



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

PO5: Programowanie sterowników PLC - Programowanie przemysłowych aplikacji napędowych

Przedmiot

Kierunek studiów

Elektromobilność

Studia w zakresie (specjalność)

Poziom studiów

pierwszego stopnia

Forma studiów

stacjonarne

Rok/semestr

3/6

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

30

Ćwiczenia

Laboratoria

30

Projekty/seminaria

15

Inne (np. online)

Liczba punktów ECTS

6

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr hab. inż. Cezary Jędryczka

mail: Cezary.Jedryczka@put.poznan.pl

tel.: 61 665 2396

Wydział Automatyki, Robotyki i Elektrotechniki

ul. Piotrowo 3a, 60-965 Poznań

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Michał Krystkowiak

mail: Michal.Krystkowiak@put.poznan.pl

tel.: 616652388

Wydział Automatyki, Robotyki i Elektrotechniki

ul. Piotrowo 3a, 60-965 Poznań

Wymagania wstępne

Wiedza - student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z podstaw elektroniki cyfrowej, programowania i automatyki, powinien również posiadać umiejętność pozyskiwania informacji ze wskazanych źródeł oraz mieć gotowość do podjęcia współpracy w ramach zespołu.

Umiejętności - umiejętność efektywnego samokształcenia w dziedzinie związanej z wybranym kierunkiem studiów; umiejętność podejmowania właściwych decyzji przy rozwiązywaniu prostych zadań oraz formułowaniu problemów z zakresu programowania sterowników PLC.



Kompetencje - student ma świadomość poszerzania swoich kompetencji, wykazuje gotowość do pracy w zespole, zdolność do podporządkowania się regułom obowiązującym podczas zajęć wykładowych i laboratoryjnych.

Cel przedmiotu

Pozyskanie wiedzy i umiejętności dotyczących programowania przemysłowych układów napędowych. Poznanie pojęć systemów czasu rzeczywistego oraz sterowników PLC, zapoznanie się z architekturą synerowników PLC, zapoznanie się z językami programowania sterowników PLC, nabycie umiejętności obsługi i konfiguracji sterowników oraz opracowania i implementowania algorytmów realizujących różne funkcje ze szczególnym uwzględnieniem aplikacji dedykowanych do szeroko pojętej elektromobilności.

Przedmiotowe efekty uczenia się

Wiedza

1. Student powinien posiadać wiedzę dotyczącą budowy oraz zasady działania programowalnych sterowników logicznych PLC.
2. Student powinien posiadać wiedzę dotyczącą znajomości wybranych języków programowania wykorzystywanych do implementacji opracowanych algorytmów sterowania.
3. Student powinien znać podstawowe pojęcia związane z konfiguracją i obsługą przemysłowych układów napędowych.

Umiejętności

1. Student będzie potrafił wykorzystać wiedzę o budowie i zasadach działania sterowników PLC i serwonapędów przemysłowych do realizacji zadań technicznych.
2. Student będzie potrafił opracować i zaimplementować algorytmy sterowania w wybranych językach programowania sterowników PLC.
3. Student będzie potrafił zaprogramować wybrane aplikacje napędowe z wykorzystaniem standaryzowanych bloków funkcyjnych PLC OPEN Motion Control.

Kompetencje społeczne

1. Student rozumie znaczenie wiedzy w rozwiązywaniu problemów i podnoszenia kompetencji zawodowych, osobistych i społecznych.
2. Student ma świadomość, że w technice wiedza i umiejętności bardzo szybko stają się przestarzałe.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykład:

- ocena wiedzy i umiejętności wykazanych w rozwiązywanym teście pisemnym o charakterze problemowym,
- ocenianie ciągłe, premiowanie aktywności i merytorycznych treści wypowiedzi.



Laboratorium:

- weryfikacja na podstawie wykonanych sprawozdań,
- ocenianie ciągłe, premiowanie aktywności i merytorycznych treści wypowiedzi.

Treści programowe

Budowa i zasada działania programowalnych sterowników logicznych PLC, cykl pracy sterownika, układy wejść i wyjść sterowników, języki programowania, podstawy programowania w języku drabinkowym. Budowa i zasada działania wybranych czujników i urządzeń pomiarowych stosowanych w automatyce i robotyce. Obsługa układów czasowych i licznikowych w PLC z uwzględnieniem, tzw. szybkich liczników (dedykowanych do współpracy z przetwornikami położenia kątownego), obsługa systemu przerwań w sterownikach PLC, obsługa wyjść impulsowych w sterownikach PLC, układy regulacji zamkniętej (algorytmy regulatorów w sterownikach PLC). Serwonapęd, struktura, parametry i wymagania. Metody programowania aplikacji napędowych z wykorzystaniem standaryzowanych bloków funkcyjnych zgodnych z PLC Motion control. Współczesne trendy rozwoju w zakresie tworzenia i programowania aplikacji napędowych.

Metody dydaktyczne

Wykład: prezentacja zagadnień z wykorzystaniem środków multimedialnych, ilustrowana przykładami podawanymi na tablicy, dyskusja nad zagadnieniami problemowymi, wyniki badań modeli symulacyjnych.

Laboratorium: wykonywanie ćwiczeń laboratoryjnych w zespołach (przygotowanie stanowiska, zbudowanie układów pomiarowych, wykonanie eksperymentów) z pomocą i pod kontrolą prowadzącego, badania modeli symulacyjnych i eksperymentalnych - porównanie uzyskanych wyników.

Literatura

Podstawowa

1. Dokumentacja techniczna wybranych sterowników PLC oraz serwonapędów
2. Kwaśniewski J., Sterowniki PLC w pracy inżynierskiej, PTC, Kraków 2008.
3. Legierski T., Programowanie sterowników PLC, WPKJS, Gliwice 1998.
4. Zieliński T.P., Cyfrowe przetwarzanie sygnałów. Od teorii do zastosowań, Wydawnictwa Komunikacji i Łączności, Warszawa 2009.
5. Sałat R., Korpysz K., Obstawski P., Wstęp do programowania sterowników PLC, WKŁ, 2014.

Uzupełniająca

1. Normy dotyczące języków programowania sterowników PLC



2. Dokumentacja standardu PLC Open Motion Control

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	155	6,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	85	3,5
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwίων/egzaminu, wykonanie projektu) ¹	70	2,5

¹ niepotrzebne skreślić lub dopisać inne czynności