

**AGH**AGH UNIVERSITY OF SCIENCE
AND TECHNOLOGY

Nazwa modułu zajęć:	Technika mikroprocesorowa 2				
Rok akademicki:	2019/2020	Kod:	IETP-1-501-s	Punkty ECTS:	4
Wydział:	Informatyki, Elektroniki i Telekomunikacji				
Kierunek:	Elektronika i Telekomunikacja	Specjalność:	—		
Poziom studiów:	Studia I stopnia	Forma studiów:	Stacjonarne		
Język wykładowy:	Polski	Profil:	Ogólnoakademicki (A)	Semestr:	5
Strona www:	https://upel.agh.edu.pl/wiet/				
Prowadzący moduł:	dr hab. inż. Russek Paweł (russek@agh.edu.pl)				

Treści programowe zapewniające uzyskanie efektów uczenia się dla modułu zajęć

Celem kursu jest zapoznanie studentów z zaawansowanymi zagadnieniami z dziedziny techniki mikroprocesorowej. Zagadnienia są przedstawione na przykładzie mikrokontrolera 32 bitowego

Opis efektów uczenia się dla modułu zajęć

Kod MEU	Student, który zaliczył moduł zajęć zna i rozumie/potrafi/jest gotów do	Powiązania z KEU	Sposób weryfikacji i oceny efektów uczenia się osiągniętych przez studenta w ramach poszczególnych form zajęć i dla całego modułu zajęć
Wiedza: zna i rozumie			
M_W001	Student poszerzy wiedzę w zakresie architektur mikroprocesorów. Pozna metody komunikacji jednostki centralnej z innymi elementami systemu procesorowego, techniki przetwarzania współbieżnego na poziomie instrukcji, mechanizmy sprzętowe służące wsparciu pracy systemów operacyjnych, nowoczesne układy peryferyjne. Poszerzona zostanie wiedza studenta w zakresie programowania procesorów w językach wysokiego poziomu.	ETP1A_W14, ETP1A_W06	Odpowiedź ustna, Kolokwium, Egzamin

M_W002	Student dysponuje wiedzą niezbędną do tworzenia dedykowanych aplikacji mikrokontrolerów, zna współzależności pomiędzy hardwarem i softwarem oraz zasady pracy w czasie rzeczywistym.	ETP1A_W15, ETP1A_W14, ETP1A_W06	Odpowiedź ustna, Egzamin, Kolokwium
M_W003	Student dysponuje wiedzą konieczną do uruchamiania i rozbudowy systemu mikroprocesorowego.	ETP1A_W14, ETP1A_W07	Odpowiedź ustna, Egzamin, Kolokwium
Umiejętności: potrafi			
M_U001	Student umie zaprojektować układy współpracujące dla danej aplikacji mikrokontrolera, uwzględniając funkcjonalność jego interfejsów wewnętrznych. Potrafi dokonać optymalizacji wynikającej ze wzajemnej wymienialności sprzęt - oprogramowanie. Potrafi czytać dokumentację techniczną dotyczącą procesorów.	ETP1A_U03, ETP1A_U04, ETP1A_U02	Zaliczenie laboratorium, Wykonanie ćwiczeń laboratoryjnych, Projekt
M_U002	Student zdobędzie umiejętności pozwalające na samodzielną budowę systemu elektronicznego złożonego z procesora i układów peryferyjnych w oparciu o samodzielnie zaprojektowane i napisane oprogramowanie.	ETP1A_U03, ETP1A_U08, ETP1A_U02, ETP1A_U09	Zaliczenie laboratorium, Wykonanie ćwiczeń laboratoryjnych, Projekt
M_U003	Student potrafi oprogramować zaprojektowaną aplikację mikrokontrolera, posługując się językiem assemblerowym i/lub językiem wysokiego poziomu, uwzględniając uwarunkowania, wynikające z zasobów mikrokontrolera, jego listy instrukcji, pojemności pamięci i wymogów czasu rzeczywistego.	ETP1A_U09	Zaliczenie laboratorium, Wykonanie ćwiczeń laboratoryjnych, Projekt
Kompetencje społeczne: jest gotów do			
M_K001	Student jest świadomy odpowiedzialności odnośnie niezawodnego sterowania procesem technologicznym, etyki zawodowej i uwarunkowań społecznych, w odniesieniu do aplikacji dotyczących aparatury medycznej czy sprzętu powszechnego użytku.	ETP1A_K04, ETP1A_K01	Aktywność na zajęciach, Odpowiedź ustna, Udział w dyskusji
M_K002	Student nabędzie umiejętności pracy w zespole dzięki realizacji zadanych zadań projektowych.	ETP1A_K04, ETP1A_K01	Odpowiedź ustna, Aktywność na zajęciach

