



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Systemy oprogramowania układowego [S2Inf1-PB>SOUKL]

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
30

Laboratorium
30

Inne
0

Ćwiczenia
0

Projekty/seminaria
20

Liczba punktów ECTS

5,00

Koordynatorzy

dr inż. Mariusz Naumowicz
mariusz.naumowicz@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający przedmiot powinien posiadać podstawową wiedzę z zakresu systemów operacyjnych dla systemów wbudowanych, elektroniki oraz programowania. Powinien również rozumieć konieczność poszerzania swoich kompetencji oraz mieć gotowość do podjęcia współpracy w ramach zespołu.

Cel przedmiotu

- Przekazanie studentom wiedzy związanej z nowoczesnymi systemami oprogramowania układowego. - Zapoznanie studentów z nowoczesnymi metodami projektowania, testowania i prototypowania systemów oprogramowania układowego. - Rozwijanie u studentów umiejętności rozwiązywania złożonych problemów projektowych w zakresie systemów wbudowanych i systemów oprogramowania układowego. - Kształtowanie u studentów umiejętności pracy zespołowej.

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, szczególnie dotyczących budowania

warstwy sprzętowej systemów reprogramowalnych - [k2st_w1]

2. ma zaawansowaną wiedzę szczegółową dotyczącą wybranych zagadnień z zakresu informatyki, szczególnie dotyczącą konstruowania systemów wbudowanych - [k2st_w3]

3. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach informatyki i innych, wybranych, pokrewnych dyscyplin naukowych - [k2st_w4]

4. zna zaawansowane metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich i prowadzeniu prac badawczych w wybranym obszarze informatyki - [k2st_w6]

Umiejętności:

1. potrafi planować i przeprowadzać eksperymenty, w tym pomiary i symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski oraz formułować i weryfikować hipotezy związane ze złożonymi problemami inżynierskimi i prostymi problemami badawczymi - [k2st_u3]

2. 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 - [k2st_u5]

3. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [k2st_u6]

4. potrafi dokonać krytycznej analizy istniejących rozwiązań technicznych oraz zaproponować ich ulepszenia (usprawnienia) - [k2st_u8]

5. potrafi - stosując m.in. koncepcyjnie nowe metody - rozwiązywać złożone zadania informatyczne, w tym zadania nietypowe oraz zadania zawierające komponent badawczy - [k2st_u10]

6. 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 - [k2st_u11]

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [k2st_k1]

2. rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu informatyki w rozwiązywaniu problemów badawczych i praktycznych - [k2st_k2]

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: na podstawie oceny bieżącego postępu realizacji zadań.

Ocena podsumowująca:

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest poprzez prezentację wybranego zagadnienia w trakcie wykładu,

b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez sprawdzian projektowy i ocenę zadań realizowanych w ramach każdego spotkania laboratoryjnego;

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

- omówienia dodatkowych aspektów zagadnienia,

- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,

- umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium.

Treści programowe

Treści programowe przedmiotu dotyczą zagadnień związanych z szeroko rozumianym oprogramowaniem układowym(ang. firmware).

Tematyka zajęć

Tematyka wykładów jest następująca: Boot Flow, FSP, SMM, Requirements, SMBIOS and ACPI, UEFI Secure Boot, UEFI Shell API, Scripts & CLI, Network Boot, Driver Execution Environment(DXE), UEFI Drivers , developing an open source software feature for firmware.

Zajęcia laboratoryjne prowadzone są w formie 2-godzinnych spotkań, odbywających się w laboratorium, poprzedzonych sesją instruktażową na początku semestru. Ćwiczenia realizowane są przez 2-osobowe zespoły studentów. Program laboratorium obejmuje następujące zagadnienia:

- przygotowanie środowiska EDK II,
- konsola UEFI,
- pisanie aplikacji przeznaczonych dla UEFI,
- pisanie sterowników UEFI,
- debugowanie procesu uruchamiania UEFI,
- testy jednostkowe,
- przenoszenie aplikacji na inne systemy.

Projekt realizowany przez studentów dotyczy praktycznego uruchomienia i zastosowania wybranej technologii na poziomie oprogramowania układowego. Może rozszerzać materiał omówiony na laboratorium, lub dotyczyć innego zagadnienia zaakceptowanego przez prowadzącego zajęcia.

Część wymienionych wyżej treści programowych realizowana jest w ramach pracy własnej studenta.

Metody dydaktyczne

1. Wykład: prezentacja multimedialna uzupełniona przykładami podawanymi na tablicy.
2. Ćwiczenia laboratoryjne: ćwiczenia praktyczne, dyskusja, praca w zespole, zawody projektowe.

Literatura

Podstawowa:

1. Vincent Zimmer, Michael Rothman, Suresh Marisetty, Beyond BIOS, De Gruyter 2017. ISBN: 9781501505836.

Uzupełniająca:

1. Unified Extensible Firmware Interface - https://uefi.org/sites/default/files/resources/UEFI%20Spec%20_6.pdf

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	125	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	80	3,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	45	1,50