



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Systemy mikroprocesorowe [S1AiR1E>SM1]

### Przedmiot

Kierunek studiów

Automatyka i robotyka/Automatic Control and Robotics

Rok/Semestr

2/4

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

angielski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

### Liczba godzin

Wykład

30

Laboratorium

30

Inne

0

Ćwiczenia

0

Projekty/seminaria

0

### Liczba punktów ECTS

4,00

### Koordynatorzy

dr inż. Dominik Łuczak

dominik.luczak@put.poznan.pl

### Wykładowcy

### Wymagania wstępne

Wiedza: Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z elektroniki oraz podstaw programowania. Umiejętności: Student powinien posiadać umiejętność rozwiązywania podstawowych problemów z zakresu cyfrowego przetwarzania sygnałów oraz umiejętność pozyskiwania informacji ze wskazanych źródeł. Powinien również rozumieć konieczność poszerzania swoich kompetencji i być gotowy do podjęcia współpracy w zespole. Kompetencje społeczne: Ponadto w zakresie kompetencji społecznych student musi przejawiać takie cechy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawczą, kreatywność, kulturę osobistą, szacunek dla innych ludzi.

### Cel przedmiotu

1. Przekazanie studentom podstawowej wiedzy dotyczącej architektury i programowania mikrokontrolerów. 2. Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z przetwarzaniem danych i komunikacją za pomocą interfejsów w mikroprocesorowych systemach elektronicznych. 3. Kształtowanie u studentów znaczenia znajomości norm i zaleceń związanych z budową i programowaniem mikroprocesorowych urządzeń elektronicznych.

### Przedmiotowe efekty uczenia się

W zakresie wiedzy:

Ma uporządkowaną wiedzę w zakresie architektur komputerów, systemów i sieci komputerowych oraz systemów operacyjnych w tym systemów operacyjnych czasu rzeczywistego [K1\_W9 (P6S\_WG)].  
Zna i rozumie w zaawansowanym stopniu teorię i metody w zakresie architektury i programowania systemów mikroprocesorowych, zna i rozumie wybrane języki wysokiego i niskiego poziomu programowania mikroprocesorów; zna i rozumie zasadę działania podstawowych modułów peryferyjnych oraz interfejsów komunikacyjnych stosowanych w systemach mikroprocesorowych [K1\_W13 (P6S\_WG)].  
Zna i rozumie typowe technologie inżynierskie, zasady oraz techniki konstruowania prostych systemów automatyki i robotyki; zna i rozumie zasady doboru układów wykonawczych, jednostek obliczeniowych oraz elementów i urządzeń pomiarowo-kontrolnych [K1\_W20 (P6S\_WG)].

W zakresie umiejętności:

Potrafi odczytywać ze zrozumieniem projektową dokumentację techniczną oraz proste schematy technologiczne systemów automatyki i robotyki [K1\_U2 (P6S\_UW)].

Potrafi korzystać z wybranych narzędzi szybkiego prototypowania układów automatyki i robotyki [K1\_U13 (P6S\_UW)].

Potrafi dobrać rodzaj i parametry układu pomiarowego, jednostki sterującej oraz modułów peryferyjnych i komunikacyjnych dla wybranego zastosowania oraz dokonać ich integracji w postaci wynikowego systemu pomiarowo-sterującego [K1\_U22 (P6S\_UW)].

Potrafi skonstruować algorytm rozwiązania prostego zadania pomiarowego i sterującego oraz zaimplementować, przetestować i uruchomić go w wybranym środowisku programistycznym na platformie mikroprocesorowej [K1\_U27 (P6S\_UW)].

W zakresie kompetencji społecznych:

Jest gotów do krytycznej oceny posiadanej wiedzy; rozumie potrzebę i zna możliwości ciągłego dokształcania się – podnoszenia kompetencji zawodowych, osobistych i społecznych, potrafi inspirować i organizować proces uczenia się innych osób [K1\_K1 (P6S\_KK)].

