

Microinformatic systems programming - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Microinformatic systems programming
Kod przedmiotu	11.3-WE-INF-D-MicroinSP-Er
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	Program Erasmus drugiego stopnia
Semestr rozpoczęcia	semestr zimowy 2020/2021

Informacje o przedmiocie	
Semestr	3
Liczba punktów ECTS do zdobycia	6
Typ przedmiotu	obowiązkowy
Język nauczania	angielski
Sylabus opracował	dr inż. Mirosław Kozioł

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	30	2	-	-	Zaliczenie na ocenę
Laboratorium	30	2	-	-	Zaliczenie na ocenę

Cel przedmiotu

- To provide knowledge about the basic peripherals that occur in microinformatic systems and methods of its handling.
- To develop and shape the skills in the software design for microinformatic systems.

Wymagania wstępne

By entering this course, student should know the following issues:

- fundamentals of programming in the C language,
- fundamental knowledge about analog and digital circuits,
- fundamental knowledge about analog-to-digital and digital-to-analog converters.

Zakres tematyczny

Microinformatic system. Basic components of a microprocessor system. Microinformatic system and microprocessor system. Microcontroller as an example of a microinformatic system.

Cortex-M7 processor. The family of Cortex processors. Architecture of the Cortex-M7 processor: basic functional blocks, buses, programmer's model.

STM32F7 microcontrollers as an example of advanced microinformatic system with Cortex-M7 processor. Evolution of STM32 microcontrollers. Architecture of STM32F7 microcontrollers. Memory map. Available peripherals.

Software development for microinformatic systems. Software development flow. Creating the code. CMSIS - Cortex Microcontroller Software Interface Standard. Design of the program template for Cortex-M microcontroller according to CMSIS. Tools that support code creation for STM32F7 microcontrollers: STM32CubeMX configurator and Atollic TrueSUDIO for STM32 environment.

Block of clock signal generation in STM32F7 microcontrollers. Available sources of clock signals. Basic configuration registers. Block configuration using the STM32CubeMX program.

General purpose inputs-outputs. I/O ports in STM32F7 microcontrollers. Basic modes of operation of port lines. Configuration of port lines with direct use of registers and through the STM32CubeMX.

Exceptions. The concept of an exception in STM32F7 microcontrollers. NVIC interrupt controller. Interrupt priority. The group priority and subpriority of exceptions. Exception handling. Interrupts and events generated on port lines.

I2C interface as an example of a local serial interface. Basic interface characteristics. Interface configuration in STM32F7 microcontrollers. The basic functions of the HAL library to support the interface.

Timers. Main blocks of timers in STM32F7 microcontrollers. Configuration of timers using the STM32CubeMX. Examples of practical use of timers.

Cooperation of the microcontroller with analog signals. Configuration and operation of analog-to-digital (ADC) and digital-to-analog (DAC) converters in STM32F7 microcontrollers. Examples of practical use of ADC and DAC.

Metody kształcenia

- Lecture: conventional/traditional lecture.
- Laboratory: laboratory exercises.

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student can write a program for a microcontroller to create an advanced microinformatic system designed for the implementation of selected services and applications.		• obserwacje i ocena umiejętności praktycznych studenta	• Laboratorium
Student is able to use advanced techniques, methods and tools to implement software for microinformatic systems.		• obserwacje i ocena umiejętności praktycznych studenta	• Laboratorium
Student is able to select and write a program for the proper microcontroller peripherals in order to implement the appropriate functionality of the microinformatic system.		• obserwacje i ocena umiejętności praktycznych studenta • test z pytaniami zamkniętymi i otwartymi	• Wykład • Laboratorium

Warunki zaliczenia

Lecture: to receive a final passing grade student has to receive positive grade from final test.

Laboratory: to receive a final passing grade student has to receive positive grades in all laboratory exercises provided for in the laboratory syllabus.

Calculation of the final grade = lecture 45% + laboratory 55%

Literatura podstawowa

1. Yiu J.: The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors, Elsevier Science & Technology, 2011.
2. Martin T.: The Designer's Guide to the Cortex-M Processor Family, Elsevier Science & Technology, 2016.

Literatura uzupełniająca

Uwagi

Zmodyfikowane przez dr inż. Mirosław Kozioł (ostatnia modyfikacja: 26-04-2020 12:21)

Wygenerowano automatycznie z systemu SylabUZ