

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Programowanie mikrokontrolerów jednocukładowych		Kod 1010542321010546916
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 1 / 2
Ścieżka obieralności/specjalność Mikrosystemy informatyczne	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obieralny
Stopień studiów: II stopień	Forma studiów (stacjonarna/niestacjonarna) stacjonarna	
Godziny Wykłady: 30 Ćwiczenia: - Laboratoria: 30 Projekty/seminaria: -		Liczba punktów 4
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) (ogólnouczelniany, z innego kierunku) kierunkowy z danego kierunku		
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne		Podział ECTS (liczba i %) 4 100%
Odpowiedzialny za przedmiot / wykładowca: dr inż. Piotr Katarzyński email: piotr.katarzynski@put.poznan.pl tel. 61 6652592 Katedra Inżynierii Komputerowej ul. Piotrowo 3a, 61-138 Poznań		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu systemów mikroprocesorowych, oraz architektury komputerów.
2	Umiejętności:	Powinien posiadać umiejętność programowania w języku C.
3	Kompetencje społeczne	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:		
<ol style="list-style-type: none"> Przekazanie studentom poszerzonej wiedzy z zakresu projektowania aplikacji wbudowanych na mikrokontrolerach jednocukładowych popularnych rodzin. Omówienie typowych urządzeń peryferyjnych spotykanych w mikrokontrolerach ze specyfiką ich pracy, konfigurowania i stosowania. Wykształcenie umiejętności doboru optymalnej, dla danego zadania realizacji aplikacji z użyciem mikrokontrolera, z obsługą dedykowanych modułów peryferyjnych i przy uwzględnieniu wymogów związanych z oszczędnością energii i wydajnością obliczeniową. Rozwijanie u studentów umiejętności rozwiązywania problemów doboru kontrolera i wdrożenia do realizacji typowych zadań z zakresu pobierania przetwarzania sygnałów, komunikacji z użytkownikiem bądź sterowania Kształtowanie u studentów umiejętności pracy zespołowej poprzez realizację elementów projektu i połączenie ich w całość. 		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		
<ol style="list-style-type: none"> ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki; - [K_W5] ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce i w wybranych pokrewnych dyscyplinach naukowych; - [K_W6] zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z wybranego obszaru informatyki. - [K_W8] 		
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 określić kierunki dalszego uczenia się i zrealizować proces samokształcenia; - [K_U5]
3. potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne - [K_U9]
4. potrafi formułować i testować hipotezy związane z problemami inżynierskimi i prostymi problemami badawczymi - [K_U12]
5. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K_U13]
6. 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]
Kompetencje społeczne:
1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]
2. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]

Sposoby sprawdzenia efektów kształcenia
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: - przeprowadzenie na wykładach w trakcie semestru dwóch lub trzech krótkich kartkówek. 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 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, 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, - uwagi związane z udoskonaleniem materiałów dydaktycznych,
Treści programowe

Pojęcia podstawowe w opisie architektury mikrokontrolerów: rejestr, przestrzeń adresowa, urządzenia wejścia/wyjścia, architektura harwardzka, cykl rozkazowy, DMA, przerwania, pamięć programu, sekcja bootloadera. Urządzenia kontrolno-pomiarowe: mikrokontroler, mikroprocesor, ASIC, FPGA. Przegląd producentów i rodzin 8-bitowych oraz 32-bitowych kontrolerów. Mikrokontrolery 8-bitowe z rodziny AVR, obszary zastosowań, architektura, realizacja portów wejścia/wyjścia. Sposoby integrowania kontrolerów z urządzeniami wejścia/wyjścia (obsługa klawiatur, wyświetlaczy, układów regulowanych dwustawnie). Przetwarzanie analogowo cyfrowe z użyciem mikrokontrolerów. Przetwarzanie cyfrowo-analogowe z wykorzystaniem mikrokontrolerów. Niezawodność przetwarzania (układ Watchdog, układ Brown-out, Black out). Układy peryferyjne ? timery liczniki, układy komunikacji z peryferiami I2C oraz SPI. Układy interfejsów komunikacyjnych: szeregowy, Modbus, CAN, Ethernet. Interfejs JTAG. Organizacja kodu aplikacji, bootloader i jego zastosowania. Kontrolery 32-bitowe na przykładzie STM3 z rdzeniem M3, M4 lub M7. Obsługa wyświetlacza, urządzeń peryferyjnych. Omówienie otwartych systemów operacyjnych na architektury Cortex (eCos FreeRTOS).

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

Laboratoria obejmują:

Projektowanie prostych aplikacji z użyciem kontrolerów 8-bitowych oraz 32-bitowych na dedykowanych płytach ćwiczebnych. Aplikacje ukazują przypadki użycia omawianych na wykładach układów peryferyjnych, urządzeń wejścia/wyjścia, modułów komunikacyjnych. Układy kontrolerów współpracują z typowymi podzespołami komunikacji człowiek-maszyna, jednostkami pomiarowymi (np. sensory warunków środowiskowych) oraz elementami wykonawczymi automatyki.

Śledzenie wykonania aplikacji za pośrednictwem interfejsów typu JTAG. Narzędzia opensource w programowaniu systemów wbudowanych. Przygotowanie narzędzi (toolchain) Programowanie i analiza wykonania programu z wykorzystaniem dostępnych narzędzi. Implementowanie aplikacji typu bare-metal do realizacji zadanej funkcjonalności. Implementowanie aplikacji na wybrany system operacyjny (np. eCos lub freeRTOS).

Metody dydaktyczne:

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. Mikrokontrolery ATmega w praktyce, R. Baranowski, BTC, Wa-wa, 2005
2. Mikrokontrolery STM32 w praktyce, K. Paprocki BTC, Wa-wa 2009
3. Noty katalogowe wskazanych modeli kontrolerów firm Atmel oraz ST Microelectronics

Literatura uzupełniająca:

1. Wybrane artykuły z bazy wiedzy www.atmel.com ukazujące przypadki aplikowania kontrolerów.
2. Wybrane artykuły z bazy wiedzy www.st.com ukazujące przypadki aplikowania kontrolerów.

Bilans nakładu pracy przeciętnego studenta

Czynność	Czas (godz.)
1. udział w zajęciach laboratoryjnych / ćwiczeniach	30
2. przygotowanie do ćwiczeń laboratoryjnych	10
3. dokończenie (w ramach pracy własnej) sprawozdań z ćwiczeń laboratoryjnych.	5
4. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności ćwiczeń laboratoryjnych / projektu	5
5. napisanie programu / programów, uruchomienie i weryfikacja (czas poza zajęciami laboratoryjnymi)	10
6. przygotowanie do sprawdzianów / kolokwium	5
7. udział w wykładach	30
8. zapoznanie się ze wskazaną literaturą / materiałami dydaktycznymi (10 stron tekstu naukowego = 1 godz.),	5

Obciążenie pracą studenta

forma aktywności	godzin	ECTS
Łączny nakład pracy	100	4
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	65	3
Zajęcia o charakterze praktycznym	60	2