

<b>KARTA OPISU MODUŁU KSZTAŁCENIA</b>		
Nazwa modułu/przedmiotu <b>Inżynieria oprogramowania dla Syst. Wbudowanych i Mobilnych</b>		Kod <b>1010542321010540037</b>
Kierunek studiów <b>Informatyka</b>	Profil kształcenia (ogólnoakademicki, praktyczny) <b>ogólnoakademicki</b>	Rok / Semestr <b>1 / 2</b>
Ścieżka obieralności/specjalność <b>Mikrosystemy informatyczne</b>	Przedmiot oferowany w języku: <b>polski</b>	Kurs (obligatoryjny/obieralny) <b>obligatoryjny</b>
Stopień studiów: <b>II stopień</b>	Forma studiów (stacjonarna/niestacjonarna) <b>stacjonarna</b>	
Godziny Wykłady: <b>15</b> Ćwiczenia: - Laboratoria: - Projekty/seminaria: -		Liczba punktów <b>2</b>
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) <b>kierunkowy</b>		(ogólnouczelniany, z innego kierunku) <b>z danego kierunku</b>
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki <b>nauki techniczne</b>		Podział ECTS (liczba i %) <b>2 100%</b>
<b>Odpowiedzialny za przedmiot / wykładowca:</b>		
<p>dr inż. Szymon Szczęsny email: <a href="mailto:szymon.szczesny@put.poznan.pl">szymon.szczesny@put.poznan.pl</a> tel. 61 6652297 Katedra Inżynierii Komputerowej ul. Piotrowo 3a, 61-138 Poznań</p>		
<b>Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:</b>		
1	<b>Wiedza:</b>	Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę w zakresie 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.
2	<b>Umiejętności:</b>	Wymagana jest również umiejętność programowania oraz znajomość języków opisu sprzętu. Student powinien również posiadać elementarną wiedzę na temat technologii rekonfigurowalnych, np. na przykład obwodów programowalnej logiki typu FPGA oraz na temat układów programowalnych typu mikrokontrolery.
3	<b>Kompetencje społeczne</b>	Ponadto powinien prezentować postawę zainteresowania pogłębianiem posiadanej wiedzy w obszarze systemów wbudowanych, ze szczególnym uwzględnieniem metodologii tworzenia oprogramowania dla urządzeń programowalnych, hybrydowych i mobilnych.
<b>Cel przedmiotu:</b>		
<ol style="list-style-type: none"> <li>Poszerzenie wiedzy teoretycznej w zakresie metodologii rozwijania oprogramowania dla systemów wbudowanych, przekazanie informacji o głównych problemach dziedziny, metod i strategii projektowania i modelowania oprogramowania dla systemów wbudowanych wykorzystywanymi w przemysłowych rozwiązaniach</li> <li>Zaznajomienie z literaturą poświęconą projektowaniu, rozwijaniu i użytkowaniu mikrosystemów</li> <li>Przedstawienie różnic między tworzeniem oprogramowania dla systemów komputerowych a systemów wbudowanych z omówieniem poszczególnej klasy systemów</li> <li>Pokazanie znaczenia obranych metod w zakresie rozwijania i testowania oprogramowania, scharakteryzowanie narzędzi wykorzystywanych w procesie rozwijania oprogramowania dla mikrosystemów</li> <li>Zapoznanie z ograniczeniami w rozwoju oprogramowania, z zagadnieniami dotyczącymi bezpieczeństwa oraz regułami użytkowania licencji systemów wbudowanych</li> <li>Zachęcenie do zgłębiania wiedzy w zakresie rozwijania oprogramowania dla mikrosystemów i poznawania architektury istniejących rozwiązań komercyjnych</li> <li>Omówienie specyfiki rynku systemów wbudowanych, głównych trendów jego rozwoju, a także rynku pracy we wskazanej tematyce w skali kraju i świata.</li> <li>Uświadomienie ograniczeń prawnych, ekonomicznych i technicznych w rozwoju dziedziny</li> </ol>		
<b>Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia</b>		
<b>Wiedza:</b>		

