



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Interfejsy systemów wbudowanych [N2Inf1-AMiWdIP>ISW]

### Przedmiot

Kierunek studiów  
Informatyka

Rok/Semestr  
2/3

Studia w zakresie (specjalność)  
Aplikacje mobilne i wbudowane dla Internetu  
Przedmiotów

Profil studiów  
ogólnoakademicki

Poziom studiów  
drugiego stopnia

Język oferowanego przedmiotu  
polski

Forma studiów  
niestacjonarne

Wymagalność  
obieralny

### Liczba godzin

Wykład  
16

Laboratorium  
16

Inne  
0

Ćwiczenia  
0

Projekty/seminaria  
0

### Liczba punktów ECTS

3,00

### Koordynatorzy

dr inż. Zygmunt Kubiak

### Wykładowcy

### Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z fizyki, elektroniki, techniki cyfrowej i analogowej. Powinien posiadać umiejętność rozwiązywania podstawowych problemów z zakresu elektrotechniki i elektroniki, programowania w języku C, tworzenia algorytmów działania aplikacji oraz umiejętność pozyskiwania informacji ze wskazanych źródeł. Powinien również mieć gotowość do podjęcia współpracy w ramach zespołu. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.

### Cel przedmiotu

1.Przekazanie studentom podstawowej wiedzy z zakresu wybranych protokołów transmisji przewodowej i bezprzewodowej, ze szczególnym uwzględnieniem interfejsów mikrokontrolerów, sensorów układów wykonawczych. 2.Przekazanie studentom uzupełniającej wiedzy z zakresu organizacji protokołów, technicznej realizacji transmisji, rozwiązań sprzętowych i programowych modułów sieci (węzłów), bezpieczeństwa transmisji, zastosowań. 3.Rozwijanie u studentów umiejętności rozwiązywania prostych problemów związanych z budową, działaniem, programowaniem, uruchamianiem sieci. 4.Kształtowanie u studentów umiejętności pracy zespołowej w ramach zadań realizowanych w laboratorium

## Przedmiotowe efekty uczenia się

### Wiedza:

1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie wybranych działów elektroniki - [K2st\_W2]
2. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie architektury systemów komputerowych i sieci komputerowych - [K2st\_W3]
3. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w rozwoju mikroelektroniki, nanotechnologii w szczególności mikrokontrolerów, sensorów, systemów wbudowanych - [K2st\_W4]
4. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich w zakresie systemów wbudowanych na etapie projektowania, budowy i programowania - [K2st\_W6]

### Umiejętności:

1. potrafi krytycznie korzystać z informacji literaturowych, baz danych i innych źródeł w języku polskim i obcym, z zakresu wybranej grupy protokołów transmisji - [K2st\_U1]
2. potrafi planować i przeprowadzać eksperymenty wybranej grupy interfejsów, w tym pomiary i symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski oraz formułować i weryfikować hipotezy związane ze złożonymi problemami inżynierskimi i prostymi problemami badawczymi - [K2st\_U3]
3. potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych, dotyczące interfejsów systemów wbudowanych, metody analityczne, symulacyjne oraz eksperymentalne - [K2st\_U4]
4. potrafi - przy formułowaniu i rozwiązywaniu zadań inżynierskich dotyczących interfejsów systemów wbudowanych - integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne - [K2st\_U5]
5. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych w dziedzinie interfejsów systemów wbudowanych - [K2st\_U6]
6. potrafi - stosując m.in. koncepcyjnie nowe metody - rozwiązywać złożone zadania informatyczne, w tym zadania nietypowe oraz zadania zawierające komponent badawczy w zakresie interfejsów systemów wbudowanych - [K2st\_U10]
7. potrafi określić kierunki dalszego uczenia się i zrealizować proces samokształcenia, w tym innych osób w dziedzinie interfejsów systemów wbudowanych - [K2st\_U16]

### Kompetencje społeczne:

1. rozumie potrzebę ciągłego dokształcania się, rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K2st\_K1]
2. rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu informatyki w rozwiązywaniu problemów badawczych i praktycznych w zakresie interfejsów systemów wbudowanych - [K2st\_K2]

## 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 laboratoriów: - na podstawie oceny bieżącego postępu realizacji zadań,

