



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Inżynieria oprogramowania dla systemów wbudowanych i mobilnych [S2Inf1-PB>IOSWM]

### Przedmiot

Kierunek studiów  
Informatyka

Rok/Semestr  
2/3

Studia w zakresie (specjalność)  
Przetwarzanie brzegowe

Profil studiów  
ogólnoakademicki

Poziom studiów  
drugiego stopnia

Język oferowanego przedmiotu  
polski

Forma studiów  
stacjonarne

Wymagalność  
obligatoryjny

### Liczba godzin

Wykład  
15

Laboratorium  
0

Inne (np. online)  
0

Ćwiczenia  
0

Projekty/seminaria  
0

### Liczba punktów ECTS

1,00

### Koordynatorzy

dr inż. Michał Melosik  
michal.melosik@put.poznan.pl

### Wykładowcy

### Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę na temat architektury systemów wbudowanych, strategii modelowania i analizy mikrosystemów zarówno w obszarze warstwy fizycznej jak i warstwy oprogramowania oraz znać wykorzystywane w mikrosystemach protokoły transmisji przewodowej i bezprzewodowej. Ponadto student powinien posiadać elementarną wiedzę na temat technologii rekonfigurowalnych, np. obwodów programowalnej logiki typu FPGA oraz na temat układów programowalnych typu kontrolery. Powinien posiadać umiejętność programowania oraz znajomość języków opisu sprzętu. Wymagana jest również umiejętność pozyskiwania informacji ze wskazanych źródeł. Powinien również rozumieć konieczność poszerzania swoich kompetencji w zakresie systemów wbudowanych, zwłaszcza metodologii tworzenia oprogramowania dla urządzeń programowalnych, hybrydowych i mobilnych i mieć gotowość do podjęcia współpracy w ramach zespołu. Ponadto student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.

## Cel przedmiotu

1. Poszerzenie wiedzy teoretycznej w zakresie metodologii rozwijania oprogramowania dla systemów wbudowanych, przekazanie informacji o głównych problemach dziedziny. 2. Zaznajomienie z literaturą dotyczącą projektowania, rozwijania i użytkowania mikrosystemów. 3. Przekazanie wiedzy na temat metod i strategii projektowania i modelowania oprogramowania dla systemów wbudowanych wykorzystywanych w rozwiązaniach przemysłowych. 4. Przedstawienie różnic między tworzeniem oprogramowania dla systemów komputerowych a systemów wbudowanych. 5. Omówienie znaczenia wybranych metod rozwijania i testowania oprogramowania, charakterystyka narzędzi wykorzystywanych w procesie rozwijania oprogramowania dla mikrosystemów. 6. Omówienie ograniczeń w rozwoju oprogramowania oraz zagadnień bezpieczeństwa i reguł użytkowania licencji systemów wbudowanych. 7. Przedstawienie specyfiki rynku systemów wbudowanych, głównych trendów jego rozwoju oraz charakterystyka rynku pracy w tym obszarze w kraju i na świecie. 8. Uświadomienie ograniczeń prawnych, ekonomicznych i technicznych w rozwoju dziedziny.

## Przedmiotowe efekty uczenia się

Wiedza:

absolwent zna:

- podstawy teoretyczne projektowania i rozwijania dużych systemów informatycznych,
- techniki rozwijania oprogramowania dla branży systemów wbudowanych,
- zagadnienia rozwijania systemów interdyscyplinarnych,
- cykle życia oprogramowania oraz narzędzia wspomagające jego rozwijanie,
- techniki modelowania złożonych systemów informatycznych.

Umiejętności:

absolwent potrafi:

- formułować wymagania funkcjonalne i нефункционалне systemów informatycznych oraz jest świadomy problemów komunikacyjnych pomiędzy zespołami ekspertów a zespołem deweloperów oprogramowania,
- integrować wiedzę dotyczącą warstwy sprzętowej i warstwy oprogramowania w celu realizacji systemu dedykowanego również dla pozainformatycznych zastosowań,
- wykorzystać metody oceny pracochłonności projektu oraz zarządzać ryzykiem przy realizacji projektu informatycznego,
- wybrać narzędzia wspierające rozwijanie projektów dla branży systemów wbudowanych,
- ocenić złożoność zadań przewidzianych przy rozwijaniu oprogramowania dla branży systemów wbudowanych, w tym również projektów interdyscyplinarnych, cechujących się potencjałem badawczym.