Liczba godzin zajęć w ramach poszczególnych form zajęć

Suma	Forma zajęć dydaktycznych										
	Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Prace kontrolne i przejściowe	Lektorat
56	28	0	18	10	0	0	0	0	0	0	0

Matryca kierunkowych efektów uczenia się w odniesieniu do form zajęć i sposobu zaliczenia, które pozwalają na ich uzyskanie

Kod MEU	Student, który zaliczył moduł zajęć zna i rozumie/potrafi/jest gotów do	Forma zajęć dydaktycznych										
		Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Prace kontrolne i przejściowe	Lektorat
Wiedza: zna i rozumie												
M_W001	Student poszerzy wiedzę w zakresie architektur mikroprocesorów. Pozna metody komunikacji jednostki centralnej z innymi elementami systemu procesorowego, techniki przetwarzania współbieżnego na poziomie instrukcji, mechanizmy sprzętowe służące wsparciu pracy systemów operacyjnych, nowoczesne układy peryferyjne. Poszerzona zostanie wiedza studenta w zakresie programowania procesorów w językach wysokiego poziomu.	+	-	-	-	-	-	-	-	-	-	-
M_W002	Student dysponuje wiedzą niezbędną do tworzenia dedykowanych aplikacji mikrokontrolerów, zna współzależności pomiędzy hardwarem i softwarem oraz zasady pracy w czasie rzeczywistym.	+	-	-	-	-	-	-	-	-	-	-
M_W003	Student dysponuje wiedzą konieczną do uruchamiania i rozbudowy systemu mikroprocesorowego.	+	-	-	-	-	-	-	-	-	-	-
Umiejętności: potrafi												

M_U001	Student umie zaprojektować układy współpracujące dla danej aplikacji mikrokontrolera, uwzględniając funkcjonalność jego interfejsów wewnętrznych. Potrafi dokonać optymalizacji wynikającej ze wzajemnej wymienialności sprzęt - oprogramowanie. Potrafi czytać dokumentację techniczną dotyczącą procesorów.	-	-	+	+	-	-	-	-	-	-	-
M_U002	Student zdobędzie umiejętności pozwalające na samodzielną budowę systemu elektronicznego złożonego z procesora i układów peryferyjnych w oparciu o samodzielnie zaprojektowane i napisane oprogramowanie.	-	-	+	+	-	-	-	-	-	-	-
M_U003	Student potrafi oprogramować zaprojektowaną aplikację mikrokontrolera, posługując się językiem assemblerowym i/lub językiem wysokiego poziomu, uwzględniając uwarunkowania, wynikające z zasobów mikrokontrolera, jego listy instrukcji, pojemności pamięci i wymogów czasu rzeczywistego.	-	-	+	+	-	-	-	-	-	-	-
Kompetencje społeczne: jest gotów do												
M_K001	Student jest świadomy odpowiedzialności odnośnie niezawodnego sterowania procesem technologicznym, etyki zawodowej i uwarunkowań społecznych, w odniesieniu do aplikacji dotyczących aparatury medycznej czy sprzętu powszechnego użytku.	-	-	+	-	-	-	-	-	-	-	-
M_K002	Student nabędzie umiejętności pracy w zespole dzięki realizacji zadanych zadań projektowych.	-	-	+	-	-	-	-	-	-	-	-

Nakład pracy studenta (bilans punktów ECTS)

Forma aktywności studenta	Obciążenie studenta
Udział w zajęciach dydaktycznych/praktyka	56 godz
Przygotowanie do zajęć	14 godz
przygotowanie projektu, prezentacji, pracy pisemnej, sprawozdania	24 godz
Samodzielne studiowanie tematyki zajęć	20 godz
Dodatkowe godziny kontaktowe	2 godz
Sumaryczne obciążenie pracą studenta	116 godz
Punkty ECTS za moduł	4 ECTS

Pozostałe informacje

Szczegółowe treści kształcenia w ramach poszczególnych form zajęć (szczegółowy program wykładów i pozostałych zajęć)

Wykład

Podstawy budowy i programowania mikrokontrolerów 32-bitowych

Wprowadzenie do mikrokontrolerów: typy, zastosowania

Budowa mikrokontrolera ARM Corex M – moduły (rdzeń, kontroler przerwań, peryferia, magistrala), przestrzeń adresowa

Biblioteka ARM CMSIS

Programowanie mikrokontrolerów w oparciu o przerwania: cel i zalety

Architektura (ISA) współczesnych procesorów 32-bitowych

Architektura procesora – co to jest? Architektury: von Neumana, Harvardzka.