### Ocena podsumowująca:

- a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez: - ocenę wiedzy i umiejętności wykazanych na kolokwium o charakterze problemowym, złożonym z zadań problemowych wybranych z listy zagadnień udostępnionej wcześniej studentom (5 pytań z 20 zagadnień problemowych); - omówienie wyników i w wątpliwych przypadkach indywidualnych dodatkowe pytania kontrolne,
- b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez: - ocenę umiejętności związanych z realizacją ćwiczeń laboratoryjnych; - ocenianie ciągłe, na każdych zajęciach (odpowiedzi ustne); - premiowanie przyrostu umiejętności posługiwania się poznanymi zasadami i metodami; - ocenę sprawozdań przygotowywanych z wybranych zagadnień realizowanych w ramach laboratorium; ocena ta obejmuje także umiejętność pracy w zespole.

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za: - omówienia dodatkowych aspektów zagadnienia; - efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu; - umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium; - uwagi związane z udoskonaleniem materiałów dydaktycznych; - wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.

## Treści programowe

Program wykładu obejmuje następujące zagadnienia:

Wprowadzenie do protokołów i interfejsów cyfrowych. Zagadnienia związane są z interfejsami, które sprzętowo obsługuje mikrokontroler oraz pozostałe interfejsy, które wykorzystywane są w systemach wbudowanych. Zabezpieczenia transmisji w sieciach przewodowych i bezprzewodowych. Metody zapewnienia integralności przesyłanych danych (kodowanie nadmiarowe, CRC). Protokoły sieci przewodowych. Wybrane sieci miejscowe (ang. Fieldbus), np. Modbus, Hart, MBus, CAN, LIN, Ethernet. Wybrane bezprzewodowe sieci małej mocy w tym LPWAN, np. ZigBee, SimpliciTI, LoRA, Sigfox. Zagadnienie energooszczędności.

Interfejsy transmisji szeregowej wbudowane w mikrokontrolery oraz stosowane do komunikacji na bliskie odległości (między układami scalonymi oraz modułami). Protokoły i rozwiązania układowe standardów transmisji szeregowej: UART (RS232, RS422, RS485), USB, IIC, SMBus, SPI, Microware, 1-Wire). Niestandardowe interfejsy wybranych sensorów i układów wykonawczych.

Wprowadzenie do wybranych mikrokontrolerów (jeśli niezbędne). Programowanie interfejsów transmisji szeregowej w języku C dla wybranych mikrokontrolerów. Programowe zapewnienie integralności ramki transmisyjnej ? kodowanie nadmiarowe, CRC8, CRC16, CRC32.

Przykłady i zasady obsługi programowej układów wyposażonych w magistrale szeregowo (sensory, wyświetlacze LCD, przetworniki AC/CA, pamięci itp.). Czas życia produktu.

Zajęcia laboratoryjne prowadzone są w formie siedmiu 2-godzinnych ćwiczeń, odbywających się w laboratorium, poprzedzonych 2-godzinną sesją instruktażową na początku semestru. Ćwiczenia realizowane są przez 2-osobowe zespoły studentów. Program laboratorium obejmuje następujące zagadnienia:

Wprowadzenie do uruchamiania aplikacji na modułach wybranych modułach rozwojowych z mikrokontrolerami, np. firmy Atmel typu ATmega32 (moduł Arduino), firmy Silicon Labs typu Toolstick UNI DC z mikrokontrolerami C8051F020, lub firmy Texas Instruments typu MSP430, Tiva Launchpad lub firmy STM typu Nucleo. Konfiguracja mikrokontrolera. Realizacja języku C prostych programów (jeśli niezbędne) typu sterowanie diodami LED, obsługa przycisków. Programy wykorzystujące przetwarzanie AC i CA. Realizacja transmisji UART, IIC, SPI a także niestandardowych interfejsów wybranych sensorów. Programowanie mikrokontrolerów dla systemów czasu rzeczywistego zgodnie z zasadami przedstawionymi na wykładach dla różnych klas mikrokontrolerów (8-, 16-, 32-bitowych) i różnych narzędzi programistycznych.