Kompetencje społeczne:

rozumie, że wiedza i umiejętności z zakresu technik rozwijania oprogramowania dla systemów wbudowanych i mobilnych nabyte w toku kształcenia bardzo szybko stają się niewystarczające i wymagają aktualizacji i jest gotów na samodzielne zdobywanie wiedzy niezbędnej do realizacji projektów informatycznych zgodnych z obowiązującymi standardami rozwijania oprogramowania i komunikowania jego modułów; potrafi myśleć i działać w sposób przedsiębiorczy.

## Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Efekty kształcenia przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca w zakresie wykładów:

- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach;
- na podstawie prezentacji rozwiązania problemów zadanych na poprzednich wykładach.

Ocena podsumowująca:

Sprawdzanie założonych efektów kształcenia realizowane jest przez:

- ocenę wiedzy i umiejętności związanych z samodzielnym rozwiązywaniem zadań o charakterze problemowym poprzez sprawdzian pisemny,
- omówienie wyników sprawdzianu zaliczeniowego.

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:

- omówienia dodatkowych aspektów zagadnienia,
- uwagi związane z udoskonaleniem materiałów dydaktycznych,

- wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.

## Treści programowe

Program wykładu obejmuje omówienie następujących zagadnień: Różne klasy systemów wbudowanych i porównanie ich funkcjonalności do systemów komputerowych; wymagania dla systemów wbudowanych pod kątem efektywnego wykorzystania zasobów sprzętowych oraz wyboru właściwego rozwiązania technologicznego dyktowanego aspektami funkcjonalnymi i ekonomicznymi; metody projektowania, modelowania i analizy oprogramowania; osadzanie aplikacji w istniejącym systemie zależnie od jego wersji, wraz z omówieniem specyfiki pracy z systemami uniksowymi i systemem eCos; zaznajomienie z analizą BOM (Bill Of Materials); przedstawienie scenariuszy testowania i metod automatyzacji testów; omówienie narzędzi wykorzystywanych w procesie rozwijania oprogramowania: gcc, qemu, gdb, JTAG oraz narzędzi kontroli wersji: GIT, SVN; przedstawienie istniejących emulatorów platform sprzętowych i strategii rozwoju wielkosystemowych projektów; omówienie metod budowania interfejsu użytkownika wraz z opracowaniem mechanizmu interakcji z użytkownikiem i otoczeniem (UX - User eXperience); uwzględnienie w procesie projektowania oprogramowania warunków środowiskowych (warunków klimatycznych, mechanicznych, czasu pracy); wyszczególnienie zasad tworzenia dokumentacji na różnych poziomach rozwoju oprogramowania (na etapie tworzenia specyfikacji, opracowywania architektury, implementacji, rekonfiguracji, testowania, serwisowania); porównanie rozwiązań komercyjnych i Open Source; omówienie zagrożeń i zasad bezpieczeństwa; przedstawienie różnic w obszarze mikrosystemów mikroprocesorowych, mikrosystemów rekonfigurowalnych, hybrydowych; nakreślenie specyfiki rynku systemów wbudowanych i trendów jego rozwoju, a także rynku pracy w tematyce systemów wbudowanych ? zarówno w skali kraju, jak i w skali świata; omówienie ograniczeń prawnych, ekonomicznych i technicznych w rozwoju dziedziny; podsumowanie głównych wyzwań stojących na drodze rozwoju systemów wbudowanych i mikrosystemów; omówienie efektywnego wykorzystania języków programowania i skryptowych typowych dla systemów wbudowanych (C/C++, bash, Python); podanie przykładowych mechanizmów rozbudowy systemów wbudowanych, szczególnie pod kątem tworzenia sterowników dla urządzeń peryferyjnych oraz całego procesu przygotowania portu istniejącego systemu na nowe platformy sprzętowe.

## Metody dydaktyczne

Prezentacja multimedialna ilustrowana przykładami podawanymi na tablicy.

## Literatura

Podstawowa

1. UML: przewodnik użytkownika, Grady Booch, James Rumbaugh, Ivar Jacobson, WTN 2001.
2. The industrial electronics handbook Wilanowski B, Irwin D., Taylor & Francis, 2011.

Uzupełniająca

1. Real-Time Systems Design and Analysis: Tools for the Practitioner, P. A. Laplante, S. J. Ovaska, Wiley, 2012.

## Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	25	1,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	15	0,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	10	0,50