

Sylabus modułu kształcenia/przedmiotu

Nr pola	Nazwa pola	Opis
1	Jednostka	Instytut Politechniczny
2	Kierunek studiów	Elektrotechnika (studia stacjonarne)
3	Nazwa modułu kształcenia/przedmiotu	Podstawy Techniki Mikroprocesorowej
4	Kod modułu kształcenia/przedmiotu	
5	Kod Erasmusa	
6	Punkty ECTS	3
7	Rodzaj modułu	Obowiązkowy
8	Rok studiów	2
9	Semestr	3
10	Typ zajęć	Stacjonarne
11	Liczba godzin	15W, 30LO
12	Koordinator	Tomasz Drabek
13	Prowadzący	
14	Język wykładowy	Polski
15	Zakres nauk podstawowych	Nie
16	Zajęcia ogólnouczelniane/na innym kierunku	Nie
17	Wymagania wstępne	Wymagane są wiadomości z zakresu działania podstawowych układów cyfrowych (<i>TTL</i>).
18	Efekty kształcenia	<ul style="list-style-type: none"> Definiuje pojęcie mikroprocesora i podstawowe pojęcia z nim związane, w tym wielkości charakteryzujące mikroprocesor. Opisuje różnice pomiędzy architekturą von Neumanna a architekturą typu Harvard. Charakteryzuje architektury pokrewne do architektury von Neumanna. Opisuje elementy składowe mikroprocesora w architekturze von Neumanna i ich funkcje. Definiuje pojęcie przerwania sprzętowego i jego wykorzystanie w aplikacjach mikroprocesorów. Potrafi zbudować jednostkę arytmetyczno-logiczną (CPU) procesora z układów cyfrowych. Opisuje rolę pamięci stosowej w funkcjonowaniu mikroprocesora. Zna historię rozwoju mikroprocesorów i dysponuje podstawowymi informacjami dot. stosowanych technologii ich wykonania (historycznie i aktualnie). Rozróżnia architekturą von Neumanna od innych. Określa możliwości czasowo-obliczeniowe mikroprocesora na podstawie jego danych katalogowych. Buduje jednostkę arytmetyczno-logiczną (CPU) procesora z układów cyfrowych o średniej skali integracji. Opisuje architekturę i działanie prostego mikrokomputera. Poprawnie konfiguruje projektowany prosty mikrokomputer, na poziomie dokonania odpowiedniego doboru jego podzespołów i wyboru sposobu ich połączenia i komunikacji z mikroprocesorem. Definiuje pojęcie mikrokontrolera (mikrokomputera jednoukładowego). Charakteryzuje bazową architekturę 8-bitowych mikrokontrolerów rodziny ATmega firmy Atmel (na przykładzie ATmega168 i/lub ATmega328 i/lub ATmega1280). Opisuje przebieg cyklu rozkazowego jednostki ATmega. Opisuje mapy pamięci wybranego układu ATmega. Wymienia, jakie peryferia mikroprocesora mogą być zintegrowane w architekturze ATmega i w jaki sposób są one konfigurowalne. Konfiguruje podzespoły wybranego mikrokontrolera ATmega do określonych trybów pracy, za pomocą rejestrów sterujących mikrokontrolera lub za pomocą odpowiednich sterowników programowych. Charakteryzuje typowe układy wejściowe, stosowane w prostych

