

Sterowanie procesami dyskretnymi - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Sterowanie procesami dyskretnymi
Kod przedmiotu	06.0-WE-AiRP-SPD
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Automatyka i robotyka
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. inżyniera
Semestr rozpoczęcia	semestr zimowy 2018/2019

Informacje o przedmiocie	
Semestr	3
Liczba punktów ECTS do zdobycia	6
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	30	2	18	1,2	Egzamin
Laboratorium	30	2	18	1,2	Zaliczenie na ocenę

Cel przedmiotu

- Zapoznanie studentów z zagadnieniami sterowania dyskretnego, w których algorytm sterowania zapisany jest w postaci modelu sekwencyjnego (FSM), współbieżnego (sieci Petriego, SFC) i hierarchicznego (maszyna stanów UML).
- Wykształcenie umiejętności modelowania układów sterowania i ich formalnej weryfikacji.

Wymagania wstępne

- Podstawy systemów dyskretnych
- Architektura systemów komputerowych

Zakres tematyczny

- Formalna specyfikacja procesów dyskretnych na poziomie behawioralnym: sieci działań, hierarchiczna mapa stanów (statechart, maszyna stanów UML), hierarchiczne sieci Petriego.
- Modularna specyfikacja behawioralna programów sterowania logicznego z wykorzystaniem hierarchicznych sieci SFC i sieci Petriego: sieci SFC, relacje między siecią SFC a siecią Petriego, projektowanie modularne, rdzenie projektowe. Rola specyfikacji formalnej w programowaniu sterowników przemysłowych PLC.
- UML jako narzędzie specyfikacji systemów reaktywnych. Diagram maszyny stanów. Diagram aktywności. Przypadki użycia. Rola UML w dokumentowaniu i syntezie oprogramowania dla cyfrowych mikrosystemów wbudowanych.
- Weryfikacja formalna: zastosowanie teorii sieci Petriego.
- Architektura sterowników logicznych: mikrokontroler jako sterownik logiczny, mikrosystemy cyfrowe SoC. Przemysłowe sterowniki logiczne PLC. Wbudowany, rekonfigurowalny sterownik logiczny RLC.
- Programowa lub strukturalna realizacja sterowników logicznych: programowanie sterowników RLC na podstawie specyfikacji behawioralnej. Synteza strukturalna sterowników wbudowanych metodami formalnymi na podstawie specyfikacji behawioralnej. Rola języków opisu sprzętu (VHDL, Verilog) w syntezie systemowej.
- Specyfikacja i modelowanie algorytmów sterowania binarnego na poziomie systemowym z wykorzystaniem UML i profesjonalnego oprogramowania do komputerowego projektowania mikrosystemów cyfrowych.

Metody kształcenia

Wykład: wykład konwencjonalny.

Laboratorium: ćwiczenia laboratoryjne.

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Potrafi tworzyć abstrakcyjne modele w języku UML całych systemów, w których sterowanie jest elementem centralnym.	• K_W11	• egzamin - ustny, opisowy, testowy i inne	• Wykład
Posiada wiedzę na temat weryfikacji formalnej sterowników.	• K_W11	• egzamin - ustny, opisowy, testowy i inne	• Wykład

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Posiada umiejętność opisu programu sterowania (sterowniki PLC i RLC).	• K_W11	• bieżąca kontrola na zajęciach	• Laboratorium
Posiada umiejętność projektowania sterowników zarówno PLC jak i realizowanych jako układ cyfrowy.	• K_W11	• bieżąca kontrola na zajęciach	• Laboratorium
Posiada wiedzę co do klasycznego definiowania zagadnienia sterowania dyskretnego zarówno sekwencyjnego, współbieżnego jak i hierarchicznego (FSM, sieć Petriego, maszyna stanów).	• K_W11	• egzamin - ustny, opisowy, testowy i inne	• Wykład

Warunki zaliczenia

Wykład: warunkiem zaliczenia jest uzyskanie pozytywnej oceny z egzaminu przeprowadzonego w formie pisemnej lub ustnej.

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

Składowe oceny końcowej: wykład: 50% + laboratorium: 50%

Literatura podstawowa

1. Adamski M., Chodań M.: Modelowanie układów sterowania dyskretnego z wykorzystaniem sieci SFC, Wyd. Politechniki Zielonogórskiej, Zielona Góra, 2000.
2. Dąbrowski W., Stasiak A., Wolski M.: Modelowanie systemów informatycznych w języku UML 2.1, PWN, Warszawa, 2007.
3. Adamski M., Karatkevich A., Węgrzyn M.: Design of Embedded Control Systems, Springer (USA), New York, 2005.
4. Żurawski R.(Ed.): Embedded Systems Handbook, CRC, Boca Raton, 2006.
5. Evans M., Noble J., Hochenbaum J., Arduino w akcji, Helion, Gliwice, 2014.

Literatura uzupełniająca

1. Booch G., Rumbaugh J., Jacobson I.: UML. Przewodnik użytkownika, WNT, Warszawa, 2001.
2. David D., Alla H.: Petri Nets & Grafcet. Tools for modeling discrete event systems, Prentice Hall, New York, 1992.
3. Gajski D.D, Vahid F., Narayan S., Grong J.: Specification and Design of Embedded Systems, Pretice Hall, Englewood Cliffs, New Jersey, 1994.
4. Jerraya A., Mernet J. (Ed): System-Level Synthesis, Kluwer, Dordecht, 1999.
5. Yakovlev, Gomes L., L. Lavagno (Ed.): Hardware Design and Petri Nets, Kluwers Academic Publishers, Boston, 2000.
6. Riley M., Inteligentny dom. Automatyzacja mieszkania za pomocą platformy Arduino, systemu Android i zwykłego komputera, Helion, Gliwice, 2013.

Uwagi

Zmodyfikowane przez dr inż. Grzegorz Bazydło (ostatnia modyfikacja: 16-03-2018 13:15)

Wygenerowano automatycznie z systemu SylabUZ