Posiada świadomość konieczności profesjonalnego podejścia do zagadnień technicznych, skrupulatnego zapoznania się z dokumentacją oraz warunkami środowiskowymi, w których urządzenia i ich elementy mogą funkcjonować; jest gotów do przestrzegania zasad etyki zawodowej i wymagania tego od innych, poszanowania różnorodności poglądów i kultur [K1\_K5 (P6S\_KR)].

## Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

a) w zakresie wykładów:

na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,

b) w zakresie laboratorium:

na podstawie oceny znajomości i zrozumienia bieżących zagadnień prezentowanych w ramach przedmiotu.

c) W ramach obu form zajęć istnieje możliwość wykorzystania zadań realizowanych w formule Problem Based Learning (PBL), wspierających bieżące potrzeby badawcze i techniczne koordynatora przedmiotu oraz nadzorowanych przez prowadzącego, z uwzględnieniem iteracyjnego i cyklicznego charakteru realizacji zadań, pod warunkiem że są one zbieżne z treściami programowymi kursu.

Ocena podsumowująca:

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:

i. ocenę wiedzy i umiejętności wykazanych na egzaminie w formie testu

ii. omówienie wyników egzaminu.

b) w zakresie laboratorium weryfikowanie założonych efektów kształcenia realizowane jest przez:

i. ocenę przygotowania studenta do poszczególnych zajęć,

ii. ocenianie ciągłe, na każdych zajęciach (odpowiedzi ustne) - premiowanie przyrostu umiejętności posługiwania się poznanymi zasadami i metodami,

iii. ocenę sprawozdań przygotowywanych częściowo w trakcie zajęć, a także po ich zakończeniu.

c) W ramach oceny podsumowującej istnieje możliwość uwzględnienia wyników zadań realizowanych w formule Problem Based Learning (PBL) opracowanych na potrzeby badawcze i techniczne koordynatora przedmiotu oraz nadzorowanych przez prowadzącego, pod warunkiem ich zgodności z treściami programowymi kursu.

d) Uzyskiwanie dodatkowych punktów za aktywność podczas zajęć, w szczególności za:

i. samodzielną budowę modułu elektronicznego z mikroprocesorem i opracowanie dokumentacji

ii. efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu

iii. uwagi związane z udoskonaleniem materiałów dydaktycznych.

## Treści programowe

Przedmiot ten obejmuje szeroki zakres zagadnień związanych z systemami mikroprocesorowymi, kładąc nacisk na ich budowę, programowanie i zastosowanie. Studenci poznają zarówno teoretyczne podstawy, jak i praktyczne umiejętności niezbędne do pracy z systemami mikroprocesorowymi w różnych dziedzinach.

W ramach wykładów studenci zapoznają się z logiką cyfrową, w tym z podstawowymi operacjami logicznymi, programową i sprzętową implementacją układów kombinacyjnych i sekwencyjnych, systemami kodowania liczb, operacjami bitowymi, a także z budową interfejsu użytkownika i wymianą informacji między mikrokontrolerem a innym urządzeniem. Nauczą się również projektować schematy elektryczne i mozaiki PCB dla układów cyfrowych i mikroprocesorowych.

Zajęcia laboratoryjne pozwolą studentom na praktyczne zastosowanie wiedzy zdobytej na wykładach oraz rozwijanie umiejętności niezbędnych do projektowania i implementacji systemów mikroprocesorowych.

Program kursu zapewnia kompleksowe przygotowanie do projektowania i implementacji systemów mikroprocesorowych. Studenci zdobędą wiedzę teoretyczną i praktyczne umiejętności niezbędne do pracy w różnych dziedzinach.

## Tematyka zajęć

Program wykładu obejmuje następujące zagadnienia:

1. Operacje logiczne. Podstawowe operacje algebry Boole'a, realizacja programowa i sprzętowa, zastosowania w projektowaniu układów cyfrowych.
2. Układy kombinacyjne małej skali integracji, podstawy projektowania i implementacji prostych układów kombinacyjnych. Postać sum iloczynów i iloczynów sum z wykorzystaniem map Karnaugh'a
3. Operacje bitowe i kodowanie danych. Operacje bitowe, systemy kodowania liczb, maski bitowe, manipulacje danymi wielobajtowymi. Wykorzystanie wszystkich operatorów bitowych oraz technik ekstrakcji bajtów z dowolnych typów danych, łączenia ich w nowe struktury, operowania na fragmentach danych z zastosowaniem struktur i unii, a także uzyskiwania dostępu do pojedynczych bitów zmiennej — zarówno w celu odczytu, jak i modyfikacji wartości.
4. Układy kombinacyjne – realizacja sprzętowa i programowa. Układy średniej skali integracji, projektowanie, minimalizacja wyrażeń logicznych. Obejmują definicję oraz właściwości multiplekserów i demultiplekserów wraz z ich typowymi zastosowaniami, a także sprzętową realizację układów kombinacyjnych z wykorzystaniem multiplekserów, demultiplekserów, ich konfiguracji łączonych oraz pamięci. Zakres wykładu obejmuje również programową implementację układów kombinacyjnych w języku C z użyciem instrukcji switch oraz struktur danych opartych na pamięci, w tym tablic, unii i struktur z polami bitowymi.
5. Układy sekwencyjne – komponenty, definicje, struktury i zastosowania. Zakres obejmuje definicję układów sekwencyjnych, maszyna stanów oraz charakterystykę struktur Mealy'ego i Moore'a wraz z ich właściwościami i typowymi zastosowaniami, a także omówienie przerzutników, ich budowy, parametrów i funkcji, w tym przerzutnika SR wraz z zagadnieniem eliminacji drgań styków (debouncing), przerzutnika D używanego m.in. jako preskaler częstotliwości, przerzutnika JK, przerzutników z wejściem zezwalającym (enable) oraz układów wykrywania zbocza. Wykład obejmuje również definicję, strukturę, właściwości i zastosowania rejestrów przesuwających.
6. Układy sekwencyjne – implementacja i liczniki sprzętowe. Zakres obejmuje praktyczne aspekty implementacji układów sekwencyjnych, z uwzględnieniem struktur Mealy'ego i Moore'a, ich właściwości oraz typowych zastosowań. Omawiana jest sprzętowa realizacja układów sekwencyjnych z wykorzystaniem przerzutnika D oraz implementacja programowa w języku C. Wykład obejmuje również działanie liczników sprzętowych, w tym metody zwiększania ich rozdzielczości bitowej, zliczania zboczy dodatnich i ujemnych sygnału oraz obliczania sumarycznej liczby impulsów.
7. Typy danych i arytmetyka – liczby całkowite, zmiennoprzecinkowe oraz struktury danych. Zakres obejmuje standardowe typy całkowite o szerokości 8, 16, 32 i 64 bitów, ich reprezentację bitową, interpretację wartości, rozmiary w bajtach oraz własności działań takich jak dzielenie całkowite i operacja modulo. Omawiany jest także typ zmiennoprzecinkowy, ze szczególnym uwzględnieniem problemów arytmetycznych wynikających z ograniczonej precyzji, sposobu wyodrębniania części całkowitej i ułamkowej oraz realizacji operacji modulo dla liczb zmiennoprzecinkowych. Wykład obejmuje również zastosowanie unii i struktur, analizę ich rozmiaru, wyrównania i dopełnień pamięci (alignment i padding), a także modyfikację zasad wyrównania przy użyciu dyrektywy pragma pack. Omawiane są ponadto metody przekształcania i skalowania wartości zmiennych pomiędzy różnymi zakresami numerycznymi.
8. Konwersje danych tekstowych, rozmiary tekstu i makra preprocesora. Konwersja tekst–liczba i liczba–

tekst, znaki specjalne, makra i operacje na strukturach danych. Zakres obejmuje analizę różnic między funkcjami strlen i sizeof w kontekście określania rozmiaru tekstu i danych, metody konwersji wartości zmiennych na reprezentację tekstową oraz konwersji tekstu na wartości zmiennych różnych typów. Omawiane są również techniki tworzenia i stosowania makr wykorzystujących token pasting oraz stringizing w celu generowania kodu na etapie preprocesora.

