

**AGH**AGH UNIVERSITY OF SCIENCE  
AND TECHNOLOGY

Nazwa modułu: Techniki mikroprocesorowe

Rok akademicki: 2016/2017    Kod: JIS-2-202-GK-s    Punkty ECTS: 5

Wydział: Fizyki i Informatyki Stosowanej

Kierunek: Informatyka Stosowana    Specjalność: Grafika komputerowa i przetwarzanie obrazów

Poziom studiów: Studia II stopnia    Forma i tryb studiów: Stacjonarne

Język wykładowy: Polski    Profil kształcenia: Ogólnoakademicki (A)    Semestr: 2

Strona www: —

Osoba odpowiedzialna: dr inż. Świątek  
Krzysztof (swientek@agh.edu.pl)

Osoby prowadzące: dr inż. Skoczeń  
Andrzej (skoczen@fis.agh.edu.pl)  
dr inż. Świątek  
Krzysztof (swientek@agh.edu.pl)  
dr inż. Moron Jakub (jmoron@agh.edu.pl)

### Krótką charakterystyka modułu

Moduł pozwala na zapoznanie się z omniennym od klasycznego (ogólnie pojęte komputery) wykorzystaniem mikroprocesorów od implementowania ich w układach FPGA do użycia w mikrokontrolerach.

### Opis efektów kształcenia dla modułu zajęć

Kod EKM	Student, który zaliczył moduł zajęć wie/umie/potrafi	Powiązania z EKK	Sposób weryfikacji efektów kształcenia (forma zaliczeń)
Wiedza			
M_W001	Student zna budowę wewnętrzną wybranego mikroprocesora/mikrokontrolera oraz jego assemblera. Potrafi również wyjaśnić działanie i wskazać zastosowanie poszczególnych rozkazów.	IS2A_W11	Aktywność na zajęciach, Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych
M_W002	Student zna i potrafi scharakteryzować wybrane interfejsy komunikacyjne używane w systemach mikroprocesorowych.	IS2A_W10	Aktywność na zajęciach, Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych

M_W003	Student potrafi wyjaśnić proces „cross”-kompilacji i wskazać narzędzia niezbędne do pracy z mikroprocesorem/mikrokontrolerem pracującym w systemie wbudowanym.	IS2A_W16	Aktywność na zajęciach, Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych
Umiejętności			
M_U001	Student potrafi napisać program dla zadanego mikroprocesora/mikrokontrolera wykorzystując wiedzę o jego budowie wewnętrznej i assemblerze, korzystając co najmniej z jednego interfejsu komunikacyjnego.	IS2A_U16, IS2A_U15, IS2A_U17	Aktywność na zajęciach, Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych
M_U002	Student potrafi użyć odpowiednich narzędzi do skompilowania napisanego programu i załadowania go do mikrokontrolera/mikroprocesora pracującego w systemie wbudowanym.	IS2A_U16	Aktywność na zajęciach, Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych
M_U003	Student umie przedstawić wykonany projekt w formie prezentacji. Potrafi także wskazać obszary zastosowań tworzonych aplikacji i ekonomiczne aspekty zastosowanych rozwiązań.	IS2A_U09, IS2A_U10	Wykonanie projektu, Wykonanie ćwiczeń laboratoryjnych
Kompetencje społeczne			
M_K001	Student potrafi pracować w zespole projektowym. Potrafi samodzielnie zdobyć odpowiednią wiedzę i umiejętności niezbędne do realizacji powierzonej mu części zadania zespołowego.	IS2A_K02, IS2A_K03	Wykonanie projektu

## Matryca efektów kształcenia w odniesieniu do form zajęć

Kod EKM	Student, który zaliczył moduł zajęć wie/umie/potrafi	Forma zajęć										
		Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Inne	E-learning
Wiedza												
M_W001	Student zna budowę wewnętrzną wybranego mikroprocesora/mikrokontrolera oraz jego assemblera. Potrafi również wyjaśnić działanie i wskazać zastosowanie poszczególnych rozkazów.	+	-	+	-	-	-	-	-	-	-	-
M_W002	Student zna i potrafi scharakteryzować wybrane interfejsy komunikacyjne używane w systemach mikroprocesorowych.	+	-	+	-	-	-	-	-	-	-	-

