllabus description:

1. The notion of politics and democracy. Various types of democracy.
2. The importance of knowledge for contemporary societies. Social sciences as a form of collective self-knowledge.
3. The relevance of social sciences for democracy. The peculiarities of social sciences.
5. Globalization in historical perspective. Models of global historical process.
6. Great historical transformations (neolithic revolution, the rise of civilization, industrial revolution...)
7. Material basis of globalization (from telegraph and steamship to Internet and jet)
8. Institutional/organizational infrastructure of globalization (UN, World Bank etc.)
9. Ideological debates on globalization

References:
1. Jan Aart Scholte; Globalization. A critical introduction; Palgrave, New York 2005

Number of ECTS credits: 2

<table>
<thead>
<tr>
<th>Subject: Physics</th>
<th>Code: CEIE_S1_21</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 2,3</td>
</tr>
</tbody>
</table>

Teacher: prof. Jacek Szuber

Pre-requisite qualification:

Course attendants are supposed to have general knowledge concerning physics and mathematics at the level of secondary school, which allows the understanding of main physical phenomena in the nature.

Course objectives:

The aim of the course is explain the fundamental physical phenomena in the nature, combined with their mathematical description, for application in modern automatics, electronics and computer sciences.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

60/0/30/0/30
Undergraduate obligatory courses - subject syllabi

Syllabus description:

Lectures:
Semester 2:
Introduction to physical universe
Fundamentals of kinematics and dynamics of material point and rigid body
Motion in inertial and non-inertial frames
Conservation principles in mechanics
Mechanical vibrations
Wave motion and sound propagation
Thermal physics
Thermal gas processes
Thermodynamics of gases

Semester 3:
Gravitational field
Electrostatic field including dielectric phenomena
Magnetic field including electromagnetic induction
Electromagnetic radiation and wave optics
Quantum optics and wave properties of matter
Classical atomic models
Fundamentals of quantum mechanics and quantum theory of one-electron atoms
Band structure of solids including semiconductors.

Classes:

Semester 2:
Physical quantities, vectors and fundamentals of mathematical analysis
Kinematics and dynamics of material point and rigid body
Conservation principles for material point and rigid body
Mechanical vibrations
Wave propagation and sound waves
Gas transitions and kinetic theory of gases
Thermodynamics of ideal gas

Semester 3:
Gravitational field
Electrostatic field
Magnetic field
Wave optics
Quantum optics
Atomistic nature of matter
Undergraduate obligatory courses - subject syllabi

Laboratories:

Determination of the following physical parameters: gravitational acceleration g, moment of inertia of rigid body, coefficient of viscosity, focus of lenses, constant of diffractive mesh by light diffraction, work fuction of metals, lifetime of carriers in semiconductors, carrier concentration by Hall effect, absorption of beta radiation, spectral characteristics of excited atoms.

References:


Number of ECTS credits: 3 + 6 = 9

Subject: Computer Programming  Code: CEIE_S1_22
Level of studies: BSc  Semester(s):

Teacher: dr. Piotr Fabian

Pre-requisite qualification:

Course: Fundamentals of Computer Programming

Course objectives:

The aim of the course is to teach understanding, designing and writing programs in high level languages, particularly in C and C++. The course should give students a strong foundation of both theory and practice in the field of software development. It also introduces object oriented programming. Laboratory classes give student the opportunity to test their programming skills by writing sample programs and solving programming tasks. Practical examples require analyzing of source code and designing object oriented structure of programs.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

45/0/0/0/45

Syllabus description:

Lectures:

Introduction: The first sample program. Variables, types, operators, expressions. Instructions and flow control. Functions, the structure of a programs. Arrays, pointers, dynamic memory
Undergraduate obligatory courses - subject syllabi


Laboratories:


References:

• B. W. Kernigan, D.M.Ritche, ANSI C,
• B. Stroustrup, C++,
• International Standard for Information Systems—Programming Language C++, ANSI

Number of ECTS credits: 3 + 4 = 7

Subject: Discrete Mathematics

Code: CEIE_S1_23

Level of studies: BSc

Teacher: dr. Edyta Hetmaniok

Pre-requisite qualification:

Knowledge of mathematics at the secondary school level is required

Course objectives:

Goal of this subject is to present the basic fields in mathematics concerning the discrete structures, mathematical logic, techniques of theorem proving, which makes an important supplement for mathematical analysis, algebra and analytical geometry.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/15/0/0

Syllabus description:

Lectures:
Undergraduate obligatory courses - subject syllabi

ELEMENTS OF COMBINATORICS – permutations, function of n-factorial (n!), combinations, binomial coefficient, binomial formula. LOGIC – calculus of logic sentences, truth-tables, tautologies and contradictions, quantifiers. METHODS OF THEOREMS PROVING – direct proof, modus ponens, modus tollens, proof by contradiction and contrapositive, proof of equivalence, proof by mathematical induction. SETS – definition and notation, operations on sets, Venn diagrams, Cartesian product, size (the cardinality) of the set, countable and uncountable sets. RELATIONS – definition and properties, graph of the relation, equivalence relation, relation of partial and total (linear) order. FUNCTION AS A RELATION – function in the sense of relation, surjective, injective, bijective function. INTRODUCTION TO GRAPH THEORY – definition and operation on graphs, graph isomorphism, walk, trail, path, trees. Z TRANSFORM AND DIFFERENCE EQUATIONS – recurrence relation and methods of its solving: iteration method, by using the generating functions, by using the z transform.

Classes:

1. **Elements of combinatorics** – solving tasks by using the combinatoric constructions.
2. **Mathematical logic** – determining the logic value of statements, giving the converse, inverse and contrapositive of the statement. Examples of theorems proving and mathematical induction.
3. **Sets** – testing properties of sets, constructing the generalized union and intersection of the given indexed family of sets, checking whether the given set is countable or uncountable.
4. **Relations** – testing properties of relations and drawing its graph, identifying the equivalence and the order relations, constructing the equivalence classes, finding the extremal elements of ordered set.
5. **Function as a relation** – constructing the image and contrainmage, verifying whether the function is surjective, injective or bijective.
6. **Introduction to graph theory** – solving tasks concerning the special kinds of graphs, walks, trails and paths, identifying the Eulerian and Hamiltonian graphs, trees constructing.
7. **Difference equations and z transform** – solving the recurrence relations by using the generating function and z transform.

References:


**Number of ECTS credits:** 3

---

**Subject:** Optimization and decision making  
**Code:** CEIE_S1_31

**Level of studies:** BSc  
**Semester(s):** 3
Undergraduate obligatory courses - subject syllabi

Teacher: prof. Jarosław Śmieja

Pre-requisite qualification:
Calculus and differentia equations, Algebra and analytic geometry, Fundamentals of computer
programming. Prior to this course, students should learn how to calculate derivatives, solve
linear differential equations, perform calculations using matrix notation and develop simple
software given the algorithm.

Course objectives:
Introduction of various types of practical optimization problems and analytical methods for
solving them. Development of skills necessary for application of basic numerical algorithms.
Introduction of mathematically based decision making in multistage decision processes.

Laboratory exercises aim at teaching how to implement theoretical results and standard algo-
rithms to find solutions of real life optimization problems

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/30

Syllabus description:

Lectures:

- Examples of optimization problems; defining optimality criteria
- Modeling of decision processes
- Unconstrained extrema
- Examples of using necessary conditions
- Constrained optimization
- Necessary conditions for constrained minimum
- Inequality constraints
- Convex programming
- Linear programming and simplex algorithm
- Dynamic programming
- Zero- and nonzero sum games
- Decision trees

Laboratory:

- Analytical methods of solving constrained and unconstrained static optimization prob-
lems
- Optimization of functions of one variable
- Multivariable optimization
- Linear programming, simplex method
- Quadratic programming
Direct methods of dynamic optimization, gradient methods  
Dynamic programming  
Decision trees/game theory

References:


Number of ECTS credits: 5

Subject: Probability and Statistics  
Code: CEIE_S1_32  
Level of studies: BSc  
Semester(s): 3  
Teacher: prof. Joanna Polanska, prof. Marek Kimmel  
Pre-requisite qualification:  
Courses: Algebra and Analytic Geometry, Calculus and Differential Equations  
Course objectives:  
The objective of this course is to give a theoretical basis of probability theory and statistics in very general context and to demonstrate the possible applications of this theory to applied models in system engineering, in operation research, and time series.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):  
30/0/0/0/30  

Syllabus description:  
The course consists of two parts: probability theory and applied statistics. The probability part starts with set theoretic concepts such as sigma-algebras, and denumerable operations on sets. Then, probability is introduced as a denumerably additive nonnegative normed set function. Properties of probabilities, including conditional probability follow. Random variables are introduced as measurable maps from the probability space in to the set of real numbers with Borel sigma-algebra. Distribution functions are discussed, including important examples of continuous and discrete distributions binomial/Poisson, geometric, uniform, exponential, normal, multivariate normal, gamma and chisquare). Independence of events is shown to lead
to strong results such as Borel-Cantelli theorems and Kolmogorov 0-1 law. Expected values are defined as Lebesgue integrals of random variables. Monotone and Dominated Convergence theorems follow. Law of Large Numbers and Central Limit Theorem are discussed. The second part starts with the survey of the methods of basic statistical testing where special emphasis is put on the hypothesis tests for the mean and variance of a normal population. Then the nonparametric methods are introduced followed by the ANOVA algorithms. Next we focus on the way of describing the relations among random variables. We introduce the measures of correlation (for both Gaussian and non-Gaussian random variables) and basic statistical tests. We give a general introduction to linear regression and consider the estimation problem for unknown parameters of probability distribution. Here we discuss the following three main methods: maximum likelihood method, the least squares method and the method of moments. Finally we present the basis of the analysis of frequencies.

All the theoretical material is broadly illustrated by the examples whose purpose is to help understanding the theoretical concepts and to show the possibility of applications of the probability methods in engineering practice.

1. Probability theory, part 1 – total and conditional probability
2. Probability theory, part 2 – distribution functions
3. Descriptive statistics
4. Graphical representation
5. Elements of theory of estimation. Point and interval estimators
6. Parametric tests, part 1 – testing for population mean,
7. Parametric tests, part 2 – testing for population variance
8. Goodness of fit tests
9. Nonparametric tests - unpaired measurements
10. Nonparametric tests – paired measurements
11. Inference for proportions – testing for population proportion
12. Inference for proportions – odds ratio, test of independence
13. Measures of correlation, linear regression

References:
2. Sokal RR, Rohlf JF: Biometry WH Freeman, 3rd edition or later

Number of ECTS credits: 5
Pre-requisite qualification:

Course attendants have to possess basic knowledge in calculus, algebra, physics and circuit theory.

Course objectives:

The objective of the course is to provide basic understanding of the operating principles of semiconductor devices and an introduction to the theory and operation of electronic circuits.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

60/0/30/0/30/

Syllabus description:

Lectures:


Classes:


Laboratory:

1. Introduction to measurement instruments
Undergraduate obligatory courses - subject syllabi

2. Semiconductor diodes
3. Bipolar transistors
4. Unipolar transistors
5. Optoelectronic devices
6. Rectifier circuits
7. Linear voltage regulators
8. Sine wave oscillators
9. Applications of operational amplifiers
10. Square wave and ramp oscillators
11. Power amplifiers

References:


Number of ECTS credits: 3 + 6 = 9

Subject: Introduction to System Dynamics  Code: CEIE_S1_34

Level of studies: BSc  Semester(s): 3

Teacher: prof. Andrzej Polański

Pre-requisite qualification:
Algebra and analytic geometry, Calculus and differential equations, Physics

Course objectives:
The aim of the course is making students familiar with problem and methods related to modeling physical dynamical systems and dynamical systems as engineering constructions.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/15/0/0

Syllabus description:

Lectures:
Undergraduate obligatory courses - subject syllabi


Classes:

Modeling physical and engineering systems by using the method of balances. Lagrange equations I Lagrange equations II Electromechanical analogies I Electromechanical analogies II Linear systems Nonlinear systems and state space

References:


Number of ECTS credits: 5
Mathematics. It is assumed that before starting to attend this course student has adequate knowledge in the field related to linear algebra, calculus and differential equations.

Course objectives:

The aim of the course is to teach basic terms related to numerical analysis and numerical methods used for approximate solving fundamental engineering problems. After completing the course students should be ready on his own to choose proper numerical method for solving fundamental engineering problems.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/0/30

Syllabus description:

Lectures:


Computation of function values. Horner's scheme and its applications: evaluating polynomial values, dividing polynomial by binomial, determining interval of real roots, evaluating values of polynomial derivatives, determining decimal representation of numbers of different positional numeral systems, expanding polynomial in terms of powers of binomial, determining coefficients of polynomial after change of variable.


Laboratory:

1. Theory of Errors
2. Calculation of Function Value
3. Interpolation
4. Numerical Differentiation
5. Numerical Integration
6. Approximation
7. Solving Systems of Linear Equations
8. Determining Eigenvalues And Eigenvectors of Matrix
9. Approximate Solving of Non-Linear Equations
10. Approximate Solving of Ordinary Differential Equations
11. Approximate Solving of Partial Differential Equations
12. Approximate Solving of Integral Equations

References:

Number of ECTS credits: 4

Subject: Digital Circuits  
Code: CEIE_S1_42

Level of studies: BSc  
Teacher: dr. Wojciech Sakowski  
Semester(s): 5

Pre-requisite qualification:
Course attendants are supposed to have general knowledge concerning Boolean Algebra and Automata Theory that enable design of combinational and sequential logical circuits. It is assumed that students passed the course: Theory of Logical Circuits.

Course objectives:
Course is a part of specialized curriculum content and covers basics of digital hardware design. The course objectives include having the students got acquainted with available hardware component base, methods of implementing a specified functionality in hardware and related design methodologies and tools.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/15/0/15

Syllabus description:

Lectures:
Basic information about digital signals: quantization and coding, binary codes, binary coded decimal numbers (BCD), fixed point positive and negative numbers, symbolic data representation. General description of digital integrated circuits: scale of integration, digital circuits families.

