



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Systemy mikroprocesorowe [S1AiR2P>SM]

Przedmiot

Kierunek studiów

Automatyka i robotyka

Rok/Semestr

3/5

Studia w zakresie (specjalność)

–

Profil studiów

praktyczny

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

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. Tomasz Marciniak

tomasz.marciniak@put.poznan.pl

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

Wiedza:

1. Student ma uporządkowaną wiedzę w zakresie architektur komputerów, systemów i sieci komputerowych oraz systemów operacyjnych w tym systemów operacyjnych czasu rzeczywistego - [K1_W9]
2. 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]
3. 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]

Umiejętności:

1. Student potrafi odczytywać ze zrozumieniem projektową dokumentację techniczną oraz proste schematy technologiczne systemów automatyki i robotyki- [K1_U2]
2. potrafi korzystać z wybranych narzędzi szybkiego prototypowania układów automatyki i robotyki - [K1_U13]
3. 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]
4. potrafi skonstruować algorytm rozwiązania prostego zadania pomiarowego i sterującego oraz zaimplementować, przetestować i uruchomić go w wybranym środowisku programistycznym na platformie mikroprocesorowej - [K1_U27]

Kompetencje społeczne:

1. Student 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]
2. 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]

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 ciągle, na każdym 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. Program obejmuje następujące zagadnienia: budowa i programowanie mikrokontrolerów, wejścia/wyjścia cyfrowe, przetworniki A/C i C/A, komunikacja szeregową, układy licznikowe, pamięci w systemach mikroprocesorowych, tryby obniżonego poboru mocy, komunikacja sieciowa, realizacja regulatorów, algorytmy cyfrowego przetwarzania sygnałów, systemy operacyjne czasu rzeczywistego.

Tematyka zajęć

Program wykładu obejmuje następujące zagadnienia:

1. Elementy i zadania systemu mikroprocesorowego, budowa mikrokontrolera, rynek, producenci i rodziny mikrokontrolerów, moduły rozwojowe z mikrokontrolerem, środowiska programistyczne. Motywacja do nauki.
2. Wejścia/wyjścia cyfrowe (GPIO) - budowa wewnętrzna, interfejs elektroniczny (przycisk, klawiatura, LCD, LED, wyświetlacz 7 segmentowy, optoizolacja, przełączniki, tranzystory), obsługa programowa (odpytywanie, NVIC). Problem drgań styków.
3. Komunikacja szeregową (UART) budowa wewnętrzna, interfejs elektroniczny (RS232, RS485), obsługa programowa (odpytywanie, NVIC, DMA).
4. Układy licznikowe (TIM) - budowa wewnętrzna, interfejs elektroniczny, obsługa programowa, zastosowanie jako PWM, one pulse, licznik kwadraturowy, sterowanie triakiem, mostek H, tranzystor, LED.
5. Komunikacja: SPI, I2C, CAN, 1-Wire, USB, Ethernet.
6. Przetworniki A/C i C/A - budowa wewnętrzna, interfejs elektroniczny, PWM z filtrem analogowym jako wyjście analogowe, generacja sygnałów, problem kalibracji.
7. Realizacja regulatorów i transmitancji dyskretnej. Dyskretyzacja obiektów dynamicznych. Wprowadzenie do CMSIS-DSP.
8. Implementacja algorytmów cyfrowego przetwarzania sygnałów z zastosowaniem CMSIS: filtracja cyfrowa (FIR, IIR, LMS), wyznaczenie dyskretnej transformaty Fouriera algorytmami FFT. Operacje macierzowe.
9. Wprowadzenie do systemu operacyjnego czasu rzeczywistego FreeRTOS.
10. Komunikacja sieciowa; biblioteka LwIP; protokoły TCP, UDP.
11. Interfejs WWW (serwer HTTP) na systemie mikroprocesorowym (FreeRTOS+LwIP).
12. Pamięci stosowane w systemach mikroprocesorowych. Weryfikacja integralności danych (CRC).
13. Tryby obniżonego poboru mocy. Zabezpieczenia systemów mikroprocesorowych przed błędnym działaniem programu (watchdog).
14. Czas rzeczywisty w systemach mikroprocesorowych (RTC i protokół NTP).
15. Podsumowanie.

Program zajęć laboratoryjnych obejmuje następujące zagadnienia:

1. Zajęcia organizacyjne - zapoznanie z aparaturą i przypisami BHP, wstęp do środowiska projektowego
2. Wejścia/wyjścia cyfrowe, obsługa przerwań; LED, przyciski monostabilne, impulsator obrotowy
3. Port szeregowy; komunikacja uC z PC
4. Programowalne liczniki; układ fazowego sterowania żarówką
5. Sterowanie PWM; LED RGB
6. I2C; cyfrowy czujnik natężenia światła
7. SPI; cyfrowy czujnik temp./ciśnienia, biblioteki producenta
8. Przetwornik A/C; obsługa analogowych czujników (fotorezystor, termistor)
9. Przetwornik C/A; generacja sygnałów analogowych o zadanych parametrach z wykorzystaniem przerwań i DMA
10. Biblioteka CMSIS - operacje macierzowe, filtry cyfrowe FIR/IIR
11. Biblioteka CMSIS - regulator PID

12. System czasu rzeczywistego FreeRTOS
13. Obsługa karty SD; system plików FatFS;
14. Komunikacja sieciowa; biblioteka LwIP; protokoły TCP, UDP, HTTP (web server)
15. Prezentacja zadania zaliczeniowego: mikroprocesorowy system pomiaru i sterowania

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. M. Szumski, Mikrokontrolery STM32 w systemach sterowania i regulacji, BTC, 2018.
2. A. Kurczyk, Mikrokontrolery STM32 dla początkujących, BTC, 2019.
3. K. Paprocki, Mikrokontrolery STM32 w praktyce, BTC, 2009.
4. P. Hadam, Projektowanie systemów mikroprocesorowych, BTC, 2004.

Uzupełniająca:

1. W. Gay, Beginning STM32 Developing with FreeRTOS, libopencm3 and GCC, APRESS, 2018.
2. T. Marciniak, A. Dąbrowski, R. Puchalski, D. Dratwiak, W. Marciniak, Zastosowanie mikrokontrolera STM32F410 do prezentacji zagadnień cyfrowego przetwarzania sygnałów, Przegląd Elektrotechniczny R. 95, s. 118-120, 2019.
3. 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.
4. Mikrokontrolery i IoT zapewniają elektronice szybki rozwój - raport, Elektronik nr 8, s. 28-47, 2019.
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