		<p>mikrokomputerach budowanych do celów przemysłowych: bramy, przetworniki A/C, przetworniki U/f, U/URMS, klawiatury. Charakteryzuje również typowe układy wyjściowe: zatrzaski, przetworniki C/A, f/U, wyświetlacze diodowe i wyświetlacze LCD alfanumeryczne i graficzne, wyjścia PWM.</p> <ul style="list-style-type: none"> • Odpowiednio dobiera i łączy z mikroprocesorem lub mikrokontrolerem typowe układy we/wy. • Opisuje szczegółowo zasady budowy prostych mikrokomputerów opartych o mikrokontrolery rodziny ATmega. Poznaje część sprzętową platformy sprzętowo-programowej Arduino. • Projektuje (w sensie sprzętowym) mikrokomputer zbudowany wokół mikrokontrolera rodziny ATmega. • Programuje w języku C, w zakresie używanym do programowania jednostek centralnych ATmega platformy Arduino. Opisuje część programową platformy sprzętowo-programowej Arduino. • Programuje mikrokontrolery rodziny ATmega w języku C. Posługuje się oprogramowaniem narzędziowym na komputery PC, wspomagającym proces uruchamiania własnych aplikacji na platformie Arduino. • Opisuje zasady realizacji programowej na mikrokontrolerach rodziny ATmega algorytmów przetwarzania sygnałów, na przykładach algorytmów stosowanych w urządzeniach automatyki. • Implementuje na mikrokontrolerach rodziny ATmega algorytmy przetwarzania sygnałów, stosowane w urządzeniach sterowania przemysłowego.
19	Stosowane metody dydaktyczne	Tradycyjny wykład (tablica, kreda) wspomagany wyświetlanymi schematami układów, tabelami i zdjęciami. Laboratorium o charakterze programistycznym – programowanie mikrokontrolerów rodziny ATmega firmy Atmel, w języku C, na platformie Arduino.
20	Metody sprawdzania i kryteria oceny efektów kształcenia	Praca kontrolna pisemna, pytania kontrolne w laboratorium, Zaliczanie sprawozdań z zajęć laboratoryjnych
21	Forma i warunki zaliczenia	<ol style="list-style-type: none"> 1. Aby uzyskać pozytywną ocenę końcową niezbędne jest uzyskanie pozytywnych ocen ze wszystkich ćwiczeń laboratoryjnych i zdanie egzaminu końcowego. 2. Ocena końcowa (OK) jest obliczana z oceny zaliczenia zajęć laboratoryjnych (OI) i oceny z pracy kontrolnej pisemnej (Ok). 3. Podstawą ustalenia OK jest liczba Wl obliczona z wzoru: $Wl = 0,8 * OI + 0,2 * Ok$. <p>W zależności od wartości liczbowej Wl ocena końcowa jest ustalana w sposób zgodny z Regulaminem Studiów w PWSZ w Tarnowie</p>
22	Treści kształcenia (skrócony opis)	Mikroprocesory i mikrokontrolery – pojęcia podstawowe, wielkości charakteryzujące, architektury, funkcjonowanie, pojęcia powiązane. Otoczenie mikroprocesora/mikrokontrolera – pamięci, układy wejścia/wyjścia, układy towarzyszące. 8-bitowe mikrokontrolery rodziny ATmega - architektura, funkcjonowanie i programowanie. Mikrokomputer - architektura, budowa i funkcjonowanie. Język C jako język programowania mikrokontrolerów rodziny ATmega na platformie Arduino. Środki techniczne wspomagające programowanie i uruchamianie układów mikroprocesorowych. Przykłady zastosowań techniki mikroprocesorowej w urządzeniach automatyki przemysłowej.
23	Treści kształcenia (pełny opis)	<p>W ramach modułu zajęcia są prowadzone w formie wykładu (15 godzin) i zajęć laboratoryjnych (30 godzin).</p> <p>WYKŁADY (15 godz.):</p> <ol style="list-style-type: none"> 1. Zagadnienia wstępne. Pojęcie mikroprocesora i podstawowe pojęcia z nim związane, w tym wielkości charakteryzujące mikroprocesor. Mikroprocesor jako automat skończony. Architektura von Neumanna i jej elementy składowe. Funkcjonowanie mikroprocesora w architekturze von Neumanna. Architektury pokrewne. Architektura typu Harvard i jej

funkcjonowanie. Historia rozwoju mikroprocesorów i podstawowe informacje dot. stosowanych technologii ich wykonania. (4 godz.).

2. Architektura i działanie mikrokomputera. Typowe podzespoły prostego mikrokomputera przeznaczonego do celów sterowania przemysłowego i ich połączenie z mikroprocesorem. Funkcjonowanie takiego mikrokomputera. (2 godz.).

3. Mikrokomputery jednoukładowe. Pojęcie mikrokomputera jednoukładowego (mikrokontrolera). Architektura 8-bitowych mikrokontrolerów rodziny ATmega firmy Atmel. Przebieg cyklu rozkazowego wybranej jednostki rodziny ATmega, jej mapa pamięci, stos, podzespoły funkcjonalne i ich konfiguracja za pomocą rejestrów sterujących. (4 godz.)

4. Układy wejścia/wyjścia mikrokomputera. Typowe układy wejściowe, stosowane w prostych mikrokomputerach budowanych do celów przemysłowych: bramy, przetworniki A/C, przetworniki U/f, U/URMS, klawiatury. Typowe układy wyjściowe: zatraski, przetworniki C/A, f/U, wyświetlacze diodowe i wyświetlacze LCD alfanumeryczne i graficzne, wyjścia PWM. (3 godz.)

