

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Architektura systemów wbudowanych		Kod 1010542311010540034
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 1 / 1
Ścieżka obieralności/specjalność Mikrosystemy informatyczne	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obligatoryjny
Stopień studiów: II stopień	Forma studiów (stacjonarna/niestacjonarna) stacjonarna	
Godziny Wykłady: 30 Ćwiczenia: - Laboratoria: 30 Projekty/seminaria: -		Liczba punktów 6
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) kierunkowy		(ogólnouczelniany, z innego kierunku) z danego kierunku
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne		Podział ECTS (liczba i %) 6 100%
Odpowiedzialny za przedmiot / wykładowca:		
<p>dr inż. Adam Turkot email: adam.turkot@put.poznan.pl tel. 61 6652284 Katedra Inżynierii Komputerowej ul. Piotrowo 3a, 61-138 Poznań</p>		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	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.
2	Umiejętności:	Powinien również rozumieć konieczność poszerzania swoich kompetencji / mieć gotowość do podjęcia współpracy w ramach zespołu.
3	Kompetencje społeczne	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:		
<ol style="list-style-type: none"> Zaznajomienie studentów z metodologią projektowania architektury systemów wbudowanych Przekazanie studentom poszerzonej wiedzy z zakresu architektur systemów wbudowanych ? single-procesor, multi-procesor, 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 realizacje elementów projektu i połączenie ich w całość. 		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		

1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie architektury systemów komputerowych i systemów wbudowanych - [K_W4]
2. ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki; - [K_W5]
3. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce i w wybranych pokrewnych dyscyplinach naukowych - [K_W6]
4. ma podstawową wiedzę o cyklu życia systemów informatycznych sprzętowych lub programowych - [K_W7]
5. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z wybranego obszaru informatyki - [K_W8]
6. zna podstawowe pojęcia z zakresu ekonomii odnoszące się do inwestycji informatycznych i projektów informatycznych takie, jak zwrot z inwestycji, koszty stałe i koszty zmienne, ryzyko finansowe, przychód a zysk, zysk a przepływy pieniężne (ang. cash flow) - [K_W10]
7. ma podstawową wiedzę dotyczącą transferu technologii w odniesieniu do rozwiązań informatycznych - [K_W15]

Umiejętności:

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]
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 - [K_U10]
3. potrafi przeprowadzić analizę ryzyka związanego z przedsięwzięciem informatycznym - [K_U11]
4. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K_U13]
5. ma przygotowanie niezbędne do pracy w środowisku przemysłowym oraz zna zasady bezpieczeństwa związane z pracą informatyka - [K_U14]
6. potrafi poprawnie użyć przynajmniej jedną metodę szacowania pracochłonności wytwarzania oprogramowania - [K_U15]
7. potrafi ocenić architekturę oprogramowania z punktu widzenia wymagań pozafunkcyjnych - [K_U18]
8. potrafi sformułować specyfikację funkcjonalną w formie przypadków użycia - [K_U22]
9. potrafi sformułować wymagania pozafunkcyjne dla wybranych charakterystyk jakościowych - [K_U23]
10. 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; - [K_U24]
11. potrafi - zgodnie z zadaną specyfikacją, uwzględniającą aspekty pozatechniczne - zaprojektować złożone urządzenie, system informatyczny lub proces oraz zrealizować ten projekt ? co najmniej w części ? używając właściwych metod, technik i narzędzi, w tym przystosowując do tego celu istniejące lub opracowując nowe narzędzia - [K_U27]

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]
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]
3. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]
4. prawidłowo identyfikuje i rozstrzyga dylematy związane z wykonywaniem zawodu - [K_K7]
5. potrafi myśleć i działać w sposób przedsiębiorczy - [K_K8]