Procesory RISC i CISC

Architektura procesora na przykładzie procesora ARM Cortex M

Rejestry ARM Cortex-M

Tryby pracy procesora ARM Cortex-M

Instrukcje ARM, Thumb, Thumb 2

Assembler i techniki kompilacji dla współczesnego procesora 32 bitowego

Zestaw instrukcji processors

Tryby adresowania

Techniki kompilacji kodu C do assemblera (zmienne automatyczne i statyczne, tablice pętli, rozgałęzienia)

Mikroarchitektura współczesnych procesorów 32-bitowych

Budowa jednostki centralnej – ALU, ścieżka danych, rejestry, dostęp do pamięci

Realizacja instrukcji w jednym taktie zegara (ang. single cycled architecture)

Potokowe wykonywanie instrukcji – architektura potokowa, skoki, cykle oczekiwania

Arytmetyka stałoprzecinkowa komputerów

Reprezentacja binarna, BCD, znak-moduł, z przesunięciem, kod uzupełnień do dwóch (U2)

Reprezentacja liczb ze stałym przecinkiem

Realizacja konwersji pomiędzy różnymi formatami liczb

Realizacja operacji dodawania i mnożenia liczb

Arytmetyka zmiennoprzecinkowa liczb

Standard IEEE 754

Arytmetyka liczb floating-point

Błąd względny i bezwzględny reprezentacji, ULP

Utrata cyfr znaczących

Pamięci

Pamięć statyczna

Pamięć dynamiczna

Fizyczna organizacja pamięci

Cykle dostępu do pamięci

Pamięci synchroniczne i asynchroniczne

Architektura współczesnych pamięci DDR

Hierarchia pamięci

Zasada lokalność danych
Hierarchia pamięci
Pamięć podręczna
Hit i miss
Czas dostępu do pamięci:
Strategie write-through i write-back

Pamięć podręczna

Architektury pamięci podręcznej
Architektura z adresowaniem asocjacyjnym
Architektura z adresowaniem z adresowaniem bezpośrednim
Architektura z adresowaniem zbiorowo asocjacyjnym
Optymalizacja parametrów pamięci podręcznej

Wirtualizacja pamięci

Adres fizyczny i logiczny
Organizacja pamięci wirtualnej
Segmentacja i stronicowanie pamięci
Dostęp blokowy do pamięci
Ochrona pamięci, menadżer pamięci
Tryb pracy rzeczywisty i wirtualny procesora

Magistrale procesora

Rola magistrali
Magistrala wewnętrzna i zewnętrzna
Standardy i typy magistral procesora
Przykłady standardów popularnych magistral
Współpraca procesora z magistralą
Bezpośredni dostęp do pamięci DMA

Ćwiczenia laboratoryjne

Moduły peryferyjne wejścia-wyjście (GPIO)

Mini wykład:

Wprowadzenie do programowania układów peryferyjnych na przykładzie GPIO

- Wprowadzenie do środowiska Keil uVision
- Architektura GPIO – rejestry sterujące, sposób aktywowania portu
- Multipleksacja wyprowadzeń mikrokontrolera
- Programowanie rejestrów urządzeń peryferyjnych – wskaźniki, definiowanie adresów urządzeń, struktury, unie
- Biblioteki programistyczne

Ćwiczenia praktyczne:

Wprowadzenie do środowiska Keil – obsługa GPIO

- Stworzenie pierwszego projektu w środowisku Keil
- Konfiguracja GPIO jako wyjścia na przykładzie migania diodą LED
- Konfiguracja GPIO jako wejścia na przykładzie obsługi przycisku
- Podłączenie GPIO do analizatora stanów logicznych i programowalnego generatora sygnałów

Podłączenie peryferiów zewnętrznych do systemu mikroprocesorowego

Mini wykład:

Podłączenie peryferiów do systemu mikroprocesorowego za pomocą standardowej magistrali peryferyjnej

- Podstawowe sposoby podłączenia modułów zewnętrznych do systemu

mikroprocesorowego

- Protokół komunikacji procesora z wyświetlaczem zewnętrznym
- Obsługa sterownika do wyświetlacza

Ćwiczenia praktyczne:

Human-Machine Interface 1 – podłączenie wyświetlacza do mikrokontrolera

- Wyświetlenie na wyświetlaczu "Hello World"
- Wizualizacja pracy przycisku i diody LED na wyświetlaczu
- Dodanie funkcjonalności do sterownika wyświetlacza

Przerwania w mikrokontrolerach

Mini wykład:

