

# Foundations of Digital and Microprocessor Engineering - course description

## General information

|                    |                                                                                    |
|--------------------|------------------------------------------------------------------------------------|
| Course name        | Foundations of Digital and Microprocessor Engineering                              |
| Course ID          | 06.5-WE-AiRP-PTCiM                                                                 |
| Faculty            | <a href="#">Faculty of Computer Science, Electrical Engineering and Automatics</a> |
| Field of study     | Automatic Control and Robotics / Industrial Control                                |
| Education profile  | academic                                                                           |
| Level of studies   | First-cycle studies leading to Engineer's degree                                   |
| Beginning semester | winter term 2016/2017                                                              |

## Course information

|                     |                                                                                           |
|---------------------|-------------------------------------------------------------------------------------------|
| Semester            | 3                                                                                         |
| ECTS credits to win | 6                                                                                         |
| Course type         | obligatory                                                                                |
| Teaching language   | polish                                                                                    |
| Author of syllabus  | <ul style="list-style-type: none"><li>dr hab. inż. Krzysztof Sozański, prof. UZ</li></ul> |

## Classes forms

| The class form | Hours per semester (full-time) | Hours per week (full-time) | Hours per semester (part-time) | Hours per week (part-time) | Form of assignment |
|----------------|--------------------------------|----------------------------|--------------------------------|----------------------------|--------------------|
| Laboratory     | 30                             | 2                          | 18                             | 1,2                        | Credit with grade  |
| Lecture        | 30                             | 2                          | 18                             | 1,2                        | Credit with grade  |

## Aim of the course

- zapoznanie studentów z podstawowymi układami cyfrowymi realizowanymi za pomocą układów małej skali integracji i programowalnych układów logicznych,
- ukształtowanie umiejętności w zakresie projektowania i minimalizacji układów cyfrowych,
- zapoznanie studentów z podstawami układów mikroprocesorowych,
- ukształtowanie umiejętności w zakresie programowania mikroprocesorów.

## Prerequisites

Podstawy systemów dyskretnych, Programowanie z elementami algorytmiki, Podstawy elektroniki, Architektura systemów komputerowych

## Scope

Podstawy techniki cyfrowej. Podstawowe bramki logiczne - parametry techniczne. Klasy układów scalonych. Skala integracji. Systemy i kody liczbowe. Algebra Boole'a. Funkcja logiczne. Systemy funkcjonalnie pełne. Sposoby reprezentacji funkcji logicznej.

Układy kombinacyjne. Analiza i synteza układu kombinacyjnego. Minimalizacja funkcji logicznej. Hazard w układach kombinacyjnych.

Podstawowe przerzutniki asynchroniczne i synchroniczne. Układy sekwencyjne: Moore'a, Mealy'ego. Synteza automatów synchronicznych i analiza automatów synchronicznych.

Charakterystyka układów asynchronicznych oraz porównanie z układami synchronicznymi.

Cyfrowe bloki funkcjonalne w technice MSI. Liczniki, rejestry, rejestry przesuwne. Zasady projektowania liczników asynchronicznych i synchronicznych. Projektowanie układów kombinacyjnych z wykorzystaniem: multiplexerów, dekodatorów, bramek NAND.

Formaty danych stosowane w procesorach stałoprzecinkowych i zmiennoprzecinkowych. Arytmetyka stało- i zmiennoprzecinkowa.

Układy arytmetyczne. Dodawanie, odejmowanie i komparacja liczb binarnych. Układy arytmetyczne średniej skali integracji.

Pamięci ROM, RAM, EEPROM, FLASH. Układy PLD, CPLD i FPGA. Projektowanie układów cyfrowych z wykorzystaniem układów PLD i CPLD.

Mikroprocesory. Definicje i podstawowe pojęcia, klasyfikacja mikroprocesorów. Bloki funkcjonalne mikrokomputera i zasady ich współdziałania. Architektura mikroprocesora, rola jego bloków funkcjonalnych, cykl rozkazowy mikroprocesora.

Techniki programowania, lista instrukcji procesora.

Wymiana informacji w systemie mikroprocesorowym. Organizacja i synchronizacja wymiany informacji między elementami sytemu mikroprocesorowego. Sposoby adresowania pamięci i elementów wejścia-wyjścia.

Wymiana informacji między systemem mikroprocesorowym a otoczeniem zewnętrznym. Sposoby i uwarunkowania obsługi elementów otoczenia zewnętrznego systemów mikroprocesorowych. Wymiana informacji między systemami mikroprocesorowymi. Sposoby wymiany informacji: z potwierdzeniem i bez potwierdzenia, synchronicznie i asynchronicznie, równolegle i szeregowo. Wady i zalety poszczególnych sposobów, zakres stosowania.

Mikrokomputery jednoukładowe - charakterystyka zasobów, zasady aplikacji. Środki wspomagające oprogramowanie i uruchamianie systemów mikroprocesorowych.

Mikrokontrolery rodziny AVR. Platforma Arduino.