<ol style="list-style-type: none"><li>1. ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki - [K_W5]</li><li>2. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce i w wybranych pokrewnych dyscyplinach naukowych - [K_W6]</li><li>3. ma podstawową wiedzę o cyklu życia systemów informatycznych sprzętowych lub programowych - [K_W7]</li><li>4. ma wiedzę nt. kodeksów etycznych dotyczących informatyki, rozumie zagrożenia związane z przestępczością elektroniczną, rozumie specyfikę systemów krytycznych ze względu na bezpieczeństwo (ang. mission-critical systems) - [K_W9]</li><li>5. ma podstawową wiedzę dotyczącą zarządzania i prowadzenia działalności gospodarczej - [K_W11]</li><li>6. ma podstawową wiedzę dotyczącą zarządzania jakością, w tym podstawową wiedzę nt. standardów serii ISO 9000 - [K_W12]</li><li>7. zna ogólne zasady tworzenia i rozwoju form indywidualnej przedsiębiorczości - [K_W14]</li><li>8. ma podstawową wiedzę dotyczącą transferu technologii w odniesieniu do rozwiązań informatycznych - [K_W15]</li></ol>
<b>Umiejętności:</b>
<ol style="list-style-type: none"><li>1. potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł (w języku ojczystym i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie - [K_U1]</li><li>2. potrafi określić kierunki dalszego uczenia się i zrealizować proces samokształcenia - [K_U5]</li><li>3. potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne - [K_U9]</li><li>4. 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 - [K_U10]</li><li>5. potrafi sformułować specyfikację funkcjonalną w formie przypadków użycia - [K_U22]</li></ol>
<b>Kompetencje społeczne:</b>
<ol style="list-style-type: none"><li>1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]</li><li>2. zna przykłady i rozumie przyczyny wadliwie działających systemów informatycznych, które doprowadziły do poważnych strat finansowych, społecznych lub też do poważnej utraty zdrowia, a nawet życia - [K_K4]</li><li>3. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]</li></ol>

<b>Sposoby sprawdzenia efektów kształcenia</b>
Ocena formująca: <ul style="list-style-type: none"><li>- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,</li><li>- na podstawie prezentacji rozwiązania problemów zadanych na poprzednich wykładach</li></ul> Ocena podsumowująca: <p>Weryfikowanie założonych efektów kształcenia realizowane jest przez:</p> <ul style="list-style-type: none"><li>- ocenę wiedzy i umiejętności wykazanych na sprawdzianie pisemnym o charakterze problemowym</li><li>- omówienie wyników sprawdzianu zaliczeniowego,</li></ul> <p>Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:</p> <ul style="list-style-type: none"><li>- omówienia dodatkowych aspektów zagadnienia,</li><li>- uwagi związane z udoskonaleniem materiałów dydaktycznych,</li><li>- wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.</li></ul>
<b>Treści programowe</b>

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 funkcjonalnymi i ekonomicznymi aspektami; 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, demonstracja, pogadanka

#### Literatura podstawowa:

1. The industrial electronics handbook Wilanowski B, Irwin D., Taylor & Francis, 2011
2. Projektowanie układów cyfrowych z wykorzystaniem języka VHDL, Mark Zwoliński.
3. Wbudowane systemy mikroprocesorowe, Aleksander Timofiejew, Siedlce: Wydawnictwo Akademii Podlaskiej, ISBN: 978-83-7051-579-9, 2010.
4. Real-Time Systems Design and Analysis: Tools for the Practitioner, P. A. Laplante, S. J. Ovaska, Wiley, 2012
5. Intel Galileo Gen 2 and Intel Edison for Beginners, Manoel Carlos Ramon , Springer, Berlin, 2015

#### Literatura uzupełniająca:

1. Programowalne moduły Ethernetowe w przykładach, Chruściel, M. BTC, Wa-wa 2013
2. Linux. Podstawy i aplikacje dla systemów embedded Skalski Ł., BTC, Wa-wa 2012
3. Linux w systemach embedded, Marcin Bis, Wydawnictwo BTC, ISBN: 978-83-60233-74-0, 2011.
4. Building Embedded Linux Systems, Karim Yaghmour, Jon Masters, Gilad Ben-Yossef, Philippe Gerum, O'Reilly Media, ISBN-10: 0596529686, 2008.

#### Bilans nakładu pracy przeciętnego studenta

Czynność	Czas (godz.)	
1. udział w wykładach	15	
2. udział w konsultacjach związanych z realizacją procesu kształcenia (częściowo realizowane drogą elektroniczną)	2 10	
3. rozwiązywanie problemów podanych na wykładach	10	
4. zapoznanie się ze wskazaną literaturą / materiałami dydaktycznymi (10 stron tekstu naukowego = 1 godz.), 150 stron	15	
5. przygotowanie do zaliczenia wykładów		
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	52	2
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	17	1
Zajęcia o charakterze praktycznym	0	0