TTL family: structures and parameters of basic gates, immunity to interference – noise margins, comparison of TTL integrated circuits with ECL, CMOS and BiCMOS devices; basic operation and static and dynamic characteristics of TTL and CMOS gates.

Combinational circuits: complex gates, multiplexors, decoders.

Sequential circuits: flip-flops, registers, shift registers with feedback, counters.

Arithmetic circuits: adders and subtractors for binary and BCD Excess 3 numbers, number comparators, multipliers, floating point arithmetic.
Memories: static and dynamic RAMs, ROMs, UVEPROMs, EEPROMs, Flash memories. Programmable logic devices: simple PLDs, basic knowledge of FPGA.

Control unit design: hardwired control units, microprogrammed control units. Introduction to PLD/FPGA and ASIC design methodologies and hardware description languages (ABEL, VHDL).


Classes:
- Basics of interfacing digital gates with analogue elements
- Register & Counters
- Arithmetic circuits
- Microprogrammable Logic Circuits
- FPLA Devices

Laboratory:
- Static and Dynamic Gate Characteristics
- Arithmetic Circuits
- Microprogrammable Logic Circuits
- Digital Frequency Meter
- FPLA Devices

References:
Jerry D. Daniels, Digital Design from Zero to One, John Wiley & Sons, 1996
John P. Hayes, Digital Logic Design, Addison Wesley, 1993

Number of ECTS credits: 5
Undergraduate obligatory courses - subject syllabi

Objectives of the course is to acquaint the students with measurement systems. A measurement system is recognized as an information system which presents an observer with numerical value corresponding to the variable being measured. The course is intended as a reference for users of sensors, sensors systems, transducers and transducers systems. The scope of this course presents the basic principles of advanced sensor and transducers systems.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/30

Syllabus description:

Lectures:

Introduction: scope of lectures, literature; integration of intrinsically safe field instrumentation into industrial communication networks; intelligent sensors; institutions: IMEKO, IFAC, EUROSENSORS, PSST – Polish Society of Sensors Technology, COE – Optoelectronic and Electronic Sensors.

Smart sensors: Measurement of fluid flow by means of pressure differential devices - orifice plates and Venturi tubes. Smart interface. The essential sub-systems; list some of the main sensor defects. Zener Barriers (Ex).

The general measurement system: purpose, general structure, elements of system. Definition of sensor; sensor classifications. Example: “Weight measurement system” – elements of system; strain gauges (conventional and silicon).

Vocabulary of Basic and General Terms in Metrology: static characteristics - range, span, zero, zero drift, sensitivity, resolution, response, linearity, hysteresis, calibration, accuracy...; dynamic characteristics.


Non–Dispersive Infra-Red (NDIR) gas analyser: IR transmission characteristics, one path system, two path system, IR emitters, rotating chopper disc, reference cell, sample cell, radiation detectors (selective or non-selective), transfer equation

ITS-90 – The International Temperature Scale of 1990: triple points, freezing points, melting points, interpolation instruments – platinum resistance thermometer, gas and vapour thermometers, radiation pyrometer; interpolation equations; thermodynamic (Kelvin) and empirical (Celsius) scales.

Thermal radiation measurement system: high temperatures, moving body, temperature distribution over a surface; “black body”, Planck's law, emissivity of real body, characteristics of transmission medium; general form of thermal radiation measurement system, optical focus-
Undergraduate obligatory courses - subject syllabi

Ining system without and with lens, transmission characteristics, detectors – thermopiles, bolometers; total detected power, output signal.

*Pressure (pneumatic) measurement system:* elements of system; metal resistance Strain Gauge - tensile stress, compressive stress, longitudinal strain, transverse strain, elastic modulus, Young’s modulus, Poisson’s ratio, GF – Gage Factor; characteristics of system.

*Review of sensors:* conventional, thick, thin and semiconductor technologies; Strain Gages, Zirconia Cell (ZrO₂), magnetic (mechanical) sensors, electromagnetic sensors, chemical sensors, gas sensors, resistance and thermocouple sensors.

*Reliability of measurement systems:* reliability, unreliability, MTBF - Mean Time Between Failures, failure rate, variation of failure rate during lifetime of equipment – “bathtub” curve, reliability of a system of n elements in series or cascade, availability, methods of improving the reliability of measurement systems.

*Laboratory:*

Practical works in laboratories concerning instrumentation as a coherent, integrated subject of measurement systems. Examples of exercises:

1. **LabVIEW**
The introduction to the National Instruments graphical environment – LabVIEW. Front and block diagram (source code) creation, application debugging, array and matrix operation, graphical presentation of measurement/calculation results.
   Equipment: PC with LabVIEW graphical environment.

2. **LabVIEW + DAQ**
Functional overview of the Data Acquisition boards architecture. Signal generation and acquisition with LabVIEW application – close look at how the signal is generated and acquired by the DAQ board.
   Equipment: PC with DAQ board and LabVIEW graphical environment, HP33120 arbitrary waveform generator, HP56003B digital oscilloscope.

3. **Digital to analog converters**
Investigation of the metrological characteristic of the D/A converters (zero and gain errors, differential and integral linearity, sensitivity to temperature, dynamic properties).
   Equipment: DAC tester, voltmeter, analogue oscilloscope, muffle furnace.

4. **Strain gauges**
Investigation of the strain gauge factor for conventional and piezoresistive strain gauges.

5. **Flow measurements**
Familiarize with the properties of some of the measurement sensors used in flow measurements (turbine, electromagnetic, ultrasonic and orifice plate flowmeters)
5. **Gas chromatography**
Familiarize with the gas chromatography. Investigation of column temperature and flow rate of carrier gas on chromatogram.
Equipment: gas chromatograph with TCD (thermo-conductive) detector

6. **Temperature measurements**
Familiarize with the widely used in industry temperature sensors – RTDs and thermocouples. Calibration of temperature sensor in one of the fixed point of the ITS-90 (freezing point of tin).

7. **GPIB Interface**
Familiarize with the modern measuring instruments used as test equipment in laboratories and industry, equipped with dedicated measuring interface – GPIB.

8. **Profibus DP**
Investigation of the metrological parameters of chosen measuring modules. System configuration, sensors connection (2, 3 and 4 wire RTD measurements, thermocouple cold junction compensation, current loop operation with HART protocol).

**References:**

**Number of ECTS credits: 5**

---

<table>
<thead>
<tr>
<th>Subject: Theory of Computer Science</th>
<th>Code: CEIE_S1_44</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 4, 5</td>
</tr>
</tbody>
</table>

**Teacher:** dr. Robert Brzeski

**Pre-requisite qualification:**
none

**Course objectives:**
The aim of the lecture is to delivery to students the information in the range of the basic notions of computer science. The aim of the classes and laboratory is to purchase by the students the skill in the
range of creating the algorithms, low-level programming, understanding of works of microprocessors and introduction with the basic structures of the data.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

30/0/30/0/30

**Syllabus description:**

**Lectures**

1. Algorithms
2. Turing machine
3. Introduction to formal grammars
4. Formal grammars - examples
5. Basic components of a computer
6. Introduction to architecture of a computer
7. Von Neumann's architecture, introduction to machine W
8. Designing an instruction set for machine W
9. Program control unit for machine W
10. Programming in assembly language of machine W
11. Advance programming in assembly language of machine W
12. Input / Output functionality
13. Interrupts
14. Introduction to operating systems
15. Problems of management of resources and synchronization

**Classes**

1. Algorithms,
2. Turing machine
3. Formal grammars
4. Designing instructions for machine W
5. Programming in assembly language of machine W
6. Management of resources and synchronization

**Laboratories**

1. Designing instructions for machine W,
2. Programming in assembly language of machine W
3. Machine W - Input/Output
4. Machine W - Interrupts
5. Data access methods
6. Turing machine

**References:**
Undergraduate obligatory courses - subject syllabi

1. Tanenbaum, Andrew S.: "Structured Computer Organization"

**Number of ECTS credits: 5 + 3 = 8**

<table>
<thead>
<tr>
<th>Subject: Social Sciences II</th>
<th>Code: CEIE_S1_45</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 4</td>
</tr>
</tbody>
</table>

**Teacher:** prof. Waldemar Czajkowski

**Pre-requisite qualification:**
The basic knowledge of Social Sciences I

**Course objectives:**
The main objective of the course is to present an outline of the development of technology and science, including the feedback between S&T and economy, politics and culture. Emphasis will be given to the 20th century development, in particular to the computer/Internet revolution and the development of information/knowledge civilization/society.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**
15/0/0/15/0

**Syllabus description:**

1. Introductory remarks on history of science & technology and global history
2. Technologies of information/knowledge: language, writing, print,
3. Mythology - Philosophy - Science
4. Outline of the history of mathematics, physics and chemistry
5. Outline of the history of geology, biology and medicine
6. Outline of the history of computers hardware and software
7. Social, cultural and political history of computers and Internet

**References:**

Subject: Control Fundamentals

Level of studies: BSc

Teacher: prof. Ryszard Gessing

Pre-requisite qualification:
Mathematics, Physics, Introduction to system dynamics. It is assumed that the student has the basic knowledge concerning linear differential equations, Laplace and Z transform, as well as the models of the systems, especially their dynamical properties.

Course objectives:
The objective of the lectures is to give basic control knowledge in the fields of analysis and design of linear control systems, continuous and discrete-time, single and multivariable. The objective of classes and laboratory exercises is to acquire some practice in control system analysis and design using advanced CAD environment, like MATLAB-SIMULINK

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
45/0/30/0/30

Syllabus description:

Lectures:

Introduction to the course. Watt centrifugal governor Feedback Control Systems-basic notions, dynamic and static elements, block diagrams. Control system classification.


Basic elements and their responses. Time and frequency responses of the basic elements: first order lag, second order, ideal integrator and differentiator, system with delay.

Undergraduate obligatory courses - subject syllabi


Classes

1. Dynamic systems description
2. Frequency responses
3. Hurwitz stability criterion
4. Nyquist stability criterion
5. Nyquist stability criterion - systems with time delay
6. Steady state analysis
7. Root locus method
8. Stability degree and resonance degree
9. Systems quality - frequency domain methods
10. Sampled data systems

Laboratories

1. CAD of control systems – Matlab introduction
2. Stability of linear systems
3. Static accuracy
4. Phase-lead and phase-lag compensation
5. Attenuation index and tracking index
6. PID controllers
7. Root locus method
8. Sampled-data systems

References:

Number of ECTS credits: 6 + 2 = 8

<table>
<thead>
<tr>
<th>Subject: Artificial Intelligence</th>
<th>Code: CEIE_S1_52</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 5</td>
</tr>
</tbody>
</table>

Teacher: prof. Ewa Straszecka

Pre-requisite qualification:

Course attendants are supposed to have general knowledge concerning computers and computer applications. They have either be able to use at least one high level programming language or an advanced numerical tool like e.g. Matlab. It is assumed that students passed the following courses: Fundamentals of Computer Programming, Theory of Computer Science.

Course objectives:

Aim of the study is to give a definition and a review of AI, together with more careful investigation in selected areas. A student has a chance to learn among others about knowledge representation, reasoning, fuzzy sets, neural networks and genetic algorithms and practice an implementation of the methods during laboratory work.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/0/30

Syllabus description:

Lectures


Laboratories

1. KOHONEN NETWORKS – TP
2. ECG MODELLING BY GENETIC ALGORITHMS – part I- function optimisation
3. ECG MODELLING BY GENETIC ALGORITHMS – part II- ECG analysis
4. NEURAL NETWORKS IN SIGNAL PROCESSING
5. MEDICAL DIAGNOSIS SUPPORT SYSTEM part I – basic probability assignment calculation
6. MEDICAL DIAGNOSIS SUPPORT SYSTEM part II – inference, incomplete data management
7. HEURISTIC CONCEPTS IN FUZZY SETS INTERPRETATION
8. EMOTION MODELING
9. EVOLUTIONARY STRATEGIES IN OPTIMIZATION PROBLEMS – part I – software preparation
10. EVOLUTIONARY STRATEGIES IN OPTIMIZATION PROBLEMS – part II – properties evaluation
11. ANT SYSTEMS - part I – software preparation
12. ANT SYSTEMS - part II – properties evaluation

References:

Number of ECTS credits: 4

Subject: Computer Networks  
Code: CEIE_S1_53

Level of studies: BSc  
Semester(s): 5

Teacher: dr. Jerzy Mościński

Pre-requisite qualification:
Course attendants are supposed to have general knowledge concerning computers and computer applications, including using computer networks services. Students are also supposed to possess practical skills concerning computers and Internet usage as well as programming in at least one high level programming language. It is assumed that students passed the following courses: Fundamentals of Computer Programming, Theory of Computer Science.

Course objectives:
Course is part of specialized curriculum content and is related to education in areas of computer network technologies, Internet techniques and communication technologies. The course aims objectives include having the students got acquainted with hardware and software solutions in computer networks, benefits from computer networks based communication, usage and administration of network operating systems and network programming.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/30

Syllabus description:
Lectures

Overall objectives of the course include providing students with basic as well as advanced knowledge concerning components of computer network: communication methods, server computers, client computers, network infrastructure. Types of services offered by computer network servers are covered in detail as part of the course. Internet services, Internet and TCP/IP protocol suite, TCP/IP protocol structure, physical, data, network, transport and application layers tasks in computer network are also considered.

Specific topics covered during course lectures and laboratory exercises include the following: computer networks types and systems; local area network LAN and wide area network WAN; packets, frames, reliable and unreliable transmission; LAN cabling systems, physical topology, interfaces; Internet and TCP/IP protocol suite; network servers and types of services; Internet network protocol, TCP/IP protocol family. TCP/IP protocol structure, tasks and services concerning the physical layer, data layer, network layer, transport and application layers in computer network; DNS system and its role in naming hosts in computer networks; WAN techniques, routing and tracing routes; internetworks, architecture and protocols; basic applications of computer networks services; using electronic mail systems; www pages and browsers; advanced elements of UNIX/Linux operating system using; multimedia networking applications; streaming audio and video with networks; quality of Internet services – differentiated and aggregated services models; security in computer networks; cryptography, authentication, certification, firewalls; Internet commerce; computer network management.

Laboratories

- UNIX/Linux operating system basics.
- Bash shell programming.
- DNS basics and configuration.
- Advanced e-mail system configuration and administration.
- Client/server programming.
- Internet Protocol details, IP addressing, DHCP service.
- DNS protocol, DNS server configuration.
- Simple www server programming and configuration.
- Security issues, firewall configuration.
- Virtual LAN configuration.
- Network protocols and applications analysis.
- E-learning support system design and setup.
- PHP programming, Internet based databases.

References:


Number of ECTS credits: 6
Subject: Fundamentals of Signal Processing  
Code: CEIE_S1_54

Level of studies: BSc  
Semester(s): 5

Teacher: dr. Katarzyna Mościńska

Pre-requisite qualification:

Prerequisites: Algebra, Calculus, Circuit Theory. Course attendants should possess satisfactory knowledge of the following issues: complex numbers, derivatives and integrals, AC circuits, frequency description of systems. Students are supposed to possess basic computer programming skills.

Course objectives:

The goal of the course is to make students acquainted with various methods of signals and systems representation. Course participants become familiar with fundamental methods of analog and digital signal processing. The course serves as foundation to more specialized courses, like digital signal processing and analog circuit design.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/0/30

Syllabus description:

Lectures

1. Introduction to signal processing: definition of signal. Signal properties. Some special signals of interest.
7. Fourier transform of discrete – time signals: definition, properties, use in signal processing.
9. Representation of a digital circuit: difference equation, block diagram, system function, pole – zero pattern.
10. DFT and FFT. Definition and properties of DFT. Linear vs circular convolution. Linear convolution with DFT. FFT decimation-in-time algorithm.
Laboratories

1. Signal generation and basic operations on signals.
2. Transfer function of a filter.
5. Individual task – Fourier series expansion and reconstruction.
6. Rectifiers and amplitude modulation.
7. Pulse amplitude modulation.
9. Z Transform and system function H(z)
10. Discrete Fourier Transform

References:


Number of ECTS credits: 5

<table>
<thead>
<tr>
<th>Subject: Microprocessor Systems</th>
<th>Code: CEIE_S1_55</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 5</td>
</tr>
</tbody>
</table>

Teacher: dr. Adam Milik

Pre-requisite qualification:

Course attendants are supposed to have general knowledge of digital circuits operation and design, algorithm implementation, basics of programming languages and principles of computer operations. Students are also supposed to possess practical skills concerning programming and algorithm implementation with use of high level programming language. It is assumed that students passed the following courses: Fundamentals of Computer Programming, Theory of Logic Circuits, Theory of Computer Science, Digital Circuits

Course objectives:

....

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
Syllabus description:

Lectures:

Introduction to microprocessor. Evolution from finite state machine through the microprogrammable device to microprocessor. Basic functional blocks of microprocessor and its feature are discussed. The influence to behavior and performance of computing system is shown.

Instruction execution. Harvard and Princeton (von Neumann) architectures. Sequential and pipelined execution concepts

Instruction set. Argument addressing model and addressing modes.


Interrupt system. Implementation concept. Classification of interrupt systems. Concept of vectorized interrupt system.


Arithmetic. Numeral systems. Integers signed and unsigned. Fixed point numbers. Floating point numbers. Arithmetic operations. Numerical algorithms: line drawing (Bresenham’s), trigonometric calculations and rotations (Cordic), Solving non-linear equation (Newton-Raphson)


Hardware Description Language (HDL) introduction to Verilog HDL. HDL vs Programming Language. Basic of automatic synthesis. Description of combinatorial blocks. Description of sequential blocks.

Laboratory:


Introduction to HDL logic synthesis and FPGA technology. Implementation of peripheral devices. Creating and using bus functional models in verification. Binding the custom peripheral device with high level programming language.

Project:

1. Implementation of mundane devices like advanced alarm clocks, timers, cycle computers etc. Students are supposed to implement software layer of the project that binds together simple hardware devices like displays, keyboards and other sensors into fully functional system. The attention is paid to simplicity of use and correct user interface simplicity.

2. Implementation of custom hardware device that supports operation of main design problem. Usually it is a specialized interface unit or arithmetic operation support device. Important part of the design concerns creating, modeling, synthesizing, implementing the device and linking it with microprocessor. The hardware component is linked with software operating by preparing appropriate declaration and drivers. Finally the component is used inside the design to proof its functionality.
References:

3. Steve Furber ARM System-on-chip Architecture
4. Niklaus Wirth Algorithms + Data Structures = Programs
5. Samir Palnitkar Verilog HDL

Number of ECTS credits: 4

Subject: Project Management  
Code: CEIE_S1_56

Level of studies: BSc  
Semester(s): 5

Teacher: dr. Seweryn Spałek

Pre-requisite qualification:

none

Course objectives:

The course of Project Management sets out to teach the basic theory and core methodologies one needs to successfully manage projects or participate in a project team

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/15/0

Syllabus description:

The lectures include the following topics:

I. The definitions of the following are discussed:
   1. Project
   2. Project Life Cycle
   3. Project Team
   4. Project Manager
   5. Project Sponsor
   6. Stakeholders

II. The definition of the success of the project is given.

III. The key areas of managing the projects are discussed:
   1. Project Integration Management
Undergraduate obligatory courses - subject syllabi

2. Project Scope Management
3. Project Time Management
4. Project Cost Management
5. Project Resource Management
6. Project Communications Management
7. Project Risk Management
8. Project Procurement Management

Project will cover the following topics:

1. Project Charter
2. Work Breakdown Structure (WBS)
3. Critical Path Method (CPM)

References:

7. US Def. Dept. - MIL-STD-881A

Number of ECTS credits: 1

---

Subject: Social Sciences III  
Code: CEIE_S1_57

Level of studies: BSc  
Semester(s): 5

Teacher: prof. Waldemar Czajkowski

Pre-requisite qualification:
Basic knowledge presented in the courses: Social Sciences I, Social Sciences II

Course objectives:
The main objective of this course is to make students familiar with the vast field of application of formal/mathematical/computational methods in the domain of social sciences while stressing both the theoretical and the practical relevance of these applications.
Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
15/0/0/0/0

Syllabus description:
1. Challenges for social sciences in the time of globalization. How computers can help to overcome some barriers social sciences face.
2. Artificial intelligence and artificial life - ideas to be both applied in and adapted to social sciences.
3. Basic ideas of game theory. Prisoner’s dilemma.
4. Computer simulations in studying more complex forms of prisoner’s dilemma.
5. The concept of artificial societies.
6. Raports to the Club of Rome as an example of computer modelling of global social processes

References:
2. William Poundstone; Prisoner’s Dilemma; Doubleday, New York 1993
3. Peter Convey, Roger Highfield; Frontiers of Complexity. The Search for Order in a Chaotic World; Ballantine Books 1996

Number of ECTS credits: 1

<table>
<thead>
<tr>
<th>Subject: Computer Graphics</th>
<th>Code: CEIE_S1_61</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 6</td>
</tr>
</tbody>
</table>

Teacher: prof. Konrad Wojciechowski

Pre-requisite qualification:
Computer Programming, Algebra and analytic geometry, Fundamentals of computer programming

Course objectives:
The course aims to provide the theoretical basis and the resulting 3D computer graphics algorithms, and selected topics of 2D computer graphics as well as providing the necessary practical experience acquired in result of the implementation of algorithms in the laboratory exercises. The lecture will enable students to get in touch with modern solutions in the field photo-realistic 3D graphics offered in world literature, create their own solutions to the projects as well as understanding fundamental conditions of modern computer animation
Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/0/30

Syllabus description:

Lecture:

- Rasterization algorithms.
- Texturing the basic concepts and application of textures, 2D texture and 3D texture procedure.
- External devices used in computer graphics systems, the basic input devices, output devices to create a permanent copy machine displaying images, methods of storage and display.
- Modelling the colour used in computer graphics: RGB, CMY, CMYK, HSV, HLS, YUV, YCbCr.

Laboratory:

- Raster algorithms
- Clipping i windowing
- 3D Transformations
- Hidden surface removal
- Illumination models
- Raytracing
Undergraduate obligatory courses - subject syllabi

- Object detection 1
- Object detection 2
- Bone animation
- Collision detection
- Particle effects
- Pixel and vertex shaders

References:

2. Peter Shirley: *Fundamentals of Computer Graphics*

Number of ECTS credits: 4

<table>
<thead>
<tr>
<th>Subject: Embedded Systems</th>
<th>Code: CEIE_S1_62</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 6</td>
</tr>
</tbody>
</table>

Teacher: dr. Krzysztof Tokarz

Pre-requisite qualification:
Theory of logic circuits, Digital circuits, Microprocessor systems

Course objectives:
Main goal of the course is to present elements of microprocessor and embedded systems like: micro-processors, memories, buses, peripheral devices. PC computer elements are also presented. The process of embedded system development is presented with attention on proper choice of hardware elements, operating system, programming methods. Methods of hardware-software design with co-design is presented. Specification and documentation preparation according to IEEE standards is also presented as important part of system development process

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/30
Syllabus description:

Lecture:

Introduction to microprocessor and embedded systems. Definition, classification and development methods of embedded systems. Elements of embedded system: microprocessor, microcontroller, memories, peripheral devices.


Operating systems for embedded systems, RTOS, cooperative and preemptive multitasking. Writing applications without operating system, superloop, interrupt driven software, finite state machine, examples of implementation.

Requirements for embedded system, IEEE standards for embedded system specification and documentation. Stages of system development: requirements analyze, general design, subsystem design, subsystem implementation, integration, testing, documentation, development errors.

Methods of hardware-software partitioning, co-design, selection of hardware elements. Hardware drivers.

Laboratory:

- In circuit emulators, debugging in embedded systems. Simulators. 8051 microcontroller.
- Serial synchronous data transmission. TWI, SPI.
- AVR microcontrollers, writing programs in assembler and C
- Simple programmable logic devices, Abel language.
- Interrupt controllers. Interrupt priorities.
- Digital signal processors.
- Example of cooperative real-time operating system

References:

3. Networking and internetworking with microcontrollers / by Fred Eady. - Burlington, Ma ; Oxford : Newnes, cop. 2004

Number of ECTS credits: 4
Subject: Databases  
Level of studies: BSc  
Teacher: dr. Paweł Kasprowski

Pre-requisite qualification:
Basic knowledge of any programming language.

Course objectives:
The purpose of the subject is to teach students how to develop and use modern database systems.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/30

Syllabus description:

Lectures

- Usage of databases – functions and architecture of Database Management System (DBMS).
- Relational model – relations, relationships, keys.
- Relational algebra – selections, projections, joins.
- Structured Query Language (SQL) - Data Definition Language (DDL), Data Manipulation Language (DML), Data Query Language (DQL).
- Searching in relational database using SELECT phrase.
- Advanced searching - grouping data, aggregations, views, outer joins, nested queries, correlations.
- Preserving database referential integrity - primary and foreign keys.
- Security in databases - users, roles, rights.
- Developing databases – functional dependencies, normal forms, ERD diagrams.
- Concurrent access to databases – locks, transactions, isolation levels.
- Programming in databases – stored procedures, functions, triggers.
- Architectures of modern database systems – client-server and 3-tier architectures.

Laboratories:

1. SQL language
2. Advanced SQL language – SELECT statement
3. SQL DDL/DCL – preparing users, rights, preserving referential integrity
4. Transactions and isolation levels
5. Constructing triggers and stored procedures
6. Preparing Entity Relationship Diagrams

References:
...

Number of ECTS credits: 5

<table>
<thead>
<tr>
<th>Subject: Operating Systems</th>
<th>Code: CEIE_S1_64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 6</td>
</tr>
</tbody>
</table>

**Teacher:** dr. Przemysław Skurowski

**Pre-requisite qualification:**
Theory of computer science, Computer programming

**Course objectives:**
The goal of a course is to introduce students into the contemporary operating systems which are considered as environments of effective resource managing environment and user interface layer in modern computer systems. During the course students will get knowledge on configuring and administering of operating systems and on the solutions of classical resource management problems with special focus on processor and memory related tasks.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**
30/0/0/0/30

**Syllabus description:**

*Lectures:*
Topics are related to the general purpose OS and to the general problems present in any kind of OS:

1. Basic concepts in OS topic: definition and fundamental roles, effectiveness criteria, processes, resources, types and architectures of OS
2. OS structure – kernel, drivers, tools, subsystems, interfaces and utilities.
3. Resource management and Inter process communication (IPC), concurrency, interference, mutual exclusion, process synchronization and communication means, semaphores, mailboxes
4. Algorithms and mechanisms of a CPU time sharing
5. Memory organization and allocation, virtual memory, memory protection
6. I/O devices handling in the OS
7. File systems – physical and logical representation
8. Hard disk head movement planning
9. Basics or realtime and distributed OS
10. Description of Windows and Linux OS

Laboratory topics

Windows 7 - Installation
Windows 7 – Administrative scripts
Windows 7 – Users, groups, permissions
Windows 7 – Basic network
Windows 7 – System services
Windows 7 – Remote access
Linux Ubuntu – Installation and configuration basics
Linux Ubuntu - Users, groups, permissions
Linux Ubuntu - Processes
Linux Ubuntu - Basic network
Linux Ubuntu – multi system collaboration
Linux Ubuntu – Fundamentals of Bash programming

References:

1. A. Silberschatz, J.L. Peterson, G. Gagne, Operating Systems Concepts, Wiley

Number of ECTS credits: 4

Subject: Algorithms and Data Structures

Code: CEIE_S1_65

Level of studies: BSc
Semester(s): 6

Teacher: dr. Piotr Fabian

Pre-requisite qualification:

It is assumed, that the student has an elementary knowledge of mathematics at the secondary level and logical thinking skills, including abstract thinking. An additional requirement is knowledge of English and the ability to write and understand simple programs.

Course objectives:
The aim of the course is to introduce students into advanced topics of algorithms and data structures. We present algorithms for sorting, searching, operating on graphs, trees. We discuss selected data structures: binary trees, heaps, priority queues. Students after this course should be able to analyze the complexity of algorithms, adapt known algorithms for new problems etc. Topics are illustrated with many examples. The course consists of lectures and classes (exercises). Classes are obligatory.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

30/0/0/0/30

**Syllabus description:**

**Lectures**

- Introduction, complexity of algorithms
- Simple sorting algorithms: selection sort, insertion sort
- Quick sort, k-selection
- Divide and conquer
- Heaps and their use: heap sort, priority queues
- Sorting with comparisons
- Sorting of specific data: integers, bucket sort, radix sort
- Binary Search Trees
- Hashing: chaining, open addressing, MPHF
- Exhaustive search
- Greedy algorithms
- Graphs, BFS, DFS
- The Dijkstra algorithm, the Floyd-Warshall algorithm
- Dynamic programming
- Pattern search

**Laboratories:**

Will follow the topics presented during the lectures.

**References:**

...

**Number of ECTS credits: 3**
Subject: Robotics

Level of studies: BSc

Teacher: dr. Aleksander Staszulonek

Pre-requisite qualification:

Before this course student should attend lectures on mathematical analysis, differential equations, algebra, physics and microcomputer programming. Strongly recommended is knowledge of assembly and C language programming. Knowledge of Lagrange equations is significant advantage as well. Additionally Mathematical analysis, Physics, Mechanics, System Dynamics.

Course objectives:

Course is part of specialized curriculum content and is related to education in areas of robotics, embedded systems, robot control systems and servomechanism control. The goal of this course is to provide students with the main elements of robot theory: mathematics, programming and control. The theory is complemented with laboratory exercises familiarizing students with practical aspects of robot control systems structure, control algorithms and programming.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/0/30

Syllabus description:

Lecture:

Program of the course includes: introduction to C and assembly language programming of embedded control systems, homogenous transformations, derivation of kinematic equations, kinematic equations solution, dynamics, control, trajectory execution and programming.

Section dedicated to homogenous transformations contains description of basic definitions like vectors, planes, coordinate frames, basic transformations, relative and inverse transformations, equivalent angle and axis of rotation. Section dedicated to derivation of kinematic equations deals with different coordinate systems, specification of A matrices for manipulator's prismatic and rotational links, specification of T matrices in terms of A matrices. As the example, kinematic equations of Stanford and Elbow Manipulators are derived. Methods leading to the solution of kinematic equations are described and solutions for Stanford and Elbow manipulators are presented. The dynamics of robot manipulators is then presented using Lagrangian equations. Requirements imposed on robot control systems are presented and set point and tracking control problems defined. Basic theory and methodology of robot control is presented on the examples of most frequently applied control structures. PID and sliding mode controllers are discussed.

Laboratory:

The laboratory exercises include the following tasks:

1. Robot control system structure and servomechanism control
Undergraduate obligatory courses - subject syllabi

2. Introductory aspects of C and assembly language programming of embedded controllers
3. Programming of communication between embedded system and servocontroller
4. Programming of single degree of freedom controller with digital PID algorithm
5. Implementation of desired trajectory specification
6. Programming of 6 degrees of freedom embedded controller and motion coordination
7. Synchronization of the robot.

References:

2. Itkis U.: Control Systems of Variable Structure

Number of ECTS credits: 3

<table>
<thead>
<tr>
<th>Subject: Mixed-signal circuits design</th>
<th>Code: CEIE_S1_67</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 6</td>
</tr>
<tr>
<td>Teacher: dr. Jerzy Fiołka</td>
<td></td>
</tr>
</tbody>
</table>

Pre-requisite qualification:

Course attendants have to possess basic knowledge in algebra, physics, circuit theory, electronic, signal processing, digital circuits. Students are also supposed to possess practical skills concerning design, simulation and construction of electronic systems.

Course objectives:

The main purpose of this course is to provide theoretical and practical knowledge to the students about the design of mixed-signal circuits.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/30/0

Syllabus description:

Lecture:

Mixed signal circuits design lecture covers the following topics:

1) Definitions and basic features of mixed-signal circuits;
2) Hardware and software subsystems od mixed-signal systems;
3) CAD tools for circuits design and simulation;
4) Sampling of continuous-time signals;
5) Operational amplifiers: types, structures, operation, parameters, applications;
6) Analog filters: fundamentals, design, simulation, realizations;
7) A/D converters: types, principle of operation, parameters;
8) D/A converters: types, principle of operation, parameters;
9) Sensor signal conditioning;
10) Phase-locked loop (PLL): principle of operation, parameters, applications;
11) Switched capacitor circuits (SC): principle of operation, parameters, applications;
12) Direct digital synthesis (DDS): principle of operation, parameters, applications;
13) Power supplies: types, structures, simulation, design
14) Interfacing Analog to Digital Circuits;
15) PCB Design for mixed-signal circuits;

Project:
Group of students (max. 3 people) choose project concerning design, simulation, construction and development of a mixed-signal circuit. The final result of the project is a working circuit and documentation.

References:

Number of ECTS credits: 3

Subject: CAD of control systems  
Code: CEIE_S1_71

Level of studies: BSc  
Semester(s): 7

Teacher: prof. Marian Błachuta

Pre-requisite qualification:
Algebra and Analytic Geometry, Introduction to System Dynamics, Control Fundamentals, Microprocessor Systems

Course objectives:
The objective of the lectures is to give fundamentals of numerical procedures and programs used for Computer Aided Design in Control Systems, while laboratory aims at fast prototyping tools used to de-
sign embedded controllers for selected laboratory plants. The theory is complemented with practical aspects of embedded control system.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/30

Syllabus description:

Lecture:
1. Introduction: History of CADCS, stages of design process, evolution of design tools, MATRIXx and MATLAB approaches to the design process.
2. Selected Problems of Linear Algebra: norms of vectors and matrices, typical norms, relationships between norms of vectors and matrices, typical induced norms. Particular types of matrices and their properties, orthogonal matrices and their properties, complex matrices, normal matrix, Hermitian matrix, unitary matrices and their properties, Schur matrix and eigenvalues of a matrix, unitary similarity transformation to a Schur matrix
5. Numerical procedures for control: algorithms for conversion from state-space to transfer function form, Markov parameters, canonical representations of state-space models and their relationship with transfer function, transformations between continuous-time and discrete-time systems, δ - operator models, computation of frequency plots, van Dooren algorithm for investigation of the structure of controllability and observability, controllability and observability Gramians, balanced realizations, model approximation, Lyapunov equations and associated numerical algorithms

Laboratory:
3. Data transmission in CAN network.
4. PWM control method. Reading analog values.
5. Modeling of the DC motor dynamics.
6. PID discrete regulation.
7. Prototyping of control system.
Undergraduate obligatory courses - subject syllabi

References:


Number of ECTS credits: 3

<table>
<thead>
<tr>
<th>Subject: Hierarchical control</th>
<th>Code: CEIE_S1_72</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 7</td>
</tr>
</tbody>
</table>

Teacher: prof. Krzysztof Fujarewicz

Pre-requisite qualification:

Students should have completed courses of: Mathematical analysis, Optimization and decision making, Control fundamentals.

Course objectives:

This course is addressed to students interested in systems analysis, control engineering, management and decision making. It covers basic methods used in solving control and optimization problems associated with large-scale and complex systems. After completing the course student has basic knowledge in optimization and control of large scale systems. This knowledge consists of methods of analysis of composite systems, solving structured optimization problems and designing hierarchical and decentralized feedback systems.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/0/30

Syllabus description:

Lectures:

1. Introduction and the terminology. Large scale systems, complex systems, decomposition coordination. Different types of hierarchical structures: multilayer structure and multilevel structure.


**Laboratory:**

1. Continuous-flow reactor. The aim of the exercise is modeling of a stirred-tank continuous-flow reactor, finding its static characteristics, and designing control system for the reactor.

2. Oil refinery optimization. The aim of the exercise is to solve the problem of optimal production in an oil refinery. The problem is formulated as a linear programming (LP) problem which is solved using Matlab software.

3. Direct method of coordination. The aim of the laboratory exercise is to apply the direct method of coordination to a complex static system composed of three cross-coupled subsystems.

4. Price method of coordination. The aim of the laboratory exercise is to apply the price method of coordination to a complex static system composed of three crosscoupled subsystems.

5. Dynamic programming. The aim of the exercise is to apply dynamic programming method in order to optimize an example of multistage process.


7. Control of complex systems I. During the laboratory students perform gradient-based optimization of close-loop control system.

8. Control of complex systems. Students make use of the adjoint sensitivity analysis in order to gradient-based optimization of a control signal for given non-linear system.

**References:**


Number of ECTS credits: 3

Subject: Applied digital signal processing
Code: CEIE_S1_73

Level of studies: BSc

Teacher: prof. Marek Pawełczyk

Pre-requisite qualification:
Calculus and differential equations, Fundamentals of computer programming, Computer programming, Introduction to system dynamics, Numerical methods, Fundamentals of signal processing. It is assumed that students have basic knowledge on: differentiation, integration, Laplace transform, Z transform, Fourier transform, transfer function, dynamic system modeling, numerical algorithms, Matlab environment.

Course objectives:
The aim of this lecture is to present issues in modern signal processing techniques with focus on applications. It constitutes a pedagogical compilation of fundamentals, algorithm forms, behavioural insights, and application guidelines. The intertwining of theory and practice is demonstrated by numerous examples and verified during project exercises.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/15

Syllabus description:

Lecture

In the era of rapid development of microprocessors, Digital Signal Processing (DSP) gains significant interest and finds applications in many fields of everyday life. DSP plays an increasingly central role in the development of telecommunications and information processing systems, and has a wide range of applications in multimedia technology, audio-visual systems, cellular mobile communications, adaptive network management, radar and ultrasonic systems, pattern analysis, medical signal processing, financial data forecasting, decision making, etc.

The lecture touches the following subjects:
1. Sampling, analogue-to-digital and digital-to-analogue conversion, quantization
2. Correlation analysis
3. Fourier decomposition and Fourier transforms
4. Signal windowing and spectral analysis
Undergraduate obligatory courses - subject syllabi

5. Conversion of sampling frequency and multi-rate signal processing
6. Finite and Infinite impulse response filters
7. Wiener and Kalman filters
8. Adaptive filters
9. Signal decomposition and forecasting
10. Fundamentals of electroacoustics
11. Active suppression of noise and vibration
12. Echo generation and cancellation
13. Speech intelligibility enhancement
14. Speech recognition and speaker identification
15. Ultrasonic signal processing
16. Vibration and acoustic signals processing for condition monitoring of working machines

Laboratory

Different project topics related to digital signal processing are offered each year. Some of them are performed individually, and some of them in groups of a few students. Students may suggest topics by themselves.

Exemplary topics are:
- Speech intelligibility enhancement
- Condition monitoring with audio or vibration signals
- EKG signal processing
- Active noise control
- Active vibration control

References:

Number of ECTS credits: 3

Subject: Computer architecture  Code: CEIE_S1_74
Level of studies: BSc
Teacher: dr. Zghidi Hafed
Pre-requisite qualification:
Theory of Computer Science

Course objectives:
The lectures are to familiarize students with main concepts related to computer architecture. They present the main directions of development of computer architecture, provide representative examples of computer organization. The main part of the lecture is to present the architecture of modern processors and parallel computers. The purpose of the laboratory is practical and introduce students to various computer architectures, different operating systems and technologies of parallel and distributed programming. Students are familiarized with computers based on Sparc processors, PowerPC, x86, running under control of different operating systems (Windows, Linux, Sun Solaris, OS/400).

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/30

Syllabus description:

Lectures
The history of the development of computer architectures: the first computers, the impact of electronic technology in computer architecture, subsequent generations of computers. Complex Instruction Set Computers (CISC). Reduced Instruction Set Computers (RISC): bases, implementation of RISC I computer. Pipelining, the problem of implementing jump instructions (jumps, delayed, branch prediction) and the problem of dependent data (changing the instruction order). Superscalar architecture and VLIW architecture. The specificity of the problems of pipelined instruction execution with dependent arguments in superscalar architecture. Renaming of registers. Examples of the superscalar processor architectures: UltraSPARC, Motorola, PowerPC, POWER. Hardware support for multithreading: fine-grained multithreading, coarse and concurrent.

Architecture of parallel computers. Classification of parallel systems - forms of parallelism: instruction-level parallelism and parallelism of processors, the Flynn classification, other classifications. Vector Computers: scalar and vector instructions - a vector computer concept, a review of vector instructions. Examples of vector computers, the use of vector computers. Array computers: the general approach, the
model in the implementation of SIMD commands, interconnect network, examples of array computer. SIMD model in modern superscalar processors. Graphics Cards and the CUDA architecture. Multi-threaded - SIMT model. multiprocessor systems

Systems with shared memory: cache coherence, MESI protocol different ways system elements connections - a common bus, multiple bus systems, cross-bar, multiport memory, multi-stage switch, non-blocking Clos network. NUMA architecture systems. Examples of commercial shared memory systems. Distributed memory systems: MPP model, connecting networks, the role of the processor in communication and transmission - the first and second generation, the development of systems with distributed memory systems: example of Intel and IBM processors. MPP systems on the TOP500 list. Clusters: definition and properties. Network connecting clusters: Topology "fat tree" network Infiniband. Beowulf Clusters. High-performance clusters. Physical construction of the clusters: rack and blade systems. Examples of clusters. MPP clusters in Top 500 list. Clusters with high reliability. Factors constituting high reliability of the clusters: Redundant nodes, access to shared resources, mechanisms for controlling the operation of nodes. Heterogeneous computer systems - conventional CPU processors support by GPUs.

Laboratory

A detailed set of laboratory exercises:

AS/400 – Communication and data access on the AS/400 system
Sparc - Sparc processor low-level programming
CUDA - parallel programming of graphics processing units (GPUs) in C / C ++ with corresponding extensions
PVM - Parallel Programming with dynamic allocation of tasks using messages passing architecture using the parallel virtual machine
Mosix – Use of a cluster of workstations for parallel computing and balance load
JavaSpaces – Parallel programming using shared and distributed memory in Java

References:
A.S. Tanenbaum, Structured Computer Organization

Number of ECTS credits: 3
Course attendants are supposed to have general knowledge about the design of combinational and sequential digital circuits. They must also have practical skills in assembling such systems. It is assumed that students have passed the following subjects: Theory of Logic Circuits, Digital Circuits.

Course objectives:

The main objective is to provide students with basic knowledge about the design of digital systems using hardware description language VHDL. During the course students should acquire basic knowledge about VHDL data structures and language constructs used for modeling combinational and sequential digital circuits. They should also develop skills in modeling in VHDL simple digital systems as well as the ability to simulate, debug and synthesize models written in that language.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/0/30

Syllabus description:

Lectures

The main objective of the course is to familiarize students with the basic aspects of designing digital circuits using VHDL. The lecture includes - among others - data structures and language constructs that are useful in modeling for synthesis and verification of digital circuit design. During the course students acquire the basic knowledge of ASIC and FPGA circuits as well as the VLSI design methodology.

1. Basic information about ASIC and FPGA circuits.
2. Introduction to VLSI design methodology.
3. VHDL – overview and application field.
4. VHDL language and syntax: general language properties, identifiers, naming convention; structural elements; data types and operators; concurrent and sequential statements; subprograms; RTL-style; behavioral, dataflow and structural modeling.
5. Simulation: sequence of compilation; simulation flow; process execution; delay models
6. Introduction to design verification – writing simple testbenches
7. Synthesis: stages of the synthesis process, VHDL constructs used in the modeling for the synthesis, best practices for modeling for the synthesis.
8. Examples illustrating how to design VHDL models of typical digital components (e.g. multiplexers, coders/decoders, counters, shift registers, FSMs, etc.).

Laboratory

During laboratory classes students design models of typical components of digital systems. Then they simulate the behavior of the models to verify the correctness of their operation. In the next step, students carry out the synthesis of the models and their implementation in an FPGA device located on a demo board.
Undergraduate obligatory courses - subject syllabi

1. Getting familiar with design tools and the demo board.
2. Multiplexers, coders / decoders, displaying characters on a 7-segment display.
3. Combinational circuits that perform binary-to-decimal (decimal-to-binary) number conversion and binary-coded-decimal (BCD) addition.
4. Latches, Flip-flops, and Registers.
5. Counters.
6. Clocks and Timers.
7. Adders, Subtractors, and Multipliers.
8. Finite State Machines.
9. Memory Blocks.

References:


Number of ECTS credits: 3
Subject: Design for manufacture  
Code: CEIE_S1_76

Level of studies: BSc  
Semester(s): 7

Teacher: prof. Zbigniew Rymarski

Pre-requisite qualification:
It has been assumed that student before attending this course have basic knowledge in designing analogue and digital microprocessor based devices.

Course objectives:
The main goal of the lectures is presenting to students the problems of the electronic devices designing taking in care the manufacturing requirements (the Design for Manufacture methodology) and appointing how to design the device to make possible to manufacture it in technologies granting the high yield with the reasonable cost simultaneously filling the assigned quality requirements.

The aim of the laboratory project is to make students familiar with creating the device documentation in the EDA environment (it was chosen Altium Designer) – the device scheme (CAE software), the printed circuit board design (CAD software) and files controlling the photoplotter (RS274X standard) and the numerically controlled drill (CAM software). During laboratory students design a simple microprocessor system that can be mounted in the standard 3U rack case using the proper design rules (concerning EMC, the tracks width, clearance and other design rules) and manufacturing rules of the chosen PCB manufacturers.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/15

Syllabus description:

Lecture
1. Introduction to the Design for Manufacture methodology
2. Design for Quality methodology, the Six Sigma program
3. The materials in the electronic devices manufacturing
   3.1. Types of the PCB, their substrates, the solder mask and finishes
   3.2. The controlled impedance PCB
   3.3. The eutectic PbSn solder and lead free solder, RoHS requirements
   3.4. The solder paste
   3.5. Conducting adhesives
   3.6. Fiducial marks
Undergraduate obligatory courses - subject syllabi

3.7. The influence of the IC case type on the manufacturing technology

4. Technologies of electronic manufacturing
   4.1. PCB manufacturing technology
   4.2. Testing of the bare PCB
   4.3. The solder paste printing on the PCB
   4.3. Sticking SMDs to the PCB
   4.4. How to avoid damaging the electronic devices by the ESD
   4.5. Pick and place machines
   4.6. Wave soldering and selective soldering
   4.7. Reflow soldering
   4.8. Typical soldering problems and defects
   4.9. PCB cleaning
   4.10. Conducting adhesives applications
   4.11. Functional testing and inspection of the mounted PCB
   4.12. Manufacturing Defects Analysis
   4.13. PCB's rework

5. Parameters of the basic passive components

6. Electric and mechanical components in the electronic manufacturing

7. PCB design methodology filling EMC requirements

8. Methodology of using CAE/CAD/CAM software in the design documentation creating and the numerically controlled devices for PCB manufacture programming.

Laboratory

Student creates the scheme and the 2 layer PCB layout design (in Altium Designer) of the simple microprocessor system that can be mounted in the standard 3U rack case using the proper design rules (concerning EMC, the tracks width and clearance standards) and manufacturing rules of the chosen PCB manufacturers. He generates the files for programming the photoplotter (RS274X files) and the numerically controlled drill. Teacher begins each of the laboratory classes with the presentation of some basic actions in the EDA software. Teacher appoints the set of components that should be used in the design (all the necessary manuals of them are enclosed in the instruction). The PCB design (3U-160 standard) should fill requirements placed in the instruction. Student should finally discuss his design that has to fill the formal demands from the instruction. The grade of the project depends on EMC of the PCB and its overall quality.

References:

Number of ECTS credits: 3
Subject: Social and professional challenges of computer science  Code: CEIE_S1_77

Pre-requisite qualification:
none

Course objectives:
The aim of the course is to provide basic knowledge in the following areas:

1. Professional and Ethical Responsibility
2. Ethical Codes and Codes of Conduct
3. Computer Systems Risks and Responsibilities
5. Fundamentals of Privacy

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/0

Syllabus description:

Lecture topics:

1. Professional and Ethical Responsibility
2. Ethical Codes and Codes of Conduct
3. Computer Systems Risks and Responsibilities
5. Fundamentals of Privacy

References:

2. ACM Code of Ethics and Professional Conduct: http://www.acm.org/about/code-of-ethics

Number of ECTS credits: 2
Undergraduate obligatory courses - subject syllabi

Subject: Software engineering  
Code: CEIE_S1_78

Level of studies: BSc

Teacher: dr. Przemysław Szmal

Pre-requisite qualification:
Fundamentals of Computer Programming, Computer Programming

Course objectives:
The course is aimed at a presentation of selected topics falling within the scope of Software Engineering with particular attention to those that relate to the life cycle of limited size software units. Among other things, students become generally familiar with the design issues of information systems using modern methodologies and CASE tools.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
30/0/0/0/30

Syllabus description:

Lectures

Laboratory (List of laboratory classes):
I. Software effectiveness estimation and improvement.

II. Software development stages:
1. Requirements modeling. Introduction to project methodics. Introduction to a selected CASE tool. Elaboration of preliminary requirements specification.
3. Use-case model. Use-case workshop. Use-cases against requirements matching.
References:


Number of ECTS credits: 4
CEIE - UNDERGRADUATE PROGRAMME

OPTIONAL COURSES – SUBJECT SYLLABI

*The minimum number of students required: 12 persons*
# Undergraduate optional courses - subject syllabi

List of the courses offered in 2013/2014

### 6th semester of CEIE (Makrokierunek) BSc programme

<table>
<thead>
<tr>
<th>Course</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Industrial measurements</td>
<td>CEIE_S1_81</td>
</tr>
<tr>
<td>Design &amp; Configuration of LAN infrastructure</td>
<td>CEIE_S1_83</td>
</tr>
<tr>
<td>Industrial Process Visualization</td>
<td>CEIE_S1_84</td>
</tr>
<tr>
<td>Constraint Logic Programming</td>
<td>CEIE_S1_88</td>
</tr>
<tr>
<td>Object oriented programming</td>
<td>CEIE_S1_92</td>
</tr>
<tr>
<td>Reliability and Intrinsic Safety</td>
<td>CEIE_S1_93</td>
</tr>
<tr>
<td>802.11 Wireless local area networks</td>
<td>CEIE_S1_95</td>
</tr>
<tr>
<td>Computer Networks II</td>
<td>CEIE_S1_96</td>
</tr>
<tr>
<td>Fiber Optics</td>
<td>CEIE_S1_97</td>
</tr>
<tr>
<td>Automotive Electronics</td>
<td>CEIE_S1_98</td>
</tr>
<tr>
<td>STM32 family ARM microcontrollers programming</td>
<td>CEIE_S1_99</td>
</tr>
<tr>
<td>Radio frequency identification systems</td>
<td>CEIE_S1_100</td>
</tr>
<tr>
<td>Digital System Design in Verilog HDL</td>
<td>CEIE_S1_101</td>
</tr>
<tr>
<td>Program-based digital control systems</td>
<td>CEIE_S1_102</td>
</tr>
<tr>
<td>GPU programming and architecture</td>
<td>CEIE_S1_103</td>
</tr>
</tbody>
</table>

### 7th semester of CEIE (Makrokierunek) BSc programme

<table>
<thead>
<tr>
<th>Course</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Face recognition and biometric systems</td>
<td>CEIE_S1_94</td>
</tr>
<tr>
<td>Industrial Control Systems Design</td>
<td>CEIE_S1_90</td>
</tr>
<tr>
<td>Linux – advanced programming</td>
<td>CEIE_S1_91</td>
</tr>
<tr>
<td>Electromechanical devices</td>
<td>CEIE_S1_89</td>
</tr>
<tr>
<td>XML technologies</td>
<td>CEIE_S1_87</td>
</tr>
<tr>
<td>Programming of mobile devices</td>
<td>CEIE_S1_86</td>
</tr>
<tr>
<td>Programming LEGO Mindstorms NTX robots</td>
<td>CEIE_S1_85</td>
</tr>
<tr>
<td>Robot vision</td>
<td>CEIE_S1_82</td>
</tr>
<tr>
<td>Computer controlled systems</td>
<td>CEIE_S1_80</td>
</tr>
</tbody>
</table>
Subject: Computer controlled systems  Code: CEIE_S1_80

Level of studies: BSc  Semester(s): 7

Teacher: dr. Ryszard Jakuszewski

Pre-requisite qualification:
Computer Graphics, Computer Programming, Computer Networks, Operating Systems, Measurement Systems, Databases. It is assumed that the student knows the basics of computer science. This lecture is suitable for students of all technical branches, which are interested in easy-to-use industrial automation software.

Course objectives:
This course is designed primarily for students wanting to create advanced control and process monitoring systems. The students should obtain knowledge of theoretical fundamentals and of practical methods used in modern SCADA systems and industrial automation software.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
15/0/0/0/30

Syllabus description:

Lecture
The world's leading industrial automation software solutions, providing process visualization, data acquisition and supervisory control of plant floor operations are discussed and trained during laboratory classes. This solutions give students the power to precisely monitor and control every aspect of manufacturing industry processes, as well as equipment and resources, resulting in faster response to production issues, less waste, improved quality, faster time-to-market with new products and increased profitability. The specialists who have acquired skills in SCADA systems are looked for in job market all over the world. The course teaches basic SCADA and HMI topics like: graphic design, data archiving, process database management, driver configuration, reporting, alarm strategies and security. The course is intended to provide the student a base level of proficiency using some of the American software solutions and more advanced features. VBA scripting is covered primarily as a tool for automating tasks for the operator. The student will also become familiar with some of the tools and concepts available for optimizing and troubleshooting such software.

Basic topics:
- Introduction to Computer Graphics
- Process Database - Development and Management
- Advanced Graphical Objects - Signal Generators
- Picture System Management
- Trending and Archiving of Data
Undergraduate optional courses - subject syllabi

- Industrial Databases (Proficy Historian)
- Implementing Shortcut Keys
- Tag Group Technology
- Scheduling Tasks
- Multilevel Security System
- Communication with PLC Controllers
- Alarming – Defining, Acknowledgement, Viewing and Printing
- Network Solutions – Controlling In Computer Networks
- Visual Basic for Applications in Industry Environment
- Configuring SCDA Systems

More advanced topics:

- Troubleshooting of SCADA Systems
- Windows and SCADA Security
- Recipes In Beer Production
- VisiconX Objects
- Alarm and Event Archiving In Relational Databases
- Electronic Signatures
- Using ActiveX
- Understanding Database Dynamos
- OPC Servers and Clients In SCADA systems
- Introduction to ADO DB and ODBC
- Using SQL Database Tags
- Reporting
- LAN and Auto-Failover Redundancy
- Using Terminal Servers

Laboratory

1. Animation of graphical objects; Development and management of industrial databases.
2. Signal generators; Tools and methods of picture development.
3. Presenting archived data on charts.
4. Shortcut keys and Tag group technology.
6. Communication with PLC Controllers, ADO DB and ODBC.

References:

1. Technical documentations for SCADA systems.
Number of ECTS credits: 2

**Subject:** Industrial measurements

**Code:** CEIE_S1_81

**Level of studies:** BSc

**Teacher:** prof. Stanisław Waluś

**Pre-requisite qualification:**

Physics, Mathematics, Fundamentals Metrology, Measurement Systems

**Course objectives:**

To acquaint the students with industrial measurements on the base of chosen values (flowrate, level, pH, conductivity), expressing of metrological properties, uncertainty calculations and using buses (for example - Profibus) for communication and data processing.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

15/0/0/0/15

**Syllabus description:**

**Lecture**

1. Introduction to industrial measurements. Evaluation of measuring instruments for steel works process – as an example of typical difficulties in measuring and controlling industrial process.

2. Practical advices and recommendations for one who can control a process.


4. The mathematical modelling of flowmeter sensors (role of the sensor in flow measurement, classification of flowmeter sensors, full-bore and sampling flowmeters, mathematical models of flowmeter primary device for various sensors (point, surface, segment, whole flow area).

5. Measuring instruments selection and evaluation on the example of flowmeter selection procedure.

6. Level measurement: classification of level measurement techniques, mechanical based level measurement, ultrasonic level sensors, nucleonic in level measurement.

7. Ionselective (ISE) measurements, pH measurements and conductivity measurements (concentration measures; dissociation; membrane potential; Nernst equation; potentiometric measuring circuit; ISE electrode; reference electrode; troubleshooting checklist; limit of detection; interference ions; Nikolsky-Eisnemann equation; Measurement methods: direct potentiometry, know addition methods, flow injection analysis, titration; industrial and medical applications).
Undergraduate optional courses - subject syllabi

8. Profibus overview (the main advantages and disadvantages of fieldbus system, the Profibus family applicable at all levels of automation, Profibus protocol, operation including device addressing, station types and network configuration, Profibus - designed for the hazardous environments).

Laboratory

1. Level measurements: Students will familiarize with level measuring techniques. There are two level transducers on the stand: FMD78 (the differential pressure transducer which can be programmed to level measurement mode) and FMP40 (the guided level radar). During laboratory students learn how to configure industrial level transducers. Students perform also different calibration procedures, check advantages and disadvantages of both level measurements methods.

2. Open channel flow measurements: Exercise is made on the open channel flow measurement laboratory stand. On this stand ultrasonic level meter is investigated. Students will learn about flow measurement with help of weirs and methods of communication and programming smart sensor.

3. Conductometry: Students learn about conductivity measurements. The ABB 4620 Industrial Conductivity Transmitter is used during laboratory. Students make some measurements for conductivity standards and samples and perform analysis of measurements and uncertainty of measurements.

4. Profibus: Students task is to build up the software in LabVIEW via PROFIBUS RS-485 for: enumerating measured value, execute several measurements on line, estimating the response time of measuring device, warning banner after crossing alarm value of methane concentration in air, growth of concentration of carbon dioxide, fall of oxygen concentration, etc.

5. HART: During laboratory students get acquaint with fundamental features of HART technology. Students use example of industrial HART intelligent transmitter to learn how to configure transmitter with the assistance of HART protocol and dedicated configuration software. Students learn how to create commissioning documentation - configuration control – element of meeting ISO9000 requirements.

6. Ion-selective measurements: Students get acquaint with ionselective measurement techniques: direct potentiometry, know addition methods and automatic determination of ionselective electrode's parameters. Students using laboratory Orion 930 Ionanalizer learn how in industry automatic laboratory measurements can be performed.

References:


Number of ECTS credits: 2
Subject: Robot vision  
Code: CEIE_S1_82

Level of studies: BSc  
Semester(s): 7

Teacher: prof. Henryk Palus

Pre-requisite qualification:
The subject complements a basic knowledge of image processing acquired by students in the subject of Computer Graphics and Vision, and directs it to problems of automation and robotics.

Course objectives:
The course aims to familiarize students with state of the art in the field of vision systems used in automation and robotics. As a result of the subject, students should be able to both design the vision system for a particular application, as well as to construct a suitable algorithm for it, implement it and perform the necessary tests. The laboratory exercises realized as part of the subject will help achieve these goals.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

Syllabus description:

Lecture
ry systems, vision applications in automation and robotics. Universal and specialized systems. Examples of real applications of robot vision systems.

**Laboratory**

2. Size of objects.
3. Objects and holes in binary image.
4. Shape factors.
5. Introduction to colour image processing.
7. Case studies.

**References:**


**Number of ECTS credits: 2**

<table>
<thead>
<tr>
<th>Subject: Design &amp; Configuration of LAN infrastructure</th>
<th>Code: CEIE_S1_83</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 6</td>
</tr>
</tbody>
</table>

**Teacher: dr. Miroslaw Skrzewski**

**Pre-requisite qualification:**

Knowledge and basic understanding of operation of the computer networks and operating systems.

**Course objectives:**

The course deals with basic solutions of wired and wireless local area network infrastructure. Principles of structural cabling design, wireless LAN communication, protocol configuration and WAN interconnection will be presented, along with the VLAN design and configuration.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

15/0/0/0/15

**Syllabus description:**
Lectures:

The concept of LAN infrastructure. Classical Ethernet (802.3, Ethernet v2.0) segment design, transceiver, terminators, repeaters, collision domain, broadcast domain. Versions of standard (10BaseT, 10BaseF, 10Base2). Modification of basic principles of network operation – introduction of store & forward technology (hub, switch, managed switch operation). Modification of topology, problems with loops, spanning tree protocol.

Virtual networks (VLAN), VLAN connections, 802.1Q protocol, VLAN configuration. Standards Fast Ethernet (100BaseT), 1G Ethernet, 10G Ethernet, cabling standards. Cables UTP, STP, connectors, cables categories cat3, cat5, cat5e, cat6.

Wireless LAN connections, standards 802.11a/b/g/n, Bluetooth, 802.16, network organization, ad-hoc, infrastructure networks, radio network access control, connection security. Access point configuration, radio bandwidth and channel allocation.

LAN systems IP configuration, protocols rarp, arp, bootp, dhcp, apipa. LAN – WAN interconnections, access line protocols, serial protocols SLIP, PPP, PPPoE, protocol tunneling. Access router, network address translation, problems of LAN systems security and LAN access protection.

Laboratories

During lab exercises students has admin access to typical devices used in network infrastructures and familiarize with their configuration, testing tools and network protocol configuration. There are planned following lab exercises:

1. Wireless LAN channel configuration and testing
2. Testing of physical network cabling infrastructure
3. Managed switches infrastructure configuration
4. LAN IP protocol configuration and testing
5. Monitoring of LAN protocol operation
6. WAN access router configuration

References:


Number of ECTS credits: 2
Subject: Industrial Process Visualization

Level of studies: BSc

Teacher: dr. Rafał Cupek

Pre-requisite qualification:
Computer networks, Theory of computer science, Control fundamentals.

Course objectives:
The structure and methods used in industrial process visualization applied to heterogeneous industrial computer systems. Presentation of the embedded visualization tools, Supervisory Control and Data Acquisition (SCADA) Systems and Manufacturing Execution Systems (MES). Data gathering and processing in distributed visualization systems. Historian, an industrial database.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
15/0/0/0/15

Syllabus description:

Lectures
- Control panels and local visualization
- Vertical communication in distributed industrial computer systems on OPC example
- Supervisory Control and Data Acquisition (SCADA) Systems on Wonderware: ArchestraA example
- Distributed Control Systems (DCS) on ABB Frilance example
- Industrial Database – historical data visualization
- Visualisation in Manufacturing Execution Systems (MES)

Laboratories
1. Graphic Panels Programming
2. InTouch Wonderware - SCADA System
3. Industrial Application Server on ArchesrA Example
4. Vertical communication on OPC Standard Example
5. Historian an Industrial Database

References:
1. F.Iwanitz, J.Lange: OPC Fundamentals
2. Jurgen Keletti: Manufacturing Execution System

Number of ECTS credits: 2
Subject: Programming LEGO Mindstorms NTX robots

Code: CEIE_S1_85

Level of studies: BSc

Semester(s): 7

Teacher: dr. Piotr Czekalski

Pre-requisite qualification:

Computer programming, Physics

Course objectives:

The main goal is to present audience a modern robot building platform and wide range of construction and programming methods of LEGO Mindstorms NXT robots.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/0/30

Syllabus description:

Lectures:

The course contains a presentation on both constructing and programming universal robotics platform based on LEGO Mindstorms NXT. Programming in NXT-G visual language on LabVIEW-based Mindstorms EDU NXT Software environment will be presented (autonomous systems), programming in VPL and C# with means of Microsoft Robotics Development Studio (remote systems) as well, as running programs on dedicated NXT Virtual Java Machine (NXJ) will be presented. Additionaly the course covers communication ideas on both inter-robotics and robots-to-PC models. As an extra part this course presents embedded system construction of NXT Intelligent Brick considerations on popular sensors, to let the students design and implement their own interfaces, robot constructions.

Topics:

1. Why just NXT platform? From a game to the business and industry. Platform preceedors.
2. Overview of NXT Intelligent Brick construction and it's communication interfaces. Platform specification and capabilities.
3. Overview of available platform sensors and servo motors.
5. Open hardware platform Mindstorms NXT – Hardware Development Kit.
6. The most popular NXT models overview.
Undergraduate optional courses - subject syllabi

Laboratories:

The main target is to let the audience practice construction and programming of NXT base robots. All exercises are performed by students during laboratories. The laboratory is equipped with 8 complete Mindstorms NXT platforms (edu edition) each equipped with NXT Intelligent Brick, a set of genuine LEGO sensors and servo motors and about 1200 pcs. of building bricks plus suitable software and modern PC Workstations. The laboratory also contains a set of third party sensors, including object tracking cameras, tilt/accelerometer sensors, gyro sensors, compass sensors and colour recognition sensors.

Lab exercises:

2. 2-3. Programming in VPL and C# on Microsoft Robotics Studio environment - simple operations on Tribot, Spike, Roboarm popular models.
3. Programming in Java on dedicated JVM.
5. 6-8. Advanced programming – following the path seamlessly by Tribot. Running through unknown environment and avoiding obstacles.

References:


Number of ECTS credits: 2
Pre-requisite qualification:

It is assumed, that the student has elementary skills in computer programming.

Course objectives:

The aim of the course is to provide techniques for developing applications for mobile devices with restricted computing power and memory. The main focus is on design and optimization of applications and user interfaces for devices like smartphones, tablet PCs, PDAs. Within the course, students will acquire the ability to create applications for Android, Windows Phone, Symbian, iOS.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/0/30

Syllabus description:

Lectures

1. Introduction
2. Architecture of mobile devices
3. Differences between applications for desktop computers and mobile devices
4. Mobile devices based on Linux
5. The Android operating system
6. Event handling
7. Localization of programs, applets
8. Multithreaded applications
9. The Windows Phone operating system
10. Other mobile operating systems
11. Application Security

Laboratories:

Will follow topics presented during the lectures.

References:

1. Andy Wigley, Daniel Moth, Peter Foot: Mobile Development Handbook
2. Andy Wigley, Stephen Wheelwright: .NET Compact Framework
3. Ivo Salmre: Writing Mobile Code
4. Paul Yao, David Durant: .NET Compact Framework Programming with C#

Number of ECTS credits: 2
Subject: XML technologies

Code: CEIE_S1_87

Level of studies: BSc

Semester(s): 7

Teacher: dr. Dariusz Mrozek

Pre-requisite qualification:

Databases, Database Applications. It is assumed that before start of this course, students have knowledge on the relational data model, architecture of database management system, skills in SQL language and designing relational database schemas.

Course objectives:

XML is recently widely used and widely accepted standard for data exchange, due to its self-descriptive character, the ability to define the hierarchy, the ability to check a validity of data and independence from hardware and software platforms. XML Technologies Course is designed for individuals who wish to deepen their knowledge of the rich world of XML technologies – especially, database developers, database administrators, data analysts, report designers and developers of web sites.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/0/30

Syllabus description:

Lectures

1. Introduction to the world of XML technologies – creation of well-formed XML documents, rules of XML documents creation, namespaces, XML documents validation
2. Document Type Definition – creating documents describing structures of XML documents, XML data validation based on DTD
3. XML Schemas – designing XML schemas, XML data validation based on XML schema, type derivation, setting constraints, type unions, enumerations
4. DOM and SAX parsers – parsing XML documents, reading data with the use of SAX, creating document tree using DOM parser
5. XPath, XQuery query languages – formulating queries to XML documents using XPath and XQuery, defining filtering criteria, data aggregation, using functions and operators, constructors
6. XLink – defining references to local and remote resources, type of links, arcs, link databases
7. XPointer languages – pointing to fragments of XML documents, creating sequences, relative pointing, points and ranges
8. Integration of data into an XML format – mapping and transforming XML documents to XML documents, transferring and integrating relational data to XML
9. XML Spy – presentation of the XML Spy tool, creating XML schema diagrams and XML documents using XML Spy
10. Reporting from XML documents, XSL stylesheets and XSLT transformations – transforming data using XSLT expressions, generating web pages and reports using XSL and XSLT
Undergraduate optional courses - subject syllabi

11. XML in selected database systems – storing XML data in relational databases, generating XML data with the use of SQL language and user functions, XML validation in relational databases, invoking XPath/XQuery queries in SQL

12. Web services – creating and using web services, purpose of the WSDL, UDDI, SOAP documents

13. XML applications – XML in electronic data interchange, creating web portals based on XML, dedicated XML formats for data exchange

Laboratories

The aim of the laboratory is to acquaint students with XML technologies through practical exercises. The laboratory is carried out mainly using Altova's tools.

1. XML Schema
2. XSLT transformations
3. Querying data in XPath and XQuery
4. DOM and SAX parsers
5. Mapping data to XML
6. XML in selected database system

References:

2. Bill Evjen, Kent Sharkey, Thiru Thangarathinam, Michael Kay, Alessandro Vernet, Sam Ferguson: Professional XML (Programmer to Programmer), Wrox; 1 edition (April 9, 2007)

Number of ECTS credits: 2

<table>
<thead>
<tr>
<th>Subject: Constraint Logic Programming</th>
<th>Code: CEIE_S1_88</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 6</td>
</tr>
</tbody>
</table>

Teacher: dr. Szymon Ogonowski

Pre-requisite qualification:

Skills in object-oriented programming

Course objectives:

The aim of this lecture is to present Constraint Logic Programming techniques for solving Constraint Satisfaction Problems and Constraint Optimisation Problems. Applications of those techniques are demonstrated by many examples taken from real-world combinatorial problems, such as scheduling, planning or job-shop problems. The theory presented in the lecture is supported by problems solving using Java language (JaCoP library).
Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
15/0/0/0/15

Syllabus description:

Lectures

Constraint Logic Programming is a methodology having backgrounds in Operational Research and belongs to wide concept of Artificial Intelligence Methods. Its main concept is based on declarative programming – the programmer needs only to describe what computation should be performed and not how to compute it (this part is already implemented in solver). Lecture is based on a multimedia presentation (available for the course students), supported with example code testing. The course presents concepts of:

- constraint propagation, global constraints, arc-consistency,
- different domains of variables - boolean domains, finite domains, sets and real intervals,
- global constraints such as: alldifferent, count, element, cumulative, diff2, circuit, knapsack, among, regular,
- search distributions – first-fail, most-constrained, max-regret,
- tree-based search such as - branch-and-bound, depth first search, limited discrepancy search, special search techniques,
- methods of default search algorithms modification – creating own search techniques,
- modelling and solving constraint optimisation problems and soft constraints - Weighted CSP, Fuzzy/Possibilistic CSP, Probabilistic CSP, over-constrained problems,
- scheduling/planning problems,
- incorporating local search techniques into constraint programming methodology.

Details of concepts mentioned above are presented while solving examples of simple academic and more complex, real-world combinatorial problems with usage of Java language and dedicated JaCoP (Java Constraint Programming) library. Advantages of using well known and well documented language allows the students to focus on getting to know the idea behind constraint programming. During laboratory exercises students constructs Java desktop applications and incorporates in it different modules, designed to solve different combinatorial problems.

Laboratories

All laboratory exercises are focusing on different CP problems, that are solved as a separate Java modules. Designed modules are incorporated in main Java desktop application, creating compact and scalable project.
1. Design of CLP application base
2. Simple combinatorial problems modules
3. Soft constraint problem module - reified constraint
4. Scheduling/planning problem module – building bridge example
5. Search strategy module – queens problem
6. Optimisation problem module
Undergraduate optional courses - subject syllabi

References:
1. Francesca Rossi, Peter Van Beek, Toby Walsh "Handbook of constraint programming",
2. Peter van Hentenryck "Constraint-based Local Search".

Number of ECTS credits: 2

<table>
<thead>
<tr>
<th>Subject: Electromechanical devices</th>
<th>Code: CEIE_S1_89</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 7</td>
</tr>
</tbody>
</table>

Teacher: prof. Krzysztof Kluszczyński

Pre-requisite qualification:
Automatics and robotics – obligatory; Electronics and telecommunications, Informatics

Course objectives:
This curriculum develops basic skills in the field of design, measurement and application of electrical machines and modern drives. Up to date information concerning various types of machines and their control schemes is provided. Special attention is focused on applications in automatic control systems and robotics.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
15/0/0/0/30

Syllabus description:

Lectures
Undergraduate optional courses - subject syllabi


Laboratories


References:

2. Electric drives, Ion Boldea, S.A. Nasar, Taylor & Francis, 2005

Number of ECTS credits: 2

<table>
<thead>
<tr>
<th>Subject: Linux – advanced programming</th>
<th>Code: CEIE_S1_91</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s):</td>
</tr>
<tr>
<td></td>
<td>7</td>
</tr>
</tbody>
</table>

Teacher: dr. Dariusz Bismor

Pre-requisite qualification:

Computer programming, Operating systems

Course objectives:

The course has two aims. The first is to present the basics as well as more advanced techniques used in Linux kernel programming. As there are (soft) real time versions of Linux kernel, this aim agrees with real time systems teaching requirement. The second aim is to demonstrate object oriented programming techniques with application to graphical user interface programming in KDE/Qt environment. This extends the information technology knowledge of course attendants.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/0/30

Syllabus description:

The lecture discusses the following areas of Linux kernel programming: differences between user and kernel mode code, Linux kernel modular structure and design, what all modules must have; how to compile a simple module; devices in Linux system: block devices, character devices and network interfaces; character devices: numbering, file operations, talking to; memory issues: types of memory, cach-
Undergraduate optional courses - subject syllabi

es, paging; ioctls, blocking, poll and select, asynchronous operations; race conditions, semaphores, mutexes and spin locks; scheduling tasks; interrupt handling basics; block devices: differences between character and block devices, buffer cache, virtual file system, structure inode, writing simple file system, second extended file system.

The lecture discusses the following areas of KDE programming: KDE 4.x architecture overview; 3-level structure of KDE/Qt programs; signals and slots; designing applications using XML GUI framework; design patterns in application config files and setting dialogs; layout management; composing GUI with KDE Designer and Qt Designer; graphical operations basics; writing new widgets.

The lecture is based on slides displayed with multimedia projector. Students are allowed to download outlines prior to lecture. All issues mentioned above are discussed by the lecturer, with emphasize on the issues selected by students. Many issues are illustrated by working program code.

Laboratories

1. Writing a simple Linux kernel module.
2. Writing a character device driver, part 1: reading and writing from device file.
3. Writing a character device driver, part 2: more advanced operations.
4. Writing a block device driver.
5. Writing a simple KDE application.
6. Using QT Designer to compose GUI.

References:


Number of ECTS credits: 2

Subject: Industrial Control Systems Design
Code: CEIE_S1_90

Level of studies: BSc
Semester(s): 7

Teacher: dr. Zbigniew Ogonowski
Undergraduate optional courses - subject syllabi

Pre-requisite qualification:
Control fundamentals, Introduction to system dynamics, Fundamentals of signal processing, Measurement systems, Numerical methods, Optimization and decision making

Course objectives:
The course provides an overview of advanced control design methods for industrial systems. It is aimed to extend control engineering knowledge of the students onto system scientist knowledge. It is then assumed that students understand control fundamentals, dynamic modeling, signals measurement and processing as well as numerical methods and optimization issues. The course gathers these aspects and assume to use them as a tool in the control system design for industrial processes. The main aim is to teach design principles according to hierarchical structure of the industrial processes. Students should acquire skills to create realistic and practical solutions of the industrial control problems.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
15/0/0/0/30

Syllabus description:
The course consists of three parts covering the most important aspects of Industrial Control System Design. Beginning with the general hierarchical structure of control system (part one) the course proceeds with the direct control layer (part two). The most important in this part are the principles of structure choice of the control system. In part three, functionality of the upper control layer is presented including acquisition and data processing together with process optimization.

Part I
3. Functional structure of control systems: General scheme of industrial process, Direct control layer, Upper control layer, Operating (optimization) control layer, managing layer.

Part II
1. Direct control layer: Tasks, Structure of direct control layer, Leading streams, Constructive postulates for direct control layer.
Part III

1. Data acquisition and processing: Periodical data acquisition, Acquisition methods, AD/DA conversion, Acquisition file, Basic data processing, Special processing of process variables.
2. Signaling, supervision and documenting the process: Purpose, Signaling, Supervision, Documenting, Alarm system, Supervision.
4. Standardization of project description: ISO norm, Topology of the project, Graphical signs, Abbreviation system

Laboratory

1. Simple industrial process analysis and control design.
2. Continuous stirred tank reactor - Three-level hierarchical control system.
3. Sodium bicarbonate technology - Leading steam choice.
5. Drying spray-both system – recycling process with losses complement.
6. Oil refiner – optimizing control layer.

References:


Number of ECTS credits: 2

Subject: Object oriented programming  
Code: CEIE_S1_92

Level of studies: BSc

Teacher: dr. Dariusz Bismor

Pre-requisite qualification:

Computer programming

Course objectives:

The aim of the course is to introduce the modern, object-oriented program design and programming techniques. Students should learn the difference between procedural and object-oriented techniques as well as analysis and design techniques for object-oriented style and those programming languages that give support for object-oriented programming. Knowledge attained during the course should allow for
easy and fast completion of even large programming projects. The examples illustrating the lecture are based on control theory and practice.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

15/0/0/0/15

**Syllabus description:**

The lecture emphasizes, but is not limited to the following topics: introduction to object-oriented analysis, object statics, object dynamics, object relationships and interactions, class as object part, class relationships, constructing a model of a system, Unified Modeling Language (UML), design patterns, creational patterns, structural patterns, behavioral patterns.

The lecture is based on slides displayed with multimedia projector. Students are allowed to download outlines prior to each lecture. All issues mentioned above and below are discussed by the lecturer, with emphasize on the issues selected by students. Many issues are illustrated by working program code.

The detailed lecture content is given below.

1. **Introduction:** the course aim, the tools and references. Unified Modeling Language (UML): basic constructs, rules and diagrams. Use case diagrams, class diagrams, object diagrams, sequence diagrams, collaboration diagrams, state-chart diagrams, activity diagrams, component and deployment diagrams.

2. **Object-oriented analysis:** the purpose, the importance, the input and the output. Understanding the needs of clients, describing what the system will do and how the system should behave, identification of objects. Four-component model, other models. Object statics: instances, classes, attribute features, constrains. Object relationships: collections, property inheritance, subclasses. Object dynamics: describing behavior, transition networks, actions, exceptions. Object interaction: transitions, sending and receiving events. Constructing a system model: the use of the above notions to describe a system.


4. **Creational patterns:** singleton. Structural patterns: adapter, decorator, facade, composite, bridge.

5. **Structural patterns:** proxy, flyweight. Behavioral patterns: interpreter, iterator, chain of responsibility.

6. **Behavioral patterns:** mediator, template method, observer, visitor, memento, command.

7. **Behavioral patterns:** state, strategy. Other patterns and concepts: lazy and eager computation, advanced reference counting, copying on write, pimpl idiom.

Laboratory exercises aim at writing a single, complete piece of code, solving the problem selected by a tutor.

1. **Introduction to the project:** analysis and design.
2. **Finding the design patterns in the project.**
3. **Creational, structural and behavioral patterns in practice, part 1.**
Undergraduate optional courses - subject syllabi

6. Other patterns and programming idioms.

References:

1. E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns, Addison Wesley, 1995.

Number of ECTS credits: 2

---

<table>
<thead>
<tr>
<th>Subject: Reliability and Intrinsic Safety</th>
<th>Code: CEIE_S1_93</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s): 6</td>
</tr>
</tbody>
</table>

Teacher: dr. Andrzej Kozyra

Pre-requisite qualification:

Understanding of most important terms like: reliability, fault, MTBF, MTTF, failure rate, types of explosion-proof protections, national and international standards, explosion-proof apparatus marking. Ability to estimate reliability parameters of elements and systems by using modular decomposition, fault tree, Markov processes.

Course objectives:

Goal of the course is to acquaint students with reliability assessment of technical objects and systems, reliability analysis methods, constructions of explosion-proof apparatus and designing an intrinsically safe systems.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/0/15

Syllabus description:

Lectures

1. Introduction, Statistical Approach to Reliability – reliability, maintainability, failure, fail function, hazard rate, MTBF, MTTF, Wiener's equation.
Undergraduate optional courses - subject syllabi

6. Intrinsically Safe Systems 'i'- An intrinsically safe protection as the most safe protection, requirements for intrinsically safe apparatus, certification procedures, basic installation requirements.

Laboratories

2. Basic reliability structures. Students learn how to create a graph of system states and estimate reliability of basic systems. Analysis bases on Markov models.
6. Fuzzy reliability. Application of fuzzy logic for estimation of reliability of systems. Students use Matlab application to estimate reliability of systems. Demonstration of system with intrinsically safe modules. Students learn how to make system more reliable and how intrinsically safe system can be built.

References:


Number of ECTS credits: 2
Subject: Face recognition and biometric systems

Code: CEIE_S1_94

Level of studies: BSc

Semester(s): 7

Teacher: dr. Michał Kawulok

Pre-requisite qualification:

Programming in C++.

Course objectives:

The course aims to familiarize students with the basic technologies used in biometrics and prepare them for designing and developing advanced biometric systems. Students will be given the experience gained in the course of operating a system of automatic face recognition. The lecture will discuss the fundamental problems associated with biometric systems with particular emphasis on face detection and recognition. The laboratories involve practical implementation of the various stages of face recognition by applying the methods known in the field. Students will create or modify parts of the existing system, which simplifies the testing and evaluation of the solutions.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/0/30

Syllabus description:

Lectures

1. Introduction to biometrics and identification systems.
2. Face detection.
4. Feature extraction using the Eigenfaces method.
5. Linear discriminant analysis (LDA) and Fisherface feature extraction method.
6. Possibilities of improving the Eigenfaces method.
7. Methods based on local features analysis (Gabor Wavelets).
8. The use of classifiers for face recognition, support vectors machines (SVM).

Laboratories

1. Presentation of the environment, basic image processing tasks.
2. Normalization of frontal facial images.
3. PCA training and generation of the eigenfaces.
4. Extraction of features using the Eigenfaces method.
5. Application of SVM to face verification.
Undergraduate optional courses - subject syllabi

References:


Number of ECTS credits: 2

Subject: 802.11 WIRELESS LOCAL AREA NETWORKS  
Code: CEIE_S1_95

Level of studies: BSc  
Semester(s): 6

Teacher: dr. Dariusz Wójcik

Pre-requisite qualification:
principles of electric circuits, computer and digital systems fundamentals, fundamentals of information systems security, fundamentals of access control systems; fundamentals of antenna theory.

Course objectives:
The aim of this course is to familiarize students with all aspects of 802.11 wireless local area networks: overview of the technology and architecture of WLANs, explanation of services and advanced features. The course gives knowledge needed to design, deploy, manage, and troubleshoot wireless local area networks (WLANs).

