



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Aplikacje sprzętowe mikrokontrolerów i mikrokomputerów [S1MiKC1>ASMiM]

Przedmiot

Kierunek studiów

Mikroelektronika i komunikacja cyfrowa

Rok/Semestr

3/5

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

Liczba godzin

Wykład

24

Laboratorium

30

Inne

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

3,00

Koordynatorzy

dr inż. Sławomir Michalak

slawomir.michalak@put.poznan.pl

dr inż. Krzysztof Arnold

krzysztof.arnold@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student posiada uporządkowaną i podbudowaną matematycznie wiedzę z podstaw teorii obwodów, niezbędną do zrozumienia działania obwodów elektrycznych. Opanował wiadomości o układach analogowych i cyfrowych w zakresie podstawowym. Ma wiedzę o podstawach technik pomiarowych. Potrafi pozyskiwać informacje z literatury w języku polskim i angielskim. Właściwie korzysta z aplikacji podstawowych elementów i układów elektronicznych. Potrafi przeprowadzić pomiary typowych parametrów sygnałów elektrycznych. Rozumie konieczność poszerzania własnej wiedzy i jest odpowiedzialny. Zachowuje się aktywnie na zajęciach, systematycznie rozwiązuje problemy w zespole.

Cel przedmiotu

Przedstawienie podstawowych zastosowań mikrokontrolerów i mikrokomputerów. Poznanie i zrozumienie mechanizmów współpracy mikrokontrolera i mikrokomputera z otoczeniem. Poznanie zasad działania mikrokontrolera w trybach przeglądania stanu modułów I/O i urządzeń zewnętrznych oraz obsługi przerwań. Opanowanie umiejętności programowania obsługi podstawowych układów, współpracujących z mikrokontrolerami. Opanowanie umiejętności tworzenia prostych systemów mikroprocesorowych i systemów zarządzanych przez mikrokomputer.

Przedmiotowe efekty uczenia się

Wiedza:

Ma uporządkowaną i podbudowaną praktycznie wiedzę w zakresie architektury zaawansowanych technicznie mikrokontrolerów i mikrokomputerów. Rozumie efektywne mechanizmy działania zasobów wbudowanych. Zna możliwości i zasady współpracy mikrokontrolerów i mikrokomputerów z układami zewnętrznymi. Ma podstawową wiedzę w zakresie podziału zadań sprzęt/oprogramowanie, programowania mikrokontrolerów w języku assemblera oraz ich aplikacji sprzętowych. Wie o postępach technologii, zmierzających do rozwoju sprzętu w kierunku integracji zasobów, zwiększenia szybkości działania i zmniejszania poboru mocy.

Umiejętności:

Potrafi korzystać z opisów firmowych współczesnych mikrokontrolerów. Umie posługiwać się środowiskiem programistycznym i pisać proste programy w języku assemblera. Potrafi analizować i zestawiać typowe systemy, budowane z wykorzystaniem mikrokontrolerów i mikrokomputerów.

Kompetencje społeczne:

Potrafi realizować projekty zespołowe. Ma poczucie odpowiedzialności za uruchamianie aplikacji mikrokontrolerów i mikrokomputerów.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Kolokwium pisemne i/lub ustne końcowe weryfikuje wiedzę i zