

Zespół Autorski:

dr hab. Wojciech Zabołotny

Wstęp Do Systemów Wbudowanych (WSW) *Introduction to Embedded Systems*

Poziom kształcenia: *I stopień*

Forma i tryb prowadzenia przedmiotu: *stacjonarna*

Kierunek studiów: *Elektronika*

Specjalność: *Elektronika i Fotonika*

Grupa przedmiotów:

Poziom przedmiotu: *podstawowy*

Status przedmiotu: *obowiązkowy*

Język przedmiotu: *polski*

Semestr nominalny: *5*

Minimalny numer semestru: *5*

Wymagania wstępne, zalecane przedmioty poprzedzające: *Podstawy techniki cyfrowej (POCY), Systemy cyfrowe i komputerowe (SCK), Podstawy techniki mikroprocesorowej (POMIK)*

Limit liczby studentów: *150*

Powód zgłoszenia przedmiotu: *modyfikacja i unowocześnienie programu studiów dla kierunku Elektronika, zmiana specjalności EiK na Elektronika i Fotonika*

Cel przedmiotu:

Przedstawienie podstawowych zagadnień związanych z systemami wbudowanymi i zaznajomienie z metodami projektowania, realizacji i uruchamiania prostych systemów wbudowanych ze szczególnym uwzględnieniem mikroprocesorów wykorzystujących architekturę Cortex-M. Po przedmiocie WSW student powinien rozumieć różnicę między zwykłym systemem komputerowym a systemem wbudowanym, powinien potrafić zaprojektować i zrealizować prosty system wbudowany wykorzystujący mikroprocesor ARM Cortex-M i komunikujący się z układami peryferyjnymi za pośrednictwem typowych interfejsów takich jak I2C, SPI, UART, 1-Wire itp. wykorzystując przerwania i DMA, powinien umieć przetestować pracę zaprojektowanego systemu przy pomocy dedykowanego oprogramowania. Powinien umieć zapewnić właściwe zasilanie systemu z uwzględnieniem trybów ograniczonego poboru mocy. Powinien umieć stworzyć proste oprogramowanie ten system używając języka C, a także powinien umieć, tam gdzie to możliwe, wykorzystać języki skryptowe dla systemów wbudowanych (micropython, eLua).

Treść kształcenia:

Informacje ogólne:

Wydziałowa Komisja Akredytacji Przedmiotów (WKAP)

Zajęcia będą obejmować klasyczne wykłady wprowadzające w zagadnienia ogólne oraz zajęcia zintegrowane, prowadzone w laboratorium, obejmujące wprowadzenie w postaci krótkiego wykładu, odwołującego się do rozszerzonych materiałów dostępnych dla studentów na ich stanowiskach. W ramach tych zajęć studenci powinni zrealizować:

- miniprojekt obejmujący zaplanowanie rozwiązania zadania postawionego przed studentem w ramach części laboratoryjnej (przesłany prowadzącemu)
- pracę laboratoryjną z wykorzystaniem dostępnego sprzętu i oprogramowania,
- przedstawienie prowadzącemu opracowanego rozwiązania zadania i zademonstrowanie jego działania.
- Przygotowanie raportu podsumowującego uzyskane wyniki (przesłany prowadzącemu)

Zajęcia podlegać będą ocenie wynikającej z jakości miniprojektu i raportu końcowego oraz przebiegu pracy w laboratorium.

Końcowe zajęcia laboratoryjne mogą być realizowane jako większy projekt zespołowy, wykonywane na kilku terminach laboratorium.

Opis wykładu:

1. Wstęp, omówienie tematyki systemów wbudowanych, klasyfikacja systemów wbudowanych, schemat blokowy systemu wbudowanego
2. Zagadnienia związane z zasilaniem systemu wbudowanego i podłączaniem układów zewnętrznych – analogowych i cyfrowych. Zagadnienia związane z odsprężeniem zasilania, prowadzeniem masy, barierami izolacyjnymi
3. Omówienie architektury ARM
4. Zagadnienia ekonomiczne związane z realizacją systemów wbudowanych – optymalizacja kosztów. Zagadnienia związane z certyfikacją systemów wbudowanych.

Laboratorium:

Zajęcia laboratoryjne pozwolą studentom zapoznać się w praktyce z następującymi zagadnieniami:

- Środowisko do tworzenia oprogramowania dla używanych systemów, biblioteki
- Uruchamianie i debugowanie oprogramowania na systemie wbudowanym
- Konfiguracja i obsługa wewnętrznych zegarów i urządzeń peryferyjnych
- Wykorzystanie przerwań i DMA
- Zapewnienie niezawodności systemu
- Obsługa interfejsów komunikacyjnych
- Obsługa wejść i wyjść analogowych
- Korzystanie z pamięci nieulotnej
- Zarządzanie poborem mocy

Zajęcia projektowo/laboratoryjne:

Zajęcia (większy projekt grupowy obejmują następujące elementy:

- Wybór zadania do rozwiązania, dobór właściwego procesora z wykorzystaniem CubeMX, wybór właściwego modułu mikroprocesorowego i płytek rozszerzających, opracowanie schematu zaprojektowanej platformy sprzętowej.

