

OPIS PRZEDMIOTU/MODUŁU KSZTAŁCENIA (SYLABUS)

1.	Nazwa przedmiotu/modułu w języku polskim oraz angielskim Pracownia systemów wbudowanych/Embedded System Laboratory
2.	Dyscyplina Informatyka techniczna 2 ECTS, Automatyka, elektronika i elektrotechnika 2 ECTS
3.	Język wykładowy polski
4.	Jednostka prowadząca przedmiot Wydział Fizyki i Astronomii
5.	Kod przedmiotu/modułu 24-ISSP-S1-E6-PSW
6.	Rodzaj przedmiotu/modułu (<i>obowiązkowy lub do wyboru</i>) do wyboru
7.	Kierunek studiów (specjalność/specjalizacja) Informatyka Stosowana i Systemy Pomiarowe
8.	Poziom studiów (<i>I lub II stopień, jednolite studia magisterskie, studia doktoranckie</i>) I stopień
9.	Rok studiów (<i>jeśli obowiązuje</i>) 3
10.	Semestr (<i>zimowy lub letni</i>) letni
11.	Forma zajęć i liczba godzin Metody nauczania Pracowania 45 godzin
12.	Imię, nazwisko, tytuł/stopień naukowy osoby prowadzącej zajęcia Radosław Wasielewski, dr inż., Maciej Zagrabski mgr inż.
13.	Wymagania wstępne w zakresie wiedzy, umiejętności i kompetencji społecznych dla przedmiotu/modułu Znajomość zagadnień omawianych na przedmiocie wstęp do elektroniki oraz elektronice cyfrowej. Wymagana jest umiejętność pisania prostych programów w języku C/C++. Konieczna jest umiejętność korzystania z dokumentacji technicznych napisanych w języku angielskim.
14.	Cele przedmiotu Zapoznanie z architekturą mikrokontrolerów rodziny AVR oraz STM. Poznania zasad programowania mikrokontrolerów za pomocą asemblera oraz języka C. Konfiguracji urządzeń peryferyjnych, takich jak wyświetlaczy LED, LCD, klawiatur,

	ADC, DAC, czujników, magistral komunikacyjnych, portów szeregowych, mikroelektromechnosystemów, czy RTC.
15.	<p>Treści programowe</p> <ol style="list-style-type: none"> 1. Wprowadzenie do programowania mikroprocesorów avr. <ul style="list-style-type: none"> - Budowa procesora, bloki funkcyjne, zestaw instrukcji - Omówienie narzędzi: gcc, ld, as, make - Pierwszy program (miganie diodami), wersja ASM i C - Deasemblowanie kodu źródłowego 2. System przerwań w avr. <ul style="list-style-type: none"> - Przerwania sprzętowe, rodzaje przerwań - Konfiguracja regulatorów czasowych - Obsługa regulatorów czasowych, miganie diodami - Przerwania zewnętrzne, obsługa przycisków, debouncing - Licznik wciśnięcia przycisku - Wyświetlacz 7-segmentowy 3. Protokoły komunikacyjne 1/3. <ul style="list-style-type: none"> - Uniwersalna magistrala szeregową (UART) - Omówienie narzędzi: minicom - Nadawanie danych przed mikroprocesor - Odbieranie danych przez mikroprocesor, buforowanie, analiza 4. Protokoły komunikacyjne 2/3 <ul style="list-style-type: none"> - Magistrala i2c, warstwa fizyczna, warstwa łącza danych - Komunikacja i2c, zegar czasu rzeczywistego - Magistrala 1wire, warstwa fizyczna, warstwa łącza danych - Komunikacja 1wire termometr cyfrowy 5. Protokoły komunikacyjne 3/3 <ul style="list-style-type: none"> - Magistrala 1-wire, warstwa fizyczna, warstwa łącza danych - Komunikacja 1wire termometr cyfrowy - Ethernet, SPI/SDCard 6. Wprowadzenie do programowania mikroprocesorów ARM (STM32F4) <ul style="list-style-type: none"> - Budowa procesora, bloki funkcyjne, zestaw instrukcji - Pierwszy program (miganie diodami), wersja ASM i C - Deasemblowanie kodu źródłowego - System przerwań (różnice z AVR)

