

**KARTA PRZEDMIOTU**

Kod przedmiotu	studia stacjonarne:	<b>M#2-S2-AiR-KSSiP-211</b>
	studia niestacjonarne:	<b>M#2-N2-AiR-KSSiP-211</b>
Nazwa przedmiotu	<b>Systemy mikroprocesorowe w technice pomiarowej</b>	
Nazwa przedmiotu w języku angielskim	<b>Microprocessor-based measurement</b>	
Obowiązuje od roku akademickiego	<b>2024/2025</b>	

**USYTUOWANIE PRZEDMIOTU W SYSTEMIE STUDIÓW**

Kierunek studiów	<b>Automatyka i Robotyka</b>
Poziom kształcenia	<b>II stopień</b>
Profil studiów	<b>Ogólnoakademicki</b>
Forma i tryb prowadzenia studiów	<b>Studia stacjonarne i niestacjonarne</b>
Zakres	<b>Komputerowe Systemy Sterowania i Pomiarów</b>
Jednostka prowadząca przedmiot	<b>Katedra Automatyki i Robotyki</b>
Koordynator przedmiotu	<b>dr inż. Dawid Pietrala</b>
Zatwierdził	<b>dr hab. Jakub Takosoglu prof. PŚk, Dziekan Wydziału Mechatroniki i Budowy Maszyn</b>

**OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU**

Przynależność do grupy/bloku przedmiotów	<b>Przedmiot specjalnościowy</b>	
Status przedmiotu	<b>Obowiązkowy</b>	
Język prowadzenia zajęć	<b>Polski</b>	
Usytuowanie w planie studiów - semestr	studia stacjonarne	<b>Semestr II</b>
	studia niestacjonarne	<b>Semestr II</b>
Wymagania wstępne		
Egzamin (TAK/NIE)	<b>Nie</b>	
Liczba punktów ECTS	<b>2</b>	

Forma prowadzenia zajęć		wykład	ćwiczenia	laboratorium	projekt	inne
Liczba godzin w semestrze	studia stacjonarne:				<b>30</b>	
	studia niestacjonarne:				<b>18</b>	



**EFEKTY UCZENIA SIĘ**

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Posiada szczegółową wiedzę na temat metodyki i technik programowania w kontekście systemów mikroprocesorowych wykorzystywanych w technice pomiarowej, w tym różnych architektur sprzętowych i środowisk programistycznych.	AiR2_W08
	W02	Ma uporządkowaną wiedzę w zakresie budowy, działania oraz programowania mikrokontrolerów i komputerów jednoukładowych, a także ich zastosowania w systemach pomiarowych, uwzględniając integrację z układami peryferyjnymi, takimi jak czujniki, moduły komunikacyjne, układy przetwarzania sygnałów i elementy wykonawcze.	AiR2_W12
Umiejętności	U01	Potrafi stosować zaawansowane środowiska programistyczne oraz symulatory do projektowania, testowania i debugowania oprogramowania dla systemów mikroprocesorowych, z uwzględnieniem specyficznych wymagań techniki pomiarowej, takich jak przetwarzanie danych z czujników i implementacja algorytmów kalibracyjnych.	AiR2_U07
	U02	Potrafi projektować i analizować systemy mikroprocesorowe wykorzystywane w technice pomiarowej, w tym układy komunikacji cyfrowej (np. UART, SPI, I2C) oraz implementację protokołów wymiany danych w systemach pomiarowych.	AiR2_U11
Kompetencje społeczne	K01	Jest świadomy potrzeby konsultacji z ekspertami oraz korzystania z literatury fachowej i dokumentacji technicznej w celu rozwiązywania problemów związanych z projektowaniem systemów mikroprocesorowych w technice pomiarowej.	AiR2_K01
	K02	Rozumie znaczenie etycznego podejścia do projektowania i wdrażania systemów mikroprocesorowych w technice pomiarowej, zwłaszcza w kontekście ich niezawodności, bezpieczeństwa oraz wpływu na dokładność i wiarygodność wyników pomiarów.	AiR2_K03