Przerwania i programowanie mikrokontrolerów oparte o przerwania

- Przerwania i wyjątki – kontroler przerwań, tablica przerwań, priorytety, maskowanie przerwań
- Tryby pobierania energii przez procesor
- Programowanie w oparciu o przerwania

Ćwiczenia praktyczne:

Human-Machine Interface 2 – podłączenie klawiatury do mikrokontrolera

- Podłączenie i sprawdzenie sposobu działania klawiatury
- Wykorzystanie przerwań do obsługi klawiatury
- Budowa prostego zegarka
- Program: kalkulator

Liczniki i układy czasowe

Mini wykład:

Moduły liczników i ich obsługa

- Moduł generatora sygnałów zegarowych – konfiguracja zegara mikrokontrolera
- Prosty układ licznika peryferyjnego – sposób wykorzystania i konfiguracji
- Zaawansowany układ czasowo-licznikowy

Ćwiczenia praktyczne:

Sonar – konwersja czasu na odległość

- Generowanie impulsów z wykorzystaniem różnych modułów mikrokontrolera, a następnie ich obserwacja przy pomocy oscyloskopu
- Generowanie impulsów z zewnętrznego generatora, a następnie ich pomiar przy pomocy mikrokontrolera
- Podłączenie ultradźwiękowego czujnika odległości i pomiar odległości

Przebiegi o modulowanej szerokości impulsu (PWM)

Mini wykład:

Generator sygnału TPM

- Konfiguracja i tryby pracy generatora TPM

Zajęcia praktyczne:

Sterowanie serwomechanizmem – konwersja sygnału na kąt wychylenia

- Generowanie impulsów PWM z wykorzystaniem mikrokontrolera
- Obserwacja sygnałów PWM za pomocą oscyloskopu
- Podłączenie serwomechanizmu do mikrokontrolera

Przetworniki analogowo-cyfrowe (A/C) i cyfrowo-analogowe (C/A)

Mini wykład:

Wbudowane przetworniki A/C i C/A i ich obsługa

- Podstawy konwersji sygnału analogowego na cyfrowy, błąd kwantyzacji, kryterium Nyquista
- Budowa podstawowych przetworników A/C i C/A
- Budowa przetwornika A/C w mikrokontrolerze, kalibracja, programowanie
- Budowa przetwornika C/A w mikrokontrolerze i jego programowanie

Zajęcia praktyczne:

- Obsługa wbudowanych przetworników A/C i C/A
- Odczyt temperatury rdzenia procesora
- Generowanie przebiegu SIN z wykorzystaniem mikrokontrolera, a następnie jego obserwacja przy pomocy oscyloskopu
- Konfiguracja ADC i odczyt wartości wygenerowanej z DAC

Asynchroniczna komunikacja szeregową (UART)

Mini wykład:

- Podstawy komunikacji UART (ramka, bity startu/stopu, kontrolne)
- Budowa układu nadawczo-odbiorczego UART w mikrokontrolerze. Funkcja rejestrów
- Programowanie aplikacji do komunikacji za pomocą UART. Wykorzystanie przerwań i buforów kołowych

Ćwiczenia praktyczne:

Komunikacja mikrokontrolera z komputerem za pomocą interfejsu UART

- Moduły pośredniczące, konwersja napięć, konfiguracja komputera, program terminala
- Konfiguracja mikrokontrolera
- Aplikacja "Hello, world"

Ćwiczenia projektowe

Ćwiczenia projektowe 10 h

Studenci realizują projekt prostego systemu mikroprocesorowego z wykorzystaniem typowych układów peryferyjnych

Metody i techniki kształcenia:

Wykład: Treści prezentowane na wykładzie są przekazywane w formie prezentacji multimedialnej w połączeniu z klasycznym wykładem tablicowym wzbogaconymi o pokazy odnoszące się do prezentowanych zagadnień.

Ćwiczenia laboratoryjne: W trakcie zajęć laboratoryjnych studenci samodzielnie rozwiązują zadany problem praktyczny, dobierając odpowiednie narzędzia. Prowadzący stymuluje grupę do refleksji nad problemem, tak by otrzymane wyniki miały wysoką wartość merytoryczną.

Ćwiczenia projektowe: Studenci wykonują zadany projekt samodzielnie, bez większej ingerencji prowadzącego. Ma to wykształcić poczucie odpowiedzialności za pracę w grupie oraz odpowiedzialności za podejmowane decyzje.

