



IV. Opis programu studiów

3. KARTA PRZEDMIOTU

Kod przedmiotu	E-TD-08-s2
Nazwa przedmiotu	Technika mikroprocesorowa
Nazwa przedmiotu w języku angielskim	Microprocessor Technology
Obowiązuje od roku akademickiego	2020/21

USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	Teleinformatyka
Poziom kształcenia	I stopień
Profil studiów	Praktyczny
Forma i tryb prowadzenia studiów	Studia stacjonarne
Zakres	
Jednostka prowadząca przedmiot	Katedra Informatyki, Elektroniki i Elektrotechniki
Koordynator przedmiotu	dr inż. Remigiusz Baran
Zatwierdził	Dziekan Wydziału Elektrotechniki Automatyki i Informatyki Dr hab. inż. Antoni Różowicz, prof. PŚk

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów*	Przedmiot podstawowy
Status przedmiotu*	Obowiązkowy
Język prowadzenia zajęć	Polski
Usytuowanie modułu w planie studiów - semestr	Semestr 2
Wymagania wstępne	Podstawy techniki cyfrowej
Egzamin (TAK/NIE)	Nie
Liczba punktów ECTS	2

*pozostawić właściwe

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	15	0	15	0	0

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Posiada podstawową znajomość techniki mikroprocesorowej, organizacji jednostki centralnej, jej rodzajów oraz budowy i zasad funkcjonowania mikroprocesora.	TI1_W03
	W02	Zna maszynową reprezentację danych i zasady realizacji operacji arytmetycznych, strukturę i składnię asemblera, podstawowe techniki programowania z użyciem asemblera.	TI1_W03
	W03	Zna znaczenie i organizację systemu przerwań procesora, organizacji pamięci oraz urządzeń peryferyjnych. Zna zasady organizacji systemu mikroprocesorowego.	TI1_W03 TI1_W04
	W04	Zna zasady i metody tworzenia projektów w oparciu o kod wysokopoziomowy z wykorzystaniem procedur w języku asemblerowym.	TI1_W03 TI1_W04 TI1_W06
Umiejętności	U01	Potrafi zapisać w asemblerze elementarne składniki programu, w tym proste procedury arytmetyczne.	TI1_U07
	U02	Potrafi korzystać z narzędzi rozwojowych, zarówno programistycznych jak i sprzętowych, w projektowaniu zadań dla mikrokontrolerów, łącząc przy tym zalety języków niskiego i wysokiego poziomu.	TI1_U07 TI1_U11
	U03	Potrafi krytycznie ocenić organizację prostego systemu mikroprocesorowego.	TI1_U14
Kompetencje społeczne	K01	Rozumie procesy i czynniki określające kierunki i dynamikę rozwoju techniki cyfrowej, zwłaszcza w zakresie rozwoju mikroprocesorów.	TI1_K01

TRZĘŚCI PROGRAMOWE

Forma zajęć*	Treści programowe
wykład	1. Organizacja jednostki centralnej procesora w tym rola i znaczenie jej podstawowych komponentów: jednostki arytmetycznej, rejestrów wewnętrznej pamięci danych, rejestru rozkazów, bloku dekodowania i układu sterowania, oraz magistral danych, adresowej i sterującej.
	2. Reprezentacja binarna danych oraz podstawowe operacje arytmetyczne i logiczne. Zasady funkcjonowania jednostki arytmetyczno-logicznej i znaczenie bitów słowa stanu procesora.
	3. Pamięci RAM i ROM. Organizacja i znaczenie bloków pamięci danych i programu w odniesieniu do typowych realizacji maszyny von Neumanna (architektury Harvard i Princeton) oraz organizacji wewnętrznej i zewnętrznej przestrzeni adresowej mikroprocesora.
	4. Cykl rozkazowy procesora. Architektury typu CISC i RISC – przykłady organizacji. Przetwarzanie potokowe.
	5. Urządzenia peryferyjne jako elementy kompletnego systemu mikroprocesorowego: układy czasowo-licznikowe, porty komunikacji szeregowej, przetworniki A/C i C/A i inne). Zasady organizacji systemu mikroprocesorowego. Architektura i specyfikacja parametrów technicznych wybranego mikrokontrolera. Organizacja jego pamięci danych i pamięci programu.
	6. Składnia języka asemblerowego na przykładzie wybranego mikrokontrolera. Zasady tworzenia oprogramowania niskopoziomowego. Tryby adresowania.
	7. Zasady tworzenia oprogramowania wysokopoziomowego dla mikrokontrolerów oraz łączenia kodu w języku wysokiego poziomu z procedurami w asemblerze. Przykłady kodu.
laboratorium	1. Programistyczne zintegrowane środowiska IDE i sprzętowe narzędzia rozwojowe dla mikrokontrolerów oraz proces tworzenia, budowania, debugowania i uruchamiania projektu.

