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 2022/2023

Informacje o przedmiocie

Semestr	3
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	• dr inż. Grzegorz Bazydło

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) 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), sieci Petriego.
- Modularna specyfikacja behawioralna programów sterowania logicznego z wykorzystaniem hierarchicznych sieci Petriego. Rola specyfikacji formalnej w programowaniu sterowników przemysłowych PLC.
- UML jako narzędzie specyfikacji systemów reaktywnych. Diagram maszyny stanów, czynności, przypadków użycia. Rola UML w dokumentowaniu i syntezie oprogramowania dla cyfrowych mikrosystemów wbudowanych.
- Weryfikacja formalna z zastosowaniem 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 sieci Petriego oraz profesjonalnego oprogramowania do komputerowego projektowania mikrosystemów cyfrowych.

Metody kształcenia

Wykład: wykład konwencjonalny.
Laboratorium: ćwiczenia laboratoryjne.

Efekty uczenia się i metody weryfikacji osiągania efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Potrafi tworzyć abstrakcyjne modele w języku UML oraz za pomocą sieci Petriego 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
Posiada umiejętność opisu programu sterowania za pomocą różnych metod i języków modelowania.	• K_W11	• bieżąca kontrola na zajęciach	• Laboratorium

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Posiada umiejętność projektowania sterowników logicznych, w tym 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.
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. Dąbrowski W., Stasiak A., Wolski M., Modelowanie systemów informatycznych w języku UML 2.1, PWN, Warszawa, 2007.
2. Szpyrka M., Sieci Petriego w modelowaniu i analizie systemów współbieżnych. Inżynieria oprogramowania, WNT, Warszawa, 2016.
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. Booch G., Rumbaugh J., Jacobson I., UML. Przewodnik użytkownika, WNT, Warszawa, 2001.

Literatura uzupełniająca

1. Reisig W., Petri Nets: An Introduction, Berlin, Germany:Springer-Verlag, 2012.
2. Yakovlev, Gomes L., L. Lavagno (Ed.), Hardware Design and Petri Nets, Kluwers Academic Publishers, Boston, 2000.
3. Riley M., Inteligentny dom. Automatyzacja mieszkania za pomocą platformy Arduino, systemu Android i zwykłego komputera, Helion, Gliwice, 2013.
4. Evans M., Noble J., Hochenbaum J., Arduino w akcji, Helion, Gliwice, 2014.
5. Bazydło G., Graphic specification of programs for reconfigurable logic controllers using Unified Modeling Language, University of Zielona Góra Press, Lecture Notes in Control and Computer Science, Zielona Góra, 2012.
6. Wiśniewski R., Prototyping of Concurrent Control Systems Implemented in FPGA Devices, Cham, Switzerland:Springer, 2017.

Uwagi

Zmodyfikowane przez dr hab. inż. Wojciech Paszke, prof. UZ (ostatnia modyfikacja: 11-04-2022 09:03)

Wygenerowano automatycznie z systemu SylabUZ