



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Projektowanie układów rekonfigurowalnych

Przedmiot

Kierunek studiów

Informatyka

Studia w zakresie (specjalność)

Aplikacje mobilne i wbudowane dla Internetu przedmiotów

Poziom studiów

drugiego stopnia

Forma studiów

niestacjonarne

Rok/semestr

2/3

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

12

Laboratoria

20

Inne (np. online)

Ćwiczenia

Projekty/seminaria

Liczba punktów ECTS

3

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Marek Kropidłowski

email: marek.kropidlowski@put.poznan.pl

tel: 616652297

Instytut Informatyki

adres: ul. Piotrowo 3 60-965 Poznań

Odpowiedzialny za przedmiot/wykładowca:

dr hab. inż. Paweł Śniatała

email: pawel.sniatala@put.poznan.pl

tel: 616652297

Instytut Wydział Informatyki

adres: ul. Piotrowo 3 60-965 Poznań

Wymagania wstępne

Student rozpoczynający przedmiot powinien posiadać podstawową wiedzę z zakresu elektroniki analogowej i cyfrowej oraz programowania strukturalnego.

Student powinien posiadać umiejętność pozyskiwania informacji ze wskazanych źródeł. Powinien również rozumieć konieczność poszerzania swoich kompetencji oraz mieć gotowość do podjęcia współpracy w ramach zespołu.

W zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.



Cel przedmiotu

- Przekazanie studentom wiedzy związanej z nowoczesnymi układami reprogramowalnymi oraz ich zastosowaniem w systemach cyfrowych i platformach sprzętowych dla systemów IoT.
- Zapoznanie studentów z nowoczesnymi metodami projektowania, testowania i prototypowania urządzeń elektronicznych bazujących na układach reprogramowalnych.
- Rozwijanie u studentów umiejętności rozwiązywania złożonych problemów projektowych w zakresie systemów cyfrowych.
- Kształtowanie u studentów umiejętności pracy zespołowej.

Przedmiotowe efekty uczenia się

Wiedza

Ma uporządkowaną i podbudowaną teoretycznie wiedzę ogólną związaną z kluczowymi zagadnieniami z zakresu informatyki szczególnie projektowania systemów wbudowanych i oprogramowania dla systemów z ograniczeniami sprzętowymi (K2st_W2)

Ma zaawansowaną wiedzę szczegółową dotyczącą wybranych zagadnień z zakresu informatyki, szczególnie dotyczącą konstruowania systemów wbudowanych (K2st_W3)

Ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach informatyki i pokrewnych dyscyplin naukowych, szczególnie dotyczącą nowoczesnych systemów cyfrowych (K2st_W4)

Zna zaawansowane metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich i prowadzeniu prac badawczych związanych z uruchamianiem i testowaniem dedykowanych systemów cyfrowych (K2st_W6)

Umiejętności

Potrafi planować i przeprowadzać eksperymenty, w tym pomiary analizatorami stanów logicznych i symulacje systemów cyfrowych opisanych językami HDL, interpretować uzyskane wyniki i wyciągać wnioski oraz formułować i weryfikować hipotezy związane ze złożonymi problemami inżynierskimi i prostymi problemami badawczymi (K2st_U3)

Potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne (K2st_U4)

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)

Potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych (K2st_U6)

Potrafi - stosując m.in. koncepcyjnie nowe metody - rozwiązywać złożone zadania informatyczne związane z rekonfiguracją dynamiczną warstwy sprzętowej urządzeń. (K2st_U10)

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

Rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe (K2st_K1)



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 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: test w formie elektronicznej na platformie Moodle;

b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez: sprawdzian projektowy i ocena zadań realizowanych w ramach spotkań laboratoryjnych;

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

i. omówienia dodatkowych aspektów zagadnienia,

ii. efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,

iii. umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium,

iv. uwagi związane z udoskonaleniem materiałów dydaktycznych,

v. wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.