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
15/0/0/0/15

Syllabus description:

Lecture:

1. Fundamentals of wireless local area networks, usage of non-licensed ISM bands, the legal regulations
2. The role of IEEE standards 802.11 family of wireless networks against the reference model OSI / ISO and other standards IEEE 802
3. Medium access control, resource reservation methods, topology, ad-hoc and infrastructure networks, c
4. The physical layer, link layer protocols, the management layer protocols, upper layer protocols.
5. The organization of the network, the structure of frames.
6. Ad-hoc (IBSS) networks
7. Infrastructure networks
8. Energy management in 802.11
9. Quality of service in 802.11.
10. Roaming.
12. Network configuration.
13. Design of 802.11 networks

**Laboratory:**

1. Configuration of 802.11 networks
2. Ad-hoc networks
3. Wireless gateway configuration
4. Access-point management
5. Network management - network scanning, access control and data encryption
   802.11 network design

**References:**


**Number of ECTS credits:** 2

---

**Subject: Computer Networks II**

**Code: CEIE_S1_96**

**Level of studies:** BSc

**Teacher:** dr. Mirosław Skrzewski

**Pre-requisite qualification:**

introduction to computer science, basics of digital data transmission systems

**Course objectives:**

The aim of the course is to familiarize students with the basic principles of the communication protocols design and the construction and operation of computer networks. The solutions of the data link layer, network and transport layer protocols will be discussed in details, as well as the basic functions of higher layer protocols of the ISO model. Finally, the basic rules for the implementation of core services of Internet network, like dns., email, web will be presented.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

15/0/0/0/15

**Syllabus description:**

**Lecture:**
Undergraduate optional courses - subject syllabi

Problems of information exchange between computers, the concept of the transmission channel, the communication protocol, algorithms of reliable transmission, the form of information processing, networks services. The logical architecture of computer networks, the ISO OSI reference model, the division of tasks into layers, layer functions, communication (interface) between layers, layer data units, layer services, service delivery models, addressing.

Physical layer functions, tasks of link layer, characters oriented protocols, bit oriented protocols, the methods of obtaining reliable transmission in the presence of interference, modem protocols. Local area network channels, the problem of media access, classification of media access algorithms, CSMA protocols, token based protocols, media allocation protocols. LAN infrastructure, network management, VLAN networks.

Network of transmission channels, modes of operation, the network topology, the tasks of the network layer, network addressing, route selection algorithms, mechanisms of adaptation to changes in topology and load on the network. Protocols Distance Vector, Link State, hierarchical routing, examples of protocols (RIP, OSPF, BGP), cooperation of networks of different organization of transmission.

Transport of information, organization of transmission, addressing, synchronization of network endpoints. Connection oriented, connectionless communication, quality of service (QoS). The problem of interruptions in transmission, the tasks of the session layer, session state registration, recovery algorithms. Processing the form and structure of the information, the notation ASN-1, the problem of information security.

Examples of the wide area network architectures - XNS, Internet (TCP / IP). Structure and function of protocols, network addressing, auxiliary protocols (DNS, ARP, ICMP), transport layer algorithms. LAN architecture - the NetBIOS protocol, the principles of addressing, the SMB protocol.

Network operating systems, client-server systems, peer-to-peer networks, addressing access to services, safety. Unix communication services, rpc, ftp, telnet, smtp, http. Windows, NetBEUI, network environment, mapping drives, shared network resources (folders, printers), the system of access rights.

Laboratory:

Laboratory exercises presents the basic issues related to communication in computer networks, the rules of their configuration and network performance monitoring tools available at network operating systems. In various exercises, students will configure and test the operation of the different network protocols and services using them:

- Principles of communication in IP networks
- Application Programming Interface of TCP/IP stack
- Translation services of system names
- WAN routing protocols
- Local network infrastructure
- Windows Network Environment
Undergraduate optional courses - subject syllabi

References:
1. Tanenbaum, Computer Networks,

Number of ECTS credits: 2

Subject: Fiber Optics  
Code: CEIE_S1_97

Level of studies: BSc  
Semester(s): 6

Teacher: dr. Grzegorz Wieczorek

Pre-requisite qualification:
Course attendants are supposed to have general knowledge concerning basic electronic components and analog circuits. It is assumed that students passed the following courses: Physics, Introduction to Electronics.

Course objectives:
The course aims objectives include having the students got acquainted structure, properties and parameters of fiber optics and basic optoelectronic components applied in fiber optic transmission systems. Discussed in the lecture are fiber optic applications in telecommunication, configurations and types of connections, transmission and multiplexing techniques.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
15/0/0/0/15

Syllabus description:

Lecture:

mixers, modes separators. Removable and permanent types of connections. Optical fiber cables - production, types and parameters.

Laboratory:
1. Determination of numerical aperture and acceptance angle of the optical fibers.
2. Coupling characteristics of the optical fibers.
4. Optical Time Domain Reflectometer OTDR.
5. Optical fiber arc fusion splicing. Preparation and cutting of fiber.

References:

Number of ECTS credits: 2

<table>
<thead>
<tr>
<th>Subject: Automotive Electronics</th>
<th>Code: CEIE_S1_98</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of studies: BSc</td>
<td>Semester(s):</td>
</tr>
</tbody>
</table>

Teacher: prof. Zdzisław Filus

Pre-requisite qualification:

It is assumed that the course attendants know physical principles of operation of the internal combustion engine and have basic knowledge of circuit theory, electronic circuits, measurements and fundamentals of control and regulation.
Course objectives:

The objective of the lecture is to introduce the most important applications of electronic circuits to passenger vehicles. Particular attention is given to measurement of various physical quantities, connected with the movement of the car or with the operation of its individual blocks and to multiplexed wiring systems (automotive buses). Principles of control over various functions of the car are also discussed. The lecture should enable the students to understand peculiar features of the operation of electronic circuits in measurement systems, especially those designed for mechanical quantities, and in systems for control of mechanic and electric actuators.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

30/0/0/0/0

Syllabus description:


Microprocessor control systems in cars. Description of main electronically controlled vehicle systems. Basic features of microcontrollers for automotive applications.


Anti-lock braking systems (ABS) and Traction control systems (TCS). Principle of operation of ABS systems. Typical configuration of ABS systems. Principle of operation of TCS systems.


Suspension and steering control. Electronically controlled suspension. Steering-wheel assist. Four-wheel steering systems.


References:

Number of ECTS credits: 2

Subject: STM32 family ARM microcontrollers programming  Code: CEIE_S1_99

Level of studies: BSc  Semester(s): 6

Teacher: dr. Damian Grzechca, dr. Tomasz Golonek

Pre-requisite qualification:
A student has a basic knowledge in C/C++ programming and fundamental concepts of microprocessor systems.

Course objectives:
The aim of the course is to introduce students to STM32 family ARM microcontrollers programming with the use of JTAG interface and utilization of the standard library functions which support the microcontroller peripherals.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):
0/0/0/0/30

Syllabus description:

Laboratory:
2. Configuration of clock signals for the microcontroller. GPIO port communication.
3. Control of alphanumeric LCD display.
4. UART interface communication, interrupts handling.
5. Configuration and use of counters (“Timer”).
6. Use of A/D converter.
7. Use of D/A converter.
8. DMA data transfer.
9. SPI interface communication.
10. I2C interface communication.
11. 1-wire network communication standard.
12. SD memory card operations.

References:
1. Reference manual RM0008:
   AL/CD00171190.pdf

Number of ECTS credits: 2

**Subject: Radio frequency identification systems**  
**Code:** CEIE_S1_100

**Level of studies:** BSc  
**Semester(s):** 6

**Teacher:** dr. Tomasz Topa

**Pre-requisite qualification:**

principles of electric circuits, computer and digital systems fundamentals, fundamentals of information systems security, fundamentals of access control systems; an understanding of antenna theory and design would be useful but is not necessary, as the basics will be covered on the course.

**Course objectives:**

The aim of this course is to familiarize students with all aspects of technology used in modern RFID systems, including the near and far field electromagnetic coupling concept for detecting objects. The physics, design, data structures and control mechanisms for RFID systems are covered. Students will also be familiarized with associated standards, emerging business process models, applications, and social issues arising from the use of the RFID.

**Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):**

30/0/0/0/0

**Syllabus description:**

**Lecture:**

1. RFID background: history, architecture, applications, shareholders, social implications and privacy
2. Barcode and barcode systems: EAN-8 and EAN-13 barcode, UPC-7 and UPC-12 barcode, 2D and 3D barcodes, the pros and cons of barcode systems
3. Tag layer: tag classification, architecture, tag placement - shadowing risk and mitigation, antenna configurations
4. Reader layer: reader classification, architecture, deployment requirements, interrogation zones, antenna configurations, sessions, middleware, SmartLabel printers
5. Media interface layer: frequency bands, read range, modulation, encoding, communication protocols, data rates, reader and tag collisions, anti-collision protocols, tag-to-tag and reader-to-reader interference, tag travel speed
6. RFID security and privacy: interactions with wireless LANs, chip clones, cryptography, symmetric ciphers, asymmetric ciphers, elliptic curve ciphers, authentication protocols, weaknesses in the encryption algorithm, weaknesses in key management, EPC trust services
Undergraduate optional courses - subject syllabi

7. RFID standards: regulations, policies and guidelines: EPCglobal, ISO/IEC item management, contactless Smart Cards, animal identification, FCC rules for ISM band, identity standards and guidelines for securing RFID systems

8. Case studies: patient tracking, blood services banks, boost asset awareness, asset-tracking platform, RFID pharmaceuticals seek system, library management system, public transport, ticketing, access control systems, animal identifications, electronic immobilization, container identification, industrial automation, sport events

References:


Number of ECTS credits: 2

Subject: Digital System Design in Verilog HDL  
Code: CEIE_S1_101

Level of studies: BSc  
Semester(s): 6

Teacher: dr. Robert Czerwiński

Pre-requisite qualification:

It is assumed that the student is trained in the basics of digital technology, basics of digital circuit design and the computer programming.

Course objectives:

The aim of the course is to familiarize students with the digital systems designing using Verilog hardware description language.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

0/0/0/0/30

Syllabus description:

The purpose of the laboratory classes is to familiarize students with the Verilog hardware description language and the designing process using complex programmable logic devices. The first part of the
laboratory exercises will be based on writing simple models, synthesizing, implementing and testing. In the second part students are going to implement the project of complex digital circuit.

1. Getting to know the tools for simulation, synthesis and implementation. Writing synthesizable models of combinatorial: translators, multiplexers, decoders, demultiplexers, arithmetic units.
2. Simulation and testing of combinational circuits.
3. Writing synthesizable models of simple sequential circuits: asynchronous flip-flops, synchronization systems with a latch mechanism, edge-triggered systems.
4. Simulation and testing of sequential systems.
5. Writing synthesizable models of complex sequential devices: counters, registers, finite state machines.
7. Implementation of complex project.

References:


Number of ECTS credits: 2

Subject: Program-based digital control systems  
Code: CEIE_S1_102

Level of studies: BSc  
Semester(s): 6

Teacher: dr. Robert Czerwiński, dr. Mirosław Chmiel

Pre-requisite qualification:

It is assumed that the student has a basic knowledge of microprocessors (including programmable logic controllers), and has the ability to program in C/C++.

Course objectives:

The aim of the course is to acquaint students with the methods of implementation of digital control systems. The course focuses on the subject of software solutions, with particular emphasis on systems built using microcontrollers and programmable logic controllers.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

0/0/0/0/30
Syllabus description:

1. Microprocessors: microprocessor functional blocks, work cycles, pipelining, addressing modes, the system interrupts, etc.
2. Microcontrollers: characteristics, peripheral blocks.
3. Programmable controllers S7-200 (S7-300/400): characteristics, specifications, programming languages.
4. Designing algorithms, decomposition of complex problems into smaller, programming techniques.
5. Blocks of data, local and global variables, declarations.
6. Functions, passing parameters to / from functions.
7. Cyclic operation, interrupts.
8. I/O support, peripheral block support.
9. Programs executions: monitoring of the program, the location of faults, monitoring variables.

References:


Number of ECTS credits: 2

Subject: GPU programming and architecture  Code: CEIE_S1_

Level of studies: BSc  Semester(s): 6

Teacher: dr. Tomasz Topa

Pre-requisite qualification:

C/C++ programming basics, familiarity with CPU architecture and multi-threaded programming; an understanding of computer graphics algorithms would be useful but is not necessary, as the basics will be covered on the course.

Course objectives:

The aim of this course is to introduce the programming techniques required to develop general purpose software applications for graphics processor units (GPUs). The ATI/AMD and/or NVIDIA GPU hardware allows achieving computing power unavailable for traditional central processing units (CPUs). Using CUDA and OpenCL framework, the course will focus on the solution to common problems encountered while developing software applications on the GPU. This will include an introduction to the programming techniques required to take advantage of the architecture, as well as more advanced optimization methodologies needed to get maximum performance out of the computing platform.

Teaching modes and hours (Lecture/Seminar/Class/Project/Laboratory):

15/0/0/0/15
Syllabus description:

Lecture:

1. Overview of computer animation and graphics systems: video display devices, output primitives, three-dimensional geometric modeling and transformations, illumination and surface-rendering methods, the viewing pipeline
2. Introduction to GPU programming: GPU architecture, overview of parallelism model, arithmetic accuracy and rounding
3. GPU hardware: CUDA Core, Radeon Core, special function unit, load/store unit, texture unit, dispatch unit, streaming multiprocessor, raster operations processor, thread sequencer, thread/graphics processing cluster, streaming processor array
4. GPU programming model: threads and thread hierarchy, thread assignment and scheduling, synchronisation and transparent scalability, stream computing, host and device interactions
5. Execution model: warps, scheduling and divergence
6. Device memory: global and shared memory, local memory, constant and texture memory, registers, memory hierarchy, memory latency
7. Performance optimizations: instruction performance, memory access patterns, global memory coalescence, local memory bank conflicts, optimization strategies, data prefetching, thread granularity
8. CUDA: tools and libraries: detailed description of CUDA API, compilation using nvcc, debugging, profiling, basic libraries, project assignment
9. OpenCL: introduction to OpenCL, differences comparing to CUDA, exploiting OpenCL for hardware not accessible from CUDA
10. Case studies: acceleration of image and video compression, MRI reconstruction, molecular visualization and analysis, computational electrodynamics, quantum chemistry, bioinformatics, signal processing, financial modeling, neural networks

Laboratory exercises:

1. GPU programming environment – installation, configuration, running and deploying applications
2. Data transfer and data caching
3. Memory access pattern
4. Launching kernels – thread cooperation, thread and device synchronization
5. Optimizing kernel code
6. Atomics
7. Concurrent transfer and execution
8. Mixing CUDA/OpenCL and rendering
9. Debugging and profiling kernel code

References:

Undergraduate optional courses - subject syllabi


**Number of ECTS credits: 2**