	2. Tworzenie, symulowanie i uruchamianie przykładowego projektu w assemblerze w środowisku IDE dla wybranego mikrokontrolera.
	3. Implementacja i uruchomienie na płycie rozwojowej projektu w assemblerze z zakresu kontrolowania procesów w czasie rzeczywistym przy użyciu procedur opóźniających.
	4. Implementacja i uruchomienie na płycie rozwojowej projektu w assemblerze z zakresu kontrolowania procesów w czasie rzeczywistym przy użyciu układu czasowolicznikowego i systemu przerwań wybranego mikrokontrolera.
	5. Implementacja i uruchamianie na płycie rozwojowej projektu w języku wysokiego poziomu z zakresu obsługi przerwania z innego, wybranego urządzenia peryferyjnego mikrokontrolera.

*) zostawić tylko realizowane formy zajęć

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów uczenia się					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01			X		X	
W02			X		X	
W03			X		X	
W04			X		X	
U01					X	
U02					X	
U03					X	
K01					X	

FORMA I WARUNKI ZALICZENIA

Forma zajęć*	Forma zaliczenia*	Warunki zaliczenia
Wykład	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów z kolokwiów w trakcie zajęć.
Laboratorium	zaliczenie z oceną/zaliczenie	Uzyskanie co najmniej oceny dostatecznej ze sprawozdań z zajęć.

*) zostawić tylko realizowane formy zajęć oraz wybrać formę zaliczenia

NAKŁAD PRACY STUDENTA

Bilans punktów ECTS							
Lp.	Rodzaj aktywności	Obciążenie studenta					Jednostka
1.	Udział w zajęciach zgodnie z planem studiów	W	C	L	P	S	h
		15	-	15	-	-	
3.	Inne (konsultacje, egzamin)*	2	-	2	-	-	h
4.	Razem przy bezpośrednim udziale nauczyciela akademickiego	34					h
5.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	1,36					ECTS
6.	Liczba godzin samodzielnej pracy studenta	16					h
7.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	0,64					ECTS
8.	Nakład pracy związany z zajęciami o charakterze praktycznym	15					h
9.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	0,60					ECTS
10.	Sumaryczne godzinowe obciążenie pracą studenta	50					h
11.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	2					

* wszelkie formy weryfikacji efektów, w tym egzaminy oraz nie więcej niż 2 godziny konsultacji dla każdej formy zajęć

LITERATURA

1. Stallings W.: Organizacja i architektura systemu komputerowego, WNT, 2004
2. Pełka R.: Mikrokontrolery, architektura, programowanie, zastosowania, WKŁ, Warszawa, 2001
3. Jakubiec J.: Podstawy techniki mikroprocesorowej, Wydawnictwo Politechniki Śląskiej, 2014
4. Jabłoński T., Pławsiuk K.: Programowanie mikrokontrolerów PIC w języku C, BTC, 2005
5. Kardaś M.: Język C. Pasja programowania mikrokontrolerów 8-bitowych, Atnel, 2014
6. Peatman J.B.: Design with PIC Microcontrollers, Prentice Hall, 1997
7. Bates M.: Programming 8-bit PIC Microcontrollers in C, Elsevier Science & Technology, 2008
8. Ram B.: Fundamentals of Microprocessors and Microcontrollers 8/E PB, DHANPAT RAI PUBLICATIONS (P) LTD.-NEW DELHI, 2012

Uwaga: wykaz literatury winien uwzględniać aktualne i dostępne publikacje