Sposoby sprawdzenia efektów kształcenia

<p>Ocena formująca:</p> <p>a) w zakresie wykładów:</p> <ul style="list-style-type: none">- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach, <p>b) w zakresie laboratoriów / ćwiczeń:</p> <ul style="list-style-type: none">- na podstawie oceny bieżącego postępu realizacji zadań, <p>Ocena podsumowująca:</p> <p>a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:</p> <ul style="list-style-type: none">- przeprowadzenie egzaminu końcowego w formie testu. <p>b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:</p> <ul style="list-style-type: none">- ocenę przygotowania studenta do poszczególnych sesji zajęć laboratoryjnych poprzez sprawdzenie przygotowania zadanych projektów/ćwiczeń oraz ocenę umiejętności związanych z realizacją ćwiczeń laboratoryjnych,- ocenianie ciągle, na każdych zajęciach (odpowiedzi ustne) ? premiowanie przyrostu umiejętności posługiwania się poznanymi zasadami i metodami,- ocenę dokumentacji tworzonej systematycznie wraz z postępami prac projektowych; dokumentacja przygotowywana 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, <p>Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:</p> <ul style="list-style-type: none">- omówienia dodatkowych aspektów zagadnienia,- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,- umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium,- uwagi związane z udoskonaleniem materiałów dydaktycznych,- wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.
Treści programowe
<p>Podstawy architektury systemów wbudowanych . Techniki efektywnego wykorzystania zasobów sprzętowych . Ocena możliwości sprzętowych . Środowisko programistyczne. Techniki programowania. Metody optymalizacji kodu . Interfejs użytkownika</p> <p>Rozwiązania sprzętowe i programistyczne umożliwiające zarządzanie poborem mocy.</p> <p>Technik zabezpieczeń oprogramowania (integralność programu , odporność na nieautoryzowanym kopiowaniem)</p> <p>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.</p> <p>Protokoły komunikacji wykorzystywane w chmurze.</p> <p>Zwiększania niezawodności systemów bezobsługowych , techniki zapewniające gospodarkę energetyczną systemów autonomicznych.</p> <p>Zajęcia laboratoryjne prowadzone są w formie piętnastu 2-godzinnych ćwiczeń, odbywających się w laboratorium, poprzedzonych 2-godzinną sesją instruktazową na początku semestru. Ćwiczenia realizowane są przez 2-osobowe zespoły studentów.</p> <p>Laboratoria obejmują:</p> <p>Wprowadzenie do programowania Intel Galileo.</p> <p>Instalacja systemu Windows i sterowników na platformie Intel Galileo.</p> <p>Intel IoT Analytics w systemach wbudowanych.</p> <p>Wykorzystanie serwera WWW jako interfejsu komunikacji z systemem wbudowanym</p> <p>Wykorzystanie analizatora stanów logicznych w konstruowaniu i testowaniu systemów wbudowanych.</p> <p>Instalacja Yocto OS na karcie SD. Testowanie interfejsów Intel Galileo. Obsługa wyświetlacza LCD w skryptach Python.</p> <p>Konstrukcja woltomierza.</p> <p>Instalacja Debian OS na karcie SD. Obsługa zdarzeń cyklicznych CRON. Audio streaming z wykorzystaniem GStreamer i protokołu UDP.</p> <p>Komunikacja w chmurze ? (autoryzacja OAuth2, JSON, Google Drive). Programowanie Raspberry Pi w C i Python.</p> <p>Programowanie sterowników ? w systemach linuxowych, oraz systemach czasu rzeczywistego.</p> <p>Metody dydaktyczne:</p> <ol style="list-style-type: none">1. wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, prezentacje wybranych rozwiązań studenckich.2. ćwiczenia laboratoryjne: ćwiczenia praktyczne, wykonywanie eksperymentów, dyskusja, praca w zespole,

Literatura podstawowa:		
1. Raspberry Pi Cookbook, Simon Monk Helion 2015		
2. Intel Galileo Gen 2 and Intel Edison for Beginners, Manoel Carlos Ramon , Springer, Berlin, 2015		
Literatura uzupełniająca:		
Bilans nakładu pracy przeciętnego studenta		
Czynność	Czas (godz.)	
1. udział w zajęciach laboratoryjnych / ćwiczeniach	30	
2. dokończenie (w ramach pracy własnej) sprawozdań z ćwiczeń laboratoryjnych.	15	
3. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności ćwiczeń laboratoryjnych / projektu (częściowo realizowane drogą elektroniczną)	8	
4. napisanie programu / programów, uruchomienie i weryfikacja (czas poza zajęciami laboratoryjnymi)	20	
5. przygotowanie do sprawdzianów / kolokwium, udział w sprawdzianie (3+2godz)	30	
6. udział w wykładach	10	
7. zapoznanie się ze wskazaną literaturą / materiałami dydaktycznymi (10 stron tekstu naukowego = 1 godz.), 100 stron	10	
8. przygotowanie i udział w egzaminie (8+2 godz.)		
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	143	6
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	70	3
Zajęcia o charakterze praktycznym	50	2