## Tematyka zajęć

Program wykładu obejmuje następujące zagadnienia:

Wprowadzenie do protokołów i interfejsów cyfrowych. Zagadnienia związane są z interfejsami, które sprzętowo obsługuje mikrokontroler oraz pozostałe interfejsy, które wykorzystywane są w systemach wbudowanych. Zabezpieczenia transmisji w sieciach przewodowych i bezprzewodowych. Metody zapewnienia integralności przesyłanych danych (kodowanie nadmiarowe, CRC). Protokoły sieci przewodowych. Wybrane sieci miejscowe (ang. Fieldbus), np. Modbus, Hart, MBus, CAN, LIN, Ethernet. Wybrane bezprzewodowe sieci małej mocy w tym LPWAN, np. ZigBee, SimpliciTI, LoRA, Sigfox. Zagadnienie energooszczędności.

Interfejsy transmisji szeregowej wbudowane w mikrokontrolery oraz stosowane do komunikacji na bliskie odległości (między układami scalonymi oraz modułami). Protokoły i rozwiązania układowe standardów transmisji szeregowej: UART (RS232, RS422, RS485), USB, IIC, SMBus, SPI, Microware, 1-Wire). Niestandardowe interfejsy wybranych sensorów i układów wykonawczych.

Wprowadzenie do wybranych mikrokontrolerów (jeśli niezbędne). Programowanie interfejsów transmisji szeregowej w języku C dla wybranych mikrokontrolerów. Programowe zapewnienie integralności ramki transmisyjnej ? kodowanie nadmiarowe, CRC8, CRC16, CRC32.

Przykłady i zasady obsługi programowej układów wyposażonych w magistrale szeregowo (sensory, wyświetlacze LCD, przetworniki AC/CA, pamięci itp.). Czas życia produktu.

Zajęcia laboratoryjne prowadzone są w formie siedmiu 2-godzinnych ćwiczeń, odbywających się w

laboratorium, poprzedzonych 2-godzinną sesją instruktażową na początku semestru. Ćwiczenia realizowane są przez 2-osobowe zespoły studentów. Program laboratorium obejmuje następujące zagadnienia:

Wprowadzenie do uruchamiania aplikacji na modułach wybranych modułów rozwojowych z mikrokontrolerami, np. firmy Atmel typu ATmega32 (moduł Arduino), firmy Silicon Labs typu Toolstick UNI DC z mikrokontrolerami C8051F020, lub firmy Texas Instruments typu MSP430, Tiva Launchpad lub firmy STM typu Nucleo. Konfiguracja mikrokontrolera. Realizacja języku C prostych programów (jeśli niezbędne) typu sterowanie diodami LED, obsługa przycisków. Programy wykorzystujące przetwarzanie AC i CA. Realizacja transmisji UART, IIC, SPI a także niestandardowych interfejsów wybranych sensorów. Programowanie mikrokontrolerów dla systemów czasu rzeczywistego zgodnie z zasadami przedstawionymi na wykładach dla różnych klas mikrokontrolerów (8-, 16-, 32-bitowych) i różnych narzędzi programistycznych.

## Metody dydaktyczne

1. wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy,
2. ćwiczenia laboratoryjne: prezentacja problematyki ćwiczeń, realizacja zagadnień przedstawionych w treściach programowych ćwiczeń laboratoryjnych,
3. konsultacje z zakresu realizowanych ćwiczeń

## Literatura

### Podstawowa

1. Projektowanie systemów mikroprocesorowych, Hadam P., BTC, Warszawa, 2004
2. Embedded Systems: Real-Time Interfacing to ARM? Cortex-M? Microcontrollers - Volume 2, Valvano J.W., Jonathan W. Valvano 2013; ISBN: 978-1463590154
3. Programowanie mikrokontrolerów 8051 w języku C w praktyce, Bogusz J., BTC, Warszawa, 2005
4. Mikrokontrolery STM32 w praktyce, Paprocki K., BTC, Warszawa, 2009
5. Prezentacje do wykładów

### Uzupełniająca

1. Microcontrollers in practice, Mitescu M., Susnea I., Springer, Berlin, 2005
2. Lokalne interfejsy szeregowy w systemach cyfrowych, Bogusz J., BTC, Warszawa, 2004
3. Źródła internetowe, np. [www.silabs.com](http://www.silabs.com), [www.atmel.com](http://www.atmel.com), [www.ti.com](http://www.ti.com), [www.st.com](http://www.st.com)
4. Embedded programming, Chew M.T., Gupta G.S., Silicon laboratories, 2005

## Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	75	3,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	32	1,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	43	1,50