	<ul style="list-style-type: none"> - Obsługa regulatorów czasowych - Uniwersalna magistrala szeregowo - Magistrala I2C <p>7. Mikroukłady elektromechaniczne 1/3</p> <ul style="list-style-type: none"> - Zasada działania, elektromechaniczny żyroskop, akcelerometr i mikrofon - Konfigurowanie żyroskopu, pomiary, analiza zebranych danych - Prosty przykład, miganie diodami w zależności od przechyłu <p>8. Mikroukłady elektromechaniczne 2/3</p> <ul style="list-style-type: none"> - Konfigurowanie akcelerometru, omówienie ograniczeń pomiaru - Prosty przykład wykrywanie ruchu - Szacowanie pozycji w przestrzeni na podstawie analizy przyspieszenia, omówienie ograniczeń, akumulowanie błędów prędkości i pozycji <p>9. Mini projekt.</p>	
16.	<p>Zakładane efekty uczenia się</p> <p>Student zna budowę typowych mikrokontrolerów z rodziny AVR i ARM oraz typowe urządzenia peryferyjne.</p> <p>Student potrafi zrealizować proces cross-kompilacji i uruchomić mikrokontroler przeznaczony do pracy w systemie wbudowanym.</p> <p>Student zna podstawowe protokoły komunikacyjne służące do przekazywania danych w systemach wbudowanych.</p> <p>Student zna i wykorzystuje podstawowe mikroukłady elektromechaniczne.</p> <p>Student zna zasady bezpiecznego posługiwania się narzędziami do programowania mikrokontrolerów.</p> <p>Student potrafi napisać program na wybrany mikroprocesor w assemblerze lub C oraz dokonać analizy kodu pod deasemblacją.</p> <p>Student potrafi napisać prosty program dla wybranego mikrokontrolera z wykorzystaniem czujników lub urządzeń elektromechanicznych.</p> <p>Student potrafi tworzyć czytelną i zwięzłą dokumentację działania programu.</p>	<p>Symbole odpowiednich kierunkowych efektów uczenia się:</p> <p>I1_W03, I1_W06, I1_W09,</p> <p>I1_U09, I1_U10, I1_U11, I1_U12, I1_U17,</p> <p>I1_K02, I1_K03, I1_K05, I1_K07</p>
17.	<p>Literatura obowiązkowa i zalecana (<i>źródła, opracowania, podręczniki itp.</i>)</p> <p>Obowiązkowa: Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi: AVR Microcontroller and Embedded Systems: Using Assembly and C</p> <p>Zalecana: Mirosław Kardaś: Mikrokontrolery AVR - język C: podstawy programowania Mirosław Kardaś: Język C : pasja programowania mikrokontrolerów 8-bitowych Tomasz Francuz: Język C dla mikrokontrolerów AVR : od podstaw do zaawansowanych aplikacji Tomasz Francuz: AVR: układy peryferyjne</p>	

	Tomasz Francuz: AVR: praktyczne projekty Dokumentacje techniczne procesorów i makiet	
18.	Metody weryfikacji zakładanych efektów uczenia się: kontrola bieżących postępów podczas zajęć prace kontrolne oraz projekt (indywidualny lub grupowy).	
19.	Warunki i forma zaliczenia poszczególnych komponentów przedmiotu/modułu: - ciągła kontrola obecności i postępów w zakresie tematyki zajęć, - praca kontrolna (obejmująca zagadnienia z kilku ćwiczeń), - przygotowanie i zrealizowanie projektu (indywidualnego lub grupowego),	
20.	20. Nakład pracy studenta/doktoranta	
	forma działań studenta/doktoranta	liczba godzin na realizację działań
	Zajęcia (wg planu studiów) z prowadzącym: - laboratorium:	45
	Praca własna studenta, doktoranta (w tym udział w pracach grupowych): - bieżące przygotowanie do zajęć: - czytanie wskazanej literatury: -przygotowanie projektu: - przygotowanie do prac kontrolnych:	15 10 30 10
	łącznie liczba godzin	100
	Liczba punktów ECTS	4