Treści programowe

Program wykładu obejmuje następujące zagadnienia:

Podzbiór syntezowalny języka VHDL wg normy IEEE 1076-2008. Zasady tworzenia opisu na poziomie RTL (tzw. HDL 'good practice'). Metody projektowania układów kombinacyjnych i sekwencyjnych z wykorzystaniem języków opisu sprzętu. Zasady projektowania i implementacji automatów. Budowa złożonego środowiska testowego (testbench), testbench z automatyczną weryfikacją. Parametryzacja kodu, pakietów i bibliotek VHDL.

Synteza i implementacja urządzeń cyfrowych na platformach programowalnych FPGA i CPLD. Zależność opisu HDL od platformy implementacji (FPGA vs. ASIC). Konstrukcja nowoczesnych układów rekonfigurowalnych typu FPGA/CPLD/FPGA/FPOA/3D-PLD/PSoC. Rola komponentów IP (Intellectual Property) w projektowaniu złożonych systemów cyfrowych. Wykorzystanie rekonfiguracji zdalnej (IRL), częściowej i dynamicznej (PDR) w warstwie sprzętowej systemów wbudowanych.

Systemy z rekonfiguracją dynamiczną: modele pamięci konfiguracji, metody projektowania, DRP, ICAP, wykorzystanie w systemach wbudowanych; studium przypadku: Driver assistance system.



Softprocesory i platformy rekonfigurowalne dla systemów wbudowanych: przegląd rozwiązań, metody i narzędzia implementacji [studium przypadku: procesor LEON3], systemy hybrydowe, układy AP SoC [studium przypadku: Xilinx Zynq].

Metody testowania i weryfikacji prototypów urządzeń cyfrowych na platformach FPGA: scenariusze weryfikacji prototypów, OCI (On-Chip Instrumentation), metody testowania zdalnego, symulacja z modelem funkcjonalnym (BFM), debugowanie w systemie (ISD), emulacja (QEMU, QBox), ko-symulacja (HIL).

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

Badanie wpływu stosowanego stylu opisu na wyniki syntezy. Konstruowanie zaawansowanych środowisk testowania (testbench). Wykorzystanie techniki OCI do weryfikacji prototypów systemów cyfrowych FPGA. Projektowanie i implementacja urządzeń cyfrowych z wykorzystaniem oprogramowania firm Mentor Graphics, Xilinx. Realizacja wybranych sterowników sprzętowych dla systemów z interfejsem AXI, AXI-Lite, AXI-Stream.

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

Metody dydaktyczne

Wykład: prezentacja multimedialna, ilustrowana przykładami podawanymi na tablicy.

Ćwiczenia laboratoryjne: prezentacja multimedialna prezentacja ilustrowana przykładami podawanymi na tablicy oraz wykonanie zadań podanych przez prowadzącego - ćwiczenia praktyczne.

Literatura

Podstawowa

1. Mark Zwoliński, Projektowanie układów cyfrowych z wykorzystaniem języka VHDL, WKŁ2007, ISBN: 9788320616354.
2. Andrew Rushton, VHDL for Logic Synthesis, Third Edition, John Wiley & Sons, 2011, ISBN: 978-0-470-68847-2
3. Scott Hauck, Andre DeHon, Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation / Edition 1, Elsevier Science, November 2007, ISBN: 0123705223

Uzupełniająca

1. Michael Keating, Pierre Bricaud, Reuse Methodology Manual for System-on-a-Chip Designs / Edition 3, Springer-Verlag New York, August 2007, ISBN: 0387740988
2. Peter J. Ashenden, The Designer's Guide to VHDL / Edition 3, Elsevier Science, June 2008, ISBN: 0120887851.
3. Piotr Zbysiński, Jerzy Pasierbiński, Układy programowalne pierwsze kroki, Wydanie II, Wydawnictwo



BTC, Warszawa 2004, ISBN: 83-910067-0-0

4. Steve Kilts, Advanced FPGA Design: Architecture, Implementation, and Optimization, John Wiley & Sons, June 2007, ISBN: 0470054379.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	77	3
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	36	1.4
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych, przygotowanie do sprawdzianu, wykonanie dokumentacji projektowej do zadań)	41	1.6