Warunki i sposób zaliczenia poszczególnych form zajęć, w tym zasady zaliczeń poprawkowych, a także warunki dopuszczenia do egzaminu:

Nie określono

Zasady udziału w poszczególnych zajęciach, ze wskazaniem, czy obecność studenta na zajęciach jest obowiązkowa:

Wykład:

- Obecność obowiązkowa: Nie
- Zasady udziału w zajęciach: Studenci uczestniczą w zajęciach poznając kolejne treści nauczania

zgodnie z sylabusem przedmiotu. Studenci winni na bieżąco zadawać pytania i wyjaśniać wątpliwości. Rejestracja audiowizualna wykładu wymaga zgody prowadzącego.

Ćwiczenia laboratoryjne:

- Obecność obowiązkowa: Tak

- Zasady udziału w zajęciach: Studenci wykonują ćwiczenia laboratoryjne zgodnie z materiałami udostępnionymi przez prowadzącego. Student jest zobowiązany do przygotowania się w przedmiocie wykonywanego ćwiczenia, co może zostać zweryfikowane kolokwium w formie ustnej lub pisemnej. Zaliczenie zajęć odbywa się na podstawie zaprezentowania rozwiązania postawionego problemu. Zaliczenie modułu jest możliwe po zaliczeniu wszystkich zajęć laboratoryjnych.

Ćwiczenia projektowe:

- Obecność obowiązkowa: Tak

- Zasady udziału w zajęciach: Studenci wykonują prace praktyczne mające na celu uzyskanie kompetencji zakładanych przez sylabus. Ocenie podlega sposób wykonania projektu oraz efekt końcowy.

Sposób obliczania oceny końcowej

Na zajęciach laboratoryjnych oceniane jest przygotowanie do zajęć i aktywność.

Szczegółowe zasady oceniania pracy na zajęciach laboratoryjnych podaje prowadzący grupę laboratoryjną.

Projekt oceniany jest na podstawie stopnia zaawansowania podjętego tematu, sposobu wykonania projektu, efektu końcowego i terminowości realizacji. Szczegółowe zasady ustala prowadzący grupę projektową.

W ocenie końcowej są uwzględniane oceny cząstkowe ze:

a) egzaminu (waga 1/3)

b) ćwiczeń laboratoryjnych (waga 1/3)

c) projektu (waga 1/3)

Sposób i tryb wyrównywania zaległości powstałych wskutek nieobecności studenta na zajęciach:

Nie określono

Wymagania wstępne i dodatkowe, z uwzględnieniem sekwencyjności modułów

Znajomość podstawowych zagadnień z techniki cyfrowej

Znajomość podstaw języka programowania C oraz zasad tworzenia algorytmów

Znajomość prostych układów analogowych współpracujących z mikrokontrolerami (np.: wzmacniacze operacyjne, komparatory, obwody z diodami LED)

Podstawy systemu operacyjnego Windows

Zalecana literatura i pomoce naukowe

WWW

1. strona firmowa www.nxp.com

2. strona firmowa www.intel.com

3. strona firmowa www.arm.com

Wykład

1. P. Metzger: Anatomia PC, Helion, Gliwice 2009

2. W. Stallings: Computer Organization and Architecture

3. D.A. Patterson, J.N. Hennessy: Computer Organization and Design. The Hardware Software Interface,

4. F. Vahid, T. Givargis: Embedded System Design. A Unified Hardware/Software Introduction

5. D.M. Harris, S.L. Harris: Digital Design and Computer Architecture

Laboratorium

1. R. Baranowski: Mikrokontrolery AVR ATmega w praktyce

2. Alexander G. Dean: Embedded Systems Fundamentals with Arm Cortex-M based Microcontrollers: A Practical Approach

3. W. Mielczarek: Szeregowe interfejsy cyfrowe, Helion, Gliwice 1994

4. Z. Hajduk: Mikrokontrolery w systemach zdalnego sterowania, BTC, Warszawa 2005

Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu

Russek P., et al. A custom co-processor for the discovery of low autocorrelation binary sequences. Measurement Automation Monitoring, 2016, 62.

Kowalczyk, K., Wozniak, S., Chyrowicz, T., & Rumian, R. (2016, September). Embedded system for acquisition and enhancement of audio signals. In Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), 2016 (pp. 68-71). IEEE.

Marszałek K., Rumian R., Układ analizy samochodowego sterownika pokładowego — [Analysis system for board computer investigation], KKE '2003 : II Krajowa Konferencja Elektroniki : Kołobrzeg 9-12 czerwca 2003 : materiały konferencji. T. 2/2. — Koszalin : Wydział Elektroniki Politechniki Koszalińskiej, 2003 + CD-ROM. — S. 639-644. — Bibliogr. s. 644, Streszcz.

Informacje dodatkowe

Brak