9. Interfejs tekstowy użytkownika z wyświetlaczem LCD. Podstawy tworzenia interfejsu użytkownika, prezentacja wyników obliczeń. Zakres obejmuje podłączenie wyświetlacza LCD do mikrokontrolera, przesyłanie komend i danych w trybie 4 bitowym, tworzenie i wykorzystanie własnych znaków, wysyłanie pojedynczych znaków oraz całych ciągów tekstowych do wyświetlacza, a także realizację przełączania stron tekstowych w celu prezentacji złożonych interfejsów użytkownika.

10. Kod Graya – konwersje i zastosowania. Zakres obejmuje konwersję liczb zakodowanych w systemie BCD (Binary Coded Decimal) do kodu odbitego RBC (Reflected Binary Code, tzw. kod Graya) zarówno w realizacji sprzętowej, jak i programowej, a także odwrotną konwersję z kodu RBC do BCD z wykorzystaniem metod sprzętowych i programowych. Omawiane są również przykładowe zastosowania kodu RBC, w tym jego wykorzystanie w systemach pomiarowych, enkoderach położenia oraz układach minimalizujących ryzyko błędów podczas zmiany stanów bitowych.

11. Karta rozszerzeń procesora sygnałowego DSP. Architektura, komunikacja i zastosowanie kart DSP w systemach cyfrowych. Zakres obejmuje stosowanie pełnego workflow projektowania układów elektronicznych, w tym przygotowanie interfejsu sprzętowego dla enkodera inkrementalnego oraz interfejsu dla enkodera absolutnego z uwzględnieniem wymagań sygnałowych i czasowych charakterystycznych dla obu typów czujników. Omawiane jest również projektowanie płytek rozszerzeń dla kontrolera DSP, obejmujące wykorzystanie linii adresowych procesora wraz ze sprzętowym dekodere adresów, obsługę linii danych oraz użycie sygnałów sterujących, takich jak linie odczytu i zapisu, w celu prawidłowej integracji modułu z magistralą systemową.

12. Procesor sygnałowy DSP – budowa i działanie. Zakres obejmuje poziomy integracji układów cyfrowych oraz zasady łączenia procesora sygnałowego DSP z pozostałymi elementami systemu elektronicznego, ze szczególnym uwzględnieniem wykorzystania interfejsu pamięci. Omawiane są metody projektowania sprzętowego dekodera adresów, realizacja operacji zapisu danych do urządzeń peryferyjnych oraz odczytu danych z peryferiów za pośrednictwem magistrali pamięci. Wykład obejmuje również zagadnienia związane z przestrzenią adresową, w tym rozróżnienie pomiędzy adresami logicznymi a adresami sprzętowymi, oraz ich znaczenie dla poprawnej integracji modułów systemu.

13. Projektowanie urządzeń elektronicznych – od koncepcji do integracji układów cyfrowych. Zakres obejmuje pełny workflow projektowania urządzeń elektronicznych, w tym opracowanie obudowy urządzenia, przygotowanie graficznego layoutu interfejsu oraz projektowanie elektroniki urządzenia. Omawiane są metody realizacji sprzętowych interfejsów wyjściowych, w szczególności wykorzystanie tranzystorów oraz przekaźników jako elementów sterujących sygnałami cyfrowymi, a także zastosowanie optoizolatorów jako interfejsów wejściowych. Wykład obejmuje również techniki translacji poziomów napięć w układach cyfrowych oraz zasady okablowania i łączenia elementów urządzenia o różnych poziomach integracji, w tym podłączanie wyjść analogowych z interfejsami równoległymi oraz integrację liczników kwadraturowych.

14. Wybrane zastosowania układów cyfrowych i mikroprocesorowych. Przykłady integracji funkcjonalnej i aplikacyjnej.

15. Podsumowanie kursu. Omówienie treści, podsumowanie zagadnień.

Program zajęć laboratoryjnych obejmuje następujące zagadnienia:

