



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Architektura systemów wbudowanych

Przedmiot

Kierunek studiów

Informatyka

Studia w zakresie (specjalność)

Mikrosystemy informatyczne

Poziom studiów

drugiego stopnia

Forma studiów

stacjonarne

Rok/semestr

1/1

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obligatoryjny

Liczba godzin

Wykład

30

Ćwiczenia

Laboratoria

30

Projekty/seminaria

Inne (np. online)

Liczba punktów ECTS

5

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Adam Turkot

email: adam.turkot@put.poznan.pl

tel. 61 6652284

ul. Piotrowo 3a, 60-965 Poznań

Odpowiedzialny za przedmiot/wykładowca:

Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu protokołów sieciowych UDP, SSH, ICMP, programowania w języku C i C++, w interpreterze poleceń Bash oraz w języku Python i JavaScript. Powinien również rozumieć konieczność poszerzania swoich kompetencji / mieć gotowość do podjęcia współpracy w ramach zespołu. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.



Cel przedmiotu

Zaznajomienie studentów z metodologią projektowania architektury systemów wbudowanych. Przekazanie studentom poszerzonej wiedzy z zakresu architektur systemów wbudowanych single-processor, multiprocessor, multi-computer. Wykształcenie umiejętności techniki programowania zapewniającej: efektywne wykorzystanie zasobów sprzętowych systemów wbudowanych; optymalne, dla danego zadania realizacje aplikacji z użyciem mikrokontrolera, z obsługą dedykowanych modułów peryferyjnych i przy uwzględnieniu wymogów związanych z oszczędnością energii i wydajnością obliczeniową. Opanowanie technik komunikacji pomiędzy mikrokontrolerami, a cyfrowymi i analogowymi elementami systemów wbudowanych. Zaznajomienie studentów z możliwościami i ograniczeniami budowania systemów wbudowanych w oparciu o procesory sygnałowe i komputery osobiste. Kształtowanie u studentów umiejętności pracy zespołowej poprzez realizację elementów projektu i połączenie ich w całość.

Przedmiotowe efekty uczenia się

Wiedza

1. Ma zaawansowaną i pogłębioną wiedzę z zakresu szeroko rozumianych systemów informatycznych oraz metod i narzędzi wykorzystywanych do ich implementacji.
2. Ma zaawansowaną wiedzę szczegółową dotyczącą wybranych zagadnień z zakresu informatyki.
3. Ma zaawansowaną i szczegółową wiedzę o procesach zachodzących w cyklu życia systemów informatycznych.

Umiejętności

1. Potrafi przy formułowaniu i rozwiązywaniu zadań inżynierskich integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne.
2. Potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych.
3. Potrafi dokonać krytycznej analizy istniejących rozwiązań technicznych oraz zaproponować ich ulepszenia (usprawnienia) .
4. Potrafi ocenić przydatność metod i narzędzi służących do rozwiązania zadania inżynierskiego, polegającego na budowie lub ocenie systemu informatycznego lub jego składowych, w tym dostrzec ograniczenia tych metod i narzędzi.
5. Potrafi zgodnie z zadaną specyfikacją, zaprojektować złożone urządzenie, system informatyczny lub proces oraz zrealizować ten projekt używając właściwych metod, technik i narzędzi, w tym przystosowując do tego celu istniejące lub opracowując nowe narzędzia.

Kompetencje społeczne

1. Rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe.



2. Rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu informatyki w rozwiązywaniu problemów badawczych i praktycznych.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

a) w zakresie wykładów: na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach

b) w zakresie laboratoriów / ćwiczeń: na podstawie oceny bieżącego postępu realizacji zadań,

Ocena podsumowująca:

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez: egzamin ustny połączony z obroną projektu, w przypadku wątpliwości część pisemna (test w postaci elektronicznej na platformie Moodle);

b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez: ocenę przygotowania studenta do poszczególnych sesji zajęć laboratoryjnych poprzez sprawdzenie przygotowania zadanych projektów/ćwiczeń oraz ocenę umiejętności związanych z realizacją ćwiczeń laboratoryjnych, ocenianie ciągłe, na każdych zajęciach (odpowiedzi ustne) premiowanie przyrostu umiejętności posługiwania się poznanymi zasadami i metodami, ocenę dokumentacji tworzonej systematycznie wraz z postępami prac projektowych; dokumentacja przygotowawcza częściowo w trakcie zajęć, a częściowo po ich zakończeniu; ocena ta obejmuje także umiejętność pracy w zespole, ocenę i obronę przez studenta sprawozdania z realizacji projektu,

Treści programowe

Podstawy architektury systemów wbudowanych . Techniki efektywnego wykorzystania zasobów sprzętowych . Ocena możliwości sprzętowych . Środowisko programistyczne. Techniki programowania. Metody optymalizacji kodu . Interfejs użytkownika. Rozwiązania sprzętowe i programistyczne umożliwiające zarządzanie poborem mocy. Techniki zabezpieczeń oprogramowania (integralność programu , odporność na nieautoryzowanym kopiowaniem). Architektury komputerów. Architektury wieloprocesorowe i wielokomputerowe. Wzrost wydajności architektury wieloprocesorowej i wielokomputerowej w stosunku do systemów monoprocesorowych. Magistrale systemów wieloprocesorowych. Zasoby lokalne i współdzielone, konsekwencje współdzielenia zasobów. Typowe rozwiązania magistral STE, MULTIBUS, VME, PCI, COMPACT PCI. Zarządzanie dostępem do zasobów współdzielonych. Magistrale w systemach rozproszonych. Warstwa 4 i 7 modelu komunikacji ISO. Techniki sprzętowe i programowe dla zwiększenia niezawodności łącza komunikacyjnego. Protokoły komunikacji wykorzystywane w chmurze. Zwiększania niezawodności systemów bezobsługowych , techniki zapewniające gospodarkę energetyczną systemów autonomicznych.



Zajęcia laboratoryjne prowadzone są w formie piętnastu 2-godzinnych ćwiczeń, odbywających się w laboratorium, poprzedzonych 2-godziną sesją instruktazową na początku semestru. Ćwiczenia realizowane są przez 2-osobowe zespoły studentów.

Laboratoria obejmują: Wprowadzenie do programowania Intel Galileo. Instalacja systemu i sterowników na platformie Intel Galileo. Intel IoT Analytics w systemach wbudowanych. Wykorzystanie serwera WWW jako interfejsu komunikacji z systemem wbudowanym. Wykorzystanie analizatora stanów logicznych w konstruowaniu i testowaniu systemów wbudowanych. Komunikacja w chmurze (autoryzacja OAuth2, JSON, Google Drive). Programowanie Raspberry Pi w C i Python. Programowanie architektury SOM na przykładzie Colibri iMX7.

Metody dydaktyczne

Metody dydaktyczne:

wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, prezentacje wybranych rozwiązań studenckich.

ćwiczenia laboratoryjne: ćwiczenia praktyczne, wykonywanie eksperymentów, dyskusja, praca w zespole.

Literatura

Podstawowa

Linux w systemach embedded Bis, Marcin. Wydawnictwo btc, 2011

Building embedded Linux systems Yaghmour, Karim. O'Reilly, cop. 2003.

Uzupełniająca

Wbudowane systemy mikroprocesorowe Timofiejew, Aleksander., Akademia Podlaska (Siedlce).

Wydawnictwo. Wydawnictwo Akademii Podlaskiej, 2010.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	120	5
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2,5
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu) ¹	60	2,5

¹ niepotrzebne skreślić lub dopisać inne czynności