5. Implementacja algorytmów przetwarzania sygnałów na mikrokontrolerach ATmega. Zasady implementacji algorytmów przetwarzania sygnałów na mikrokontrolerach ATmega na przykładach algorytmów stosowanych w urządzeniach sterowania przemysłowego. (2 godz.)

LABORATORIUM (30 godz.):

1. Platforma Arduino - wprowadzenie, realizacje sekwencyjnego sterowania binarnego. W ćwiczeniu Studenci zapoznają się z zestawami uruchomieniowymi platformy Arduino i oprogramowaniem narzędziowym oraz piszą i uruchamiają proste programy w języku C, realizujące uwarunkowane czasowo sterowania binarne. (3 godz.)

2. Obsługa programowa wyświetlaczy alfanumerycznych typu LED i typu LCD. Ćwiczący mają za zadanie napisanie w języku C i uruchomienie na platformie Arduino driver'a wyświetlacza alfanumerycznego LCD. Dodatkowo, osoby zainteresowane, mogą napisać i uruchomić driver 4-pozycyjnego, 7-segmentowego wyświetlacza cyfrowego LED. (3 godz.)

3. Obsługa programowa klawiatur sekwencyjnych i matrycowych. Ćwiczący mają za zadanie napisanie w języku C i uruchomienie na platformie Arduino driver'a klawiatury. (3 godz.)

4. Generacja i filtracja sygnałów z użyciem mikrokontrolera rodziny ATmega. W ćwiczeniu studenci piszą i uruchamiają na platformie Arduino 2 programy w języku C. Pierwszy z nich realizuje generację zadanych sygnałów, z użyciem przetwornika C/A. Drugi ma stanowić implementację na mikrokontrolerze rodziny ATmega wybranego filtra typu FIR, z użyciem przetwornika A/C i przetwornika C/A. (3 godz.)

5. Mikroprocesorowa realizacja regulatora PID. Ćwiczący mają za zadanie napisanie w języku C i uruchomienie na platformie Arduino programu realizującego algorytm regulatora PID. (3 godz.)

6. Obsługa karty pamięci typu SD. Ćwiczący mają za zadanie napisanie w języku C i uruchomienie na platformie Arduino programu umożliwiającego zapis/odczyt danych na karcie pamięci typu Secure Digital.

7. Sterowanie silnikiem skokowym. Ćwiczący mają za zadanie napisanie w języku C i uruchomienie na platformie Arduino programu realizującego, za pośrednictwem odpowiedniego driver'a do zasilania faz silnika, sterowanie fazami silnika skokowego.

8. Mikroprocesorowa realizacja wybranego algorytmu przetwarzania sygnałów. Ćwiczący mają za zadanie napisanie w języku C i uruchomienie na platformie Arduino programu realizującego wybrany algorytm przetwarzania sygnałów, stosowany w urządzeniach sterowania przemysłowego (na podstawie wykładu). (3 godz.)

Pozostałe 6 godzin wykorzystywane jest do przyjmowania sprawozdań studenckich z wykonanych ćwiczeń.

24	Literatura podstawowa i uzupełniająca	<p><u>Podstawowa:</u></p> <ol style="list-style-type: none"> 1. Rafał Baranowski: Mikrokontrolery AVR ATmega w praktyce 2. Krzysztof Badźmirowski: Układy i systemy mikroprocesorowe. 3. Tomasz Francuz: Język C dla mikrokontrolerów AVR. Od podstaw do zaawansowanych aplikacji. <p><u>Pomocnicza:</u></p> <ol style="list-style-type: none"> 1. Ryszard Pełka: Mikrokontrolery. Architektura, programowanie, zastosowania. 2. Tom Igoe: Spraw, by rzeczy przemówiły. Programowanie urządzeń elektronicznych z wykorzystaniem Arduino. 3. Jon Lazar: Arduino i projekty Lego.
25	Przyporządkowanie modułu kształcenia/przedmiotu do obszaru/ obszarów kształcenia	Obszar nauk technicznych
26	Sposób określenia liczby punktów ECTS	<p>Udział w wykładach 15h, Samodzielne studiowanie materiału wykładów 15h, Udział w zajęciach laboratoryjnych 30h, Przygotowanie sprawozdań z zajęć laboratoryjnych 15h, Przygotowanie do pracy kontrolnej 5h, Sumaryczne obciążenie pracą studenta 80h przeliczone ma 3 punkty ECTS</p>
27	Liczba punktów ECTS – zajęcia wymagające bezpośredniego udziału nauczyciela akademickiego	2
28	Liczba punktów ECTS – zajęcia o charakterze praktycznym	2