1. Zajęcia organizacyjne - zapoznanie z aparaturą i przypisami BHP, wstęp do środowisk projektowych.
2. Programowa i sprzętowa implementacja i weryfikacja operacji logicznych. Bezpośrednia manipulacja rejestrami mikrokontrolera. Analiza czasu wykonania operacji.
3. Programowa i sprzętowa implementacja i weryfikacja układów kombinacyjnych.
4. Programowa implementacja i weryfikacja operacji bitowych. Rozkładanie i składanie liczb wielobajtowych z wykorzystaniem masek bitowych, unii i struktur danych. Wprowadzenie do mapowania struktur na rejestry sprzętowe.
5. Zastosowanie multiplekserów, demultiplekserów, przerzutników i pamięci do realizacji wybranego układu cyfrowego np. karty rozszerzeniowej przetwornika obrotowo-impulsowego/kodowego dla systemu mikroprocesorowego.
6. Programowa i sprzętowa implementacja układów sekwencyjnych, w tym maszyny stanów
7. Analiza arytmetyki w systemach mikroprocesorowych o skończonej precyzji – liczby całkowite i zmiennoprzecinkowe, wykorzystanie jednostki FPU, czas i precyzja obliczeń, wyrównanie i organizacja struktur danych.
8. Budowa fragmentów interfejsu użytkownika prezentującego rezultaty obliczeń – konwersja liczba–tekst i

tekst–liczba, formatowanie danych oraz zastosowanie makr preprocesora.

9. Implementacja menu użytkownika w oparciu o maszynę stanów oraz sterowanie wyświetlaczem LCD w trybie 4-bitowym.

10. Analiza przestrzeni adresowej mikrokontrolera oraz projekt logicznej mapy pamięci dla rozszerzenia systemu (symulacja dekodowania adresów i integracji peryferiów).

11. Projekt koncepcyjny układu cyfrowego współpracującego z mikrokontrolerem (np. modułu enkodera absolutnego) obejmujący schemat elektryczny oraz analizę poziomów napięć.

12. Przygotowanie mozaiki PCB dla układów cyfrowych np. karty rozszerzeniowej przetwornika obrotowo-impulsowego i/lub przetwornika obrotowo-kodowego dla systemu mikroprocesorowego.

13. Rozbudowa schematów elektrycznych i mozaiki PCB o układ mikroprocesorowy.

14. Prezentacja zadania zaliczeniowego obejmującego np. mikroprocesorowy system prostego interfejsu użytkownika dla wybranego peryferia wewnętrznego lub zewnętrznego.

## Metody dydaktyczne

1. Wykład: prezentacja multimedialna ilustrowana symulacjami komputerowymi

2. Zajęcia laboratoryjne: wykorzystanie mikroprocesorowych modułów uruchomieniowych firmy STM, środowiska programistyczne IDE

3. W ramach obu form zajęć istnieje możliwość wykorzystania elementów Problem Based Learning (PBL), w ramach którego studenci pracują nad zagadnieniami i projektami definiowanymi na potrzeby badawcze i techniczne koordynatora przedmiotu oraz nadzorowanymi przez prowadzącego. W podejściu tym szczególny nacisk kładzie się na iteracyjny charakter pracy, obejmujący analizę problemu, projektowanie rozwiązania, jego praktyczną weryfikację oraz systematyczne udoskonalanie.

## Literatura

Podstawowa

1. Geoffrey Brown, Discovering the STM32 Microcontroller, 2016

2. Donald S. Reay, Digital Signal Processing Using the ARM Cortex M4, 2015

3. Dogan Ibrahim, Microcontroller Based Applied Digital Control, 2006

4. W. Gay, Beginning STM32 Developing with FreeRTOS, libopencm3 and GCC, APRESS, 2018.

Uzupełniająca

1. D. Łuczak, A. Wójcik, DSP implementation of state observers for electrical drive with elastic coupling , Przegląd Elektrotechniczny R.92 nr 5, s. 100-105, 2016.

2. M. Szumski, Mikrokontrolery STM32 w systemach sterowania i regulacji, BTC, 2018.

3. A. Kurczyk, Mikrokontrolery STM32 dla początkujących, BTC, 2019.

4. K. Paprocki, Mikrokontrolery STM32 w praktyce, BTC, 2009.

5. Łuczak, D. Machine Fault Diagnosis through Vibration Analysis: Continuous Wavelet Transform with Complex Morlet Wavelet and Time–Frequency RGB Image Recognition via Convolutional Neural Network. Electronics 2024, 13, 452, doi:10.3390/electronics13020452.

## Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	120	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	60	2,00