Wydziałowa Komisja Akredytacji Przedmiotów (WKAP)

- Stworzenie i uruchomienie oprogramowania diagnostycznego dowodzącego poprawnego działania opracowanej platformy sprzętowej.
- Stworzenie i uruchomienie oprogramowania realizującego założoną funkcjonalność projektowanego systemu.

Egzamin: *NIE*

Literatura:

1. Mikrokontrolery STM32 w praktyce, Krzysztof Paparocki, BTC, 2011
2. Mikrokontrolery STM32 dla początkujących, Aleksander Kurczyk, BTC, 2019
3. Cortex-M3 Technical Reference Manual, ARM Infocenter

Literatura uzupełniająca dotycząca FreeRTOS i MicroPythona

4. Mastering the FreeRTOS Real Time Kernel - a Hands On Tutorial Guide
5. FreeRTOS V10.0.0 Reference Manual
6. Programming with MicroPython, Nicholas H. Tollervey, O'Reilly Media; September 2017

Oprogramowanie:

CubeMX, środowisko do tworzenia i uruchamiania oprogramowania dla procesorów ARM

Wymiar godzinowy zajęć:	W	C	L	P	
	1	1	1	1	(60h/sem.)

Wymiar w jednostkach ECTS: 4

Liczba godzin pracy studenta związanych z osiągnięciem efektów kształcenia (opis):

1. *liczba godzin kontaktowych – 65 godz., w tym*
 - *obecność na wykładach 15 godz.*
 - *obecność na ćwiczeniach audytoryjnych 15 godz*
 - *obecność na laboratorium 15 godz*
 - *udział w konsultacjach 20 godz*
2. *praca własna studenta – 35 godz., w tym*
 - *przygotowanie do ćwiczeń 5 godz.,*
 - *przygotowanie do laboratoriów 5 godz.,*
 - *przygotowanie do kolokwiiów 10 godz.,*
 - *wykonywania zadań projektowych 10 godz.,*
 - *przygotowanie sprawozdań (projekty i laboratoria) 5 godz.*

Łączny nakład pracy studenta wynosi 100 godz., co odpowiada 4 pkt ECTS.

Wydziałowa Komisja Akredytacji Przedmiotów (WKAP)

Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich: *2,6 pkt ECTS, co odpowiada 65 godz. kontaktowym.*

Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym: *1,2 pkt ECTS, co odpowiada 15 godz. ćwiczeń laboratoryjnych i 15 godz. zadań projektowych*

Efekty kształcenia/uczenia się:

Wydziałowa Komisja Akredytacji Przedmiotów (WKAP)

Efekty kształcenia/uczenia się student, który zaliczył przedmiot:	forma zajęć/ technika kształcenia	sposób weryfikacji (oceny)	odniesienie do efektów uczenia się dla programu
WIEDZA			
Ma podstawową wiedzę w zakresie informatyki i telekomunikacji	laboratorium	Projekt i raport	K1_W03
Ma uporządkowaną, podbudowaną teoretycznie wiedzę w zakresie metodyki i technik programowania oraz teorii algorytmów.	wykład / ćwiczenia	kolokwium	K1_W04
Ma szczegółową wiedzę w jednym z obszaru architektury i oprogramowania systemów mikroprocesorowych	laboratorium / projekt	raport	K1_W12
Orientuje się w obecnym stanie oraz najnowszych trendach rozwojowych elektroniki.	wykład	kolokwium	K1_W13
UMIĘJĘTNOŚCI			
Potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł, także w języku angielskim; potrafi integrować uzyskane informacje, dokonywać ich interpretacji, a także wyciągać wnioski oraz formułować i uzasadniać opinie	laboratorium / projekt	raport	K1_U04
Potrafi porównać konstrukcje elementów i prostych układów i systemów elektronicznych stosując określone kryteria użytkowe (np. Szybkość działania, pobór mocy).	laboratorium / projekt	raport	K1_U13
Potrafi korzystać z kart katalogowych i not aplikacyjnych w celu dobrania odpowiednich komponentów projektowanego układu lub systemu elektronicznego.	laboratorium / projekt	raport	K1_U19
Potrafi tworzyć oprogramowanie systemów mikroprocesorowych w języku niskiego poziomu.	laboratorium / projekt	raport	K1_U21
KOMPETENCJE SPOŁECZNE			
Ma świadomość ważności i rozumie pozatechniczne aspekty i skutki działalności inżynierskiej, w tym jej wpływ na środowisko, i związaną z tym odpowiedzialność za podejmowane decyzje.	laboratorium / projekt / ćwiczenia	raport	K1_K02
Potrafi współdziałać i pracować w grupie, przyjmując w niej różne role.	laboratorium / projekt	raport	K1_K03