**TREŚCI PROGRAMOWE**

Forma zajęć	Treści programowe
projekt	<p>Konfiguracja i rozwój projektu z wykorzystaniem środowiska Keil uVision. Przygotowanie środowiska programistycznego Keil uVision do realizacji projektu, w tym konfiguracja i instalacja niezbędnych narzędzi. Tworzenie i konfiguracja nowego projektu w IDE z uwzględnieniem wymagań projektu. Praca z mikrokontrolerem i podstawowymi funkcjonalnościami. Analiza trybów pracy pinów mikrokontrolera oraz implementacja sterowania wyjściami cyfrowymi. Projektowanie i realizacja sterowania linią diod LED przy użyciu wyjść cyfrowych. Modularne podejście do realizacji projektu. Projektowanie struktury kodu w sposób modularny – podział projektu na niezależne moduły odpowiedzialne za różne funkcjonalności systemu. Zaawansowana obsługa przerwań. Konfiguracja i obsługa timera SysTick w celu odmierzenia czasu. Implementacja przerwań timera SysTick do precyzyjnego zarządzania czasem w projekcie. Obsługa przycisków w trybie oczekiwania przy wykorzystaniu wejść cyfrowych. Integracja komponentów peryferyjnych. Projektowanie sterowania wyświetlaczem segmentowym w trybie statycznym oraz dynamicznym. Praktyczna integracja wyświetlacza segmentowego z mikrokontrolerem. Implementacja sterowania wyświetlaczem LCD przy użyciu przygotowanej biblioteki i wyjść cyfrowych. Wprowadzenie do obsługi podstawowych interfejsów komunikacyjnych, w tym UART. Implementacja wysyłania i odbierania danych przez UART w trybie oczekiwania oraz zaawansowana obsługa interfejsu z wykorzystaniem przerwań. Projektowanie mechanizmów przerwań w celu zarządzania bezczynnością, przechwytywaniem znaków i zakończeniem transmisji. Poznanie działania kontrolera DMA i jego integracji z interfejsem UART. Realizacja projektu wykorzystującego przerwanie kontrolera DMA do obsługi transmisji danych. Implementacja protokołu MODBUS RTU w dwóch etapach, obejmujących teorię i praktyczną implementację komunikacji. Wszystkie zadania realizowane w ramach projektu uwzględniają zastosowanie technik inżynierskich oraz dobrą organizację pracy zespołowej. Efektem końcowym jest kompletny i funkcjonalny system, który spełnia założone wymagania projektowe.</p>

**METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ**

Symbol efektu	Metody sprawdzania efektów kształcenia (zaznaczyć X)					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01				X		
W02				X		
U01				X		
U02				X		
K01						X
K02						X

**FORMA I WARUNKI ZALICZENIA**

Forma zajęć	Forma zaliczenia	Warunki zaliczenia
projekt	zaliczenie z oceną	Przygotowanie projektu i pozytywna jego obrona.

**NAKŁAD PRACY STUDENTA**

Bilans punktów ECTS														
Lp.	Rodzaj aktywności	Obciążenie studenta										Jednostka		
		studia stacjonarne					studia niestacjonarne							
		W	C	L	P	S	W	C	L	P	S			
1.	Udział w zajęciach zgodnie z planem studiów				30							18		h
2.	Inne (konsultacje, egzamin)				2							2		h
3.	<b>Razem przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>32</b>					<b>20</b>					h		
4.	<b>Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego</b>	<b>1,3</b>					<b>0,8</b>					ECTS		
5.	<b>Liczba godzin samodzielnej pracy studenta</b>	<b>18</b>					<b>30</b>					h		
6.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy</b>	<b>0,7</b>					<b>1,2</b>					ECTS		
7.	<b>Nakład pracy związany z zajęciami o charakterze praktycznym</b>	<b>50</b>					<b>50</b>					h		
8.	<b>Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym</b>	<b>2</b>					<b>2</b>					ECTS		
9.	<b>Sumaryczne obciążenie pracą studenta</b>	<b>50</b>					<b>50</b>					h		
10.	<b>Punkty ECTS za moduł</b> <i>1 punkt ECTS od 25 do 30 godzin obciążenia studenta</i>	<b>2</b>										ECTS		





Fundusze Europejskie  
dla Rozwoju Społecznego



Rzeczpospolita  
Polska

Dofinansowane przez  
Unię Europejską



## LITERATURA

1. Zhu Y.: Embedded Systems with Arm Cortex-M Microcontrollers in Assembly Language and C. E-MAN PR LLC, 2015.
2. Kurczyk A.: Mikrokontrolery STM32 dla początkujących. BTC, 2017.
3. Kaufmann M.: Embedded Systems: ARM Programming and Optimization. 2015.
4. Paprocki K.: Mikrokontrolery STM32 w praktyce. BTC, 2011.
5. Szumski M.: Mikrokontrolery STM32 w systemach sterowania i regulacji. BTC, 2017.



Politechnika Świętokrzyska  
Kielce University of Technology

Projekt „Dostosowanie kształcenia w Politechnice  
Świętokrzyskiej do potrzeb współczesnej gospodarki”  
nr FERS.01.05-IP.08-0234/23



Wydział Mechatroniki  
i Budowy Maszyn