M_W003	Student potrafi wyjaśnić proces „cross”-kompilacji i wskazać narzędzia niezbędne do pracy z mikroprocesorem/mikrokontrolerem pracującym w systemie wbudowanym.	+	-	+	-	-	-	-	-	-	-	-
Umiejętności												
M_U001	Student potrafi napisać program dla zadanego mikroprocesora/mikrokontrolera wykorzystując wiedzę o jego budowie wewnętrznej i assemblerze, korzystając co najmniej z jednego interfejsu komunikacyjnego.	+	-	+	+	-	-	-	-	-	-	-
M_U002	Student potrafi użyć odpowiednich narzędzi do skompilowania napisanego programu i załadowania go do mikrokontrolera/mikroprocesora pracującego w systemie wbudowanym.	+	-	+	+	-	-	-	-	-	-	-
M_U003	Student umie przedstawić wykonany projekt w formie prezentacji. Potrafi także wskazać obszary zastosowań tworzonych aplikacji i ekonomiczne aspekty zastosowanych rozwiązań.	-	-	+	+	-	-	-	-	-	-	-
Kompetencje społeczne												
M_K001	Student potrafi pracować w zespole projektowym. Potrafi samodzielnie zdobyć odpowiednią wiedzę i umiejętności niezbędne do realizacji powierzonej mu części zadania zespołowego.	-	-	-	+	-	-	-	-	-	-	-

## Treść modułu zajęć (program wykładów i pozostałych zajęć)

### Wykład

#### Tematy wykładów

1. Wprowadzenie: podstawowe pojęcia tj. mikrokontroler, a mikroprocesor, architektury, „softcore” itp.
2. Mikroprocesory typu „softcore” w układzie programowalnym FPGA
3. Mikroprocesor PlcoBlaze: architektura, rozkazy, programowanie w assemblerze
4. Szeregowe interfejsy przewodowe : RS232, SPI, I2C, PS/2
5. Mikrokontrolery na przykładzie LPC1786 z rdzeniem ARM CortexM3: architektura, budowa, programowanie, układy peryferyjne
6. Interfejs USB
7. Nowoczesne architektury mikroprocesorów: lista rozkazów CISC a RISC, tryby adresowania, potokowe i superskalarne przetwarzania rozkazów, zarządzanie pamięcią, mechanizmy wspomagania systemu operacyjnego itp.

## Ćwiczenia laboratoryjne

### Pracownia Komputerowo-Elektroniczna

1. Środowisko programistyczne i metodologia pracy z mikrokontrolerem PicoBlaze i układem FPGA

Efekty kształcenia:

- student potrafi wyjaśnić działanie poszczególnych rozkazów mikrokontrolera, zna jego budowę wewnętrzną i potrafi napisać program w asemblerze
- student potrafi dokonać „cross” kompilacji i użyć odpowiednich narzędzi do załadowania swojego programu do układu FPGA i tym samym do pamięci mikrokontrolera

2. Wykorzystanie układów i urządzeń zewnętrznych względem FPGA

Efekty kształcenia:

- student potrafi skomunikować się z zewnętrznym układem zewnętrznym (np. przetwornik A/C lub C/a) wykorzystując szeregowy protokół komunikacyjny
- student potrafi sterować i/lub odbierać dane z urządzenia zewnętrznego (np. klawiatura PS/2) wykorzystując jeden szeregowy protokół komunikacyjny

3. Środowisko programistyczne i metodologia pracy z typowym mikrokontrolerem (LPC1768)

Efekty kształcenia:

- student potrafi zapisać prosty program dla wybranego mikrokontrolera
- student potrafi dokonać „cross” kompilacji i użyć odpowiednich narzędzi do załadowania swojego programu do pamięci mikrokontrolera

4. Wykorzystanie wbudowanych układów peryferyjnych układu LPC1768

Efekty kształcenia:

- student potrafi skonfigurować wewnętrzne układy peryferyjne typu: licznik, „watchdog”, zegar czasu rzeczywistego, przetworniki A/C i C/A itp.
- student potrafi skonfigurować i wykorzystać wewnętrzne przerwania pochodzące od układów peryferyjnych

## Ćwiczenia projektowe

### Projekt zespołowy

Studenci w dwuosobowych zespołach realizują projekty opierające się o wybrany przez nich mikrokontroler. Każdy zespół otrzymuje do wykonania inny, przydzielony losowo lub uzgodniony z prowadzącym projekt. Punktem wyjścia dla studentów są dostarczone założenia i wskazówki projektowe. W ramach projektu należy w oparciu o zasoby sprzętowe dostępne na module ćwiczeniowym stworzyć działające urządzenie oraz dokumentację wykonania projektu.