Historia, tendencje rozwojowe i porównanie cyfrowych procesorów sygnałowych. Podstawowe cechy procesorów sygnałowych. Różnice pomiędzy cyfrowym procesorem sygnałowym a mikrokontrolerem i mikroprocesorem. Architektury procesorów sygnałowych: sprzętowy układ mnożący, architektura typu Harvard, architektury wieloszynowe, przetwarzanie potokowe, skoki z opóźnieniem, operacje równoległe, długi akumulator, układ przesuwający, bufor cyrkulacyjny. Tryby adresowania pamięci: bezpośrednie, pośrednie, natychmiastowe, cyrkulacyjne, z rewersją bitów. Układy bezpośredniego dostępu do pamięci (DMA). Układy wieloprocessorowe.

## Teaching methods

**wykład:** wykład konwencjonalny,

**laboratorium:** ćwiczenia laboratoryjne.

# Learning outcomes and methods of theirs verification

| Outcome description                                                                                                                          | Outcome symbols                                                                                         | Methods of verification                                                                                             | The class form                                                                 |
|----------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------|
| Potrafi programować w języku niskiego i wysokiego poziomu, przeprowadza analizę pracy procesora, potrafi identyfikować stan pracy procesora. | <ul style="list-style-type: none"><li>• <a href="#">K_W06</a></li><li>• <a href="#">K_U07</a></li></ul> | <ul style="list-style-type: none"><li>• an ongoing monitoring during classes</li></ul>                              | <ul style="list-style-type: none"><li>• Laboratory</li></ul>                   |
| Ma świadomość dynamicznego rozwoju mikroprocesorów i mikrokontrolerów.                                                                       | <ul style="list-style-type: none"><li>• <a href="#">K_W17</a></li><li>• <a href="#">K_K02</a></li></ul> | <ul style="list-style-type: none"><li>• an evaluation test</li></ul>                                                | <ul style="list-style-type: none"><li>• Lecture</li></ul>                      |
| Zna podstawy posługiwania się narzędziami do programowania, symulacji i uruchamiania procesorów.                                             | <ul style="list-style-type: none"><li>• <a href="#">K_W05</a></li><li>• <a href="#">K_U08</a></li></ul> | <ul style="list-style-type: none"><li>• an evaluation test</li><li>• an ongoing monitoring during classes</li></ul> | <ul style="list-style-type: none"><li>• Lecture</li><li>• Laboratory</li></ul> |
| Student zna budowę procesora i typy pamięci, potrafi analizować listy rozkazów.                                                              | <ul style="list-style-type: none"><li>• <a href="#">K_W06</a></li></ul>                                 | <ul style="list-style-type: none"><li>• an evaluation test</li></ul>                                                | <ul style="list-style-type: none"><li>• Lecture</li></ul>                      |
| Student potrafi analizować i projektować proste układy cyfrowe zrealizowane za pomocą układów dyskretnych lub układów programowalnych.       | <ul style="list-style-type: none"><li>• <a href="#">K_W15</a></li></ul>                                 | <ul style="list-style-type: none"><li>• an evaluation test</li></ul>                                                | <ul style="list-style-type: none"><li>• Lecture</li></ul>                      |

## Assignment conditions

**Wykład** - warunkiem zaliczenia jest uzyskanie pozytywnej oceny z kolokwiów przeprowadzonych w formie pisemnej.

**Laboratorium** - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich ćwiczeń laboratoryjnych, przewidzianych do realizacji w ramach programu laboratorium.

**Składowe oceny końcowej** = wykład: **60%** + laboratorium: **40%**.

## Recommended reading

1. Chmiel K.: Teoria układów logicznych, Wydawnictwo Politechniki Poznańskiej, Poznań, 1994.
2. G. De Micheli: Synteza i optymalizacja układów cyfrowych, WNT, Warszawa, 1998.
3. Lisiecka-Frąszczak J.: Synteza układów cyfrowych, Wydawnictwo Politechniki Poznańskiej, Poznań, 2000.
4. W. Stallings, Organizacja i architektura systemu komputerowego, WNT, Warszawa, 2004.
5. Łuba T., Zbierchowski B.: Komputerowe projektowanie układów cyfrowych, WKiŁ, Warszawa, 2000
6. Biernat J., Architektura komputerów, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2013.

## Further reading

1. Biernat J., Metody i układy arytmetyki komputerowej, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2001.
2. Łuba T.: Synteza układów logicznych, WSISiZ, Warszawa, 2000.
3. Majewski W.: Układy logiczne, WNT, Warszawa, 1992.
4. Majewski Wł., Łuba T., Jasiński K., Zbierchowski B.: Programowalne moduły logiczne w syntezie układów cyfrowych, WKiŁ, Warszawa, 1992.
5. Pieńkos J., Turczyński J.: Układy scalone TTL w systemach cyfrowych, WKiŁ, Warszawa, 1986.

## Notes

Modified by dr hab. inż. Krzysztof Sozański, prof. UZ (last modification: 19-09-2016 07:55)

Generated automatically from SylabUZ computer system