Efekty kształcenia:

- student potrafi zinterpretować specyfikację urządzenia w celu zbudowania go za pomocą dostępnych narzędzi programowych
- student potrafi efektywnie wykorzystać odpowiednie narzędzia projektowe do stworzenia prostej aplikacji na module ćwiczeniowym
- student potrafi współpracować w grupie realizując swoją część zadania
- student potrafi zademonstrować funkcjonalność zbudowanego urządzenia zgodną z zadaną specyfikacją i przygotować dokumentację

## Sposób obliczania oceny końcowej

Ocena końcowa OK z modułu obliczana jest jako średnia ważona oceny z laboratoriów OL i projektów OP:

$$OK = 0.6 \times OP + 0.4 \times OL$$

## **Wymagania wstępne i dodatkowe**

Znajomość podstaw programowania

Znajomość podstaw elektroniki cyfrowej

Podstawowa wiedza o strukturze języków programowania

## **Zalecana literatura i pomoce naukowe**

1. Marcin Nowakowski, PicoBlaze mikroprocesor w FPGA, Wydawnictwo BTC, Legionowo 2010.
2. Jacek Bogusz, Lokale interfejsy szeregowy w systemach cyfrowych. BTC, Warszawa 2004.
3. A. Sloss, D. Symes, Ch. Wright, ARM System Developer's Guide: Designing and Optimizing System Software, Morgan Kaufmann, 2004.
4. Joseph Yiu, The Definitive Guide to the ARM Cortex - M3, Newnes, 2009.
5. Tomasz Francuz, Mikrokontrolery AVR i ARM : sterowanie wyświetlaczami LCD, Helion 2017.
6. K. Qian, D. den Haring, L. Cao, Embedded Software Development with C, Springer, 2009.
7. Kucjan Brynza, Mikrokontroler ARM9 w przykładach, BTC, Legionowo 2009.
8. Krzysztof Paprocki, Mikrokontrolery STM32 w praktyce, BTC, Legionowo 2011.
9. J. Hennessy, D. Patterson, Computer architecture: a quantitative approach, Elsevier 2012.

## **Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu**

K. Swientek, M. Banachowicz,

Design of Memory Subsystem for Wide Input Data Range in the SALT ASIC,

Proceedings of the 24th International Conference "Mixed Design of Integrated Circuits and Systems", June 22-24, 2017, Bydgoszcz, Poland.

S. Bugiel, R. Dasgupta, M. Firlej, T. Fiutowski, M. Idzik, M. Kopec, J. Moron, K. Swientek, Ultra-Low Power Fast Multi-Channel 10-Bit ADC ASIC for Readout of Particle Physics Detectors IEEE Trans. Nucl. Sci. vol. 63, 2016, pp. 2622-2631.

M. Idzik, K. Swientek, T. Fiutowski, S. Kulis, D. Przyborowski, A 10-Bit Multichannel Digitizer ASIC for Detectors in Particle Physics Experiments, IEEE Trans. Nucl. Sci. vol. 59, 2012, pp. 294-302.

S. Kulis, A. Matoga, M. Idzik, K. Swientek, T. Fiutowski, D. Przyborowski, A general purpose multichannel readout system for radiation detectors, JINST 7 T01004 2012, pp. 294-302.

## **Informacje dodatkowe**

### **Sposób i tryb wyrównania zaległości powstałych wskutek nieobecności studenta na zajęciach:**

Nieobecność na zajęciach wymaga od studenta samodzielnego opanowania przerabianego w tym czasie materiału.

O ile prowadzący wyrazi zgodę, student może wziąć udział w zajęciach laboratoryjnych jednej z równoległych grup celem wyrównania zaległości.

### **Zasady zaliczania zajęć**

W ramach laboratorium komputerowo-elektronicznego studenci pracując w dwuosobowych zespołach wykonują szereg ćwiczeń, które traktowane są jako niezbędne przygotowanie do wykonania projektu. Każde spotkanie w laboratorium rozpoczyna się od kartkówki.

Oceny z tej kartkówki uzupełnione o aktywność na zajęciach i/lub sprawność wykonywania ćwiczeń są podstawą do wystawienia oceny za część laboratoryjną.

Projekt oceniany jest w oparciu o procent zrealizowanych założeń projektowych i/lub ocenę niezawodności działania stworzonego urządzenia. Dodatkowo oceniany jest sposób zaprezentowania informacji technicznych zawartych w opracowanej dokumentacji.

Wszystkie projekty muszą zostać zaliczone.

Warunkiem uzyskania zaliczenia z pojedynczego projektu jest:

- pomyślna prezentacja działania układu,
- kod wykonanej aplikacji przekazany prowadzącemu,
- dokumentacja projektu.

**Nakład pracy studenta (bilans punktów ECTS)**

Forma aktywności studenta	Obciążenie studenta
Udział w wykładach	20 godz
Samodzielne studiowanie tematyki zajęć	15 godz
Udział w ćwiczeniach laboratoryjnych	22 godz
Przygotowanie do zajęć	22 godz
Udział w ćwiczeniach projektowych	15 godz
Wykonanie projektu	35 godz
Przygotowanie sprawozdania, pracy pisemnej, prezentacji, itp.	10 godz
Sumaryczne obciążenie pracą studenta	139 godz
Punkty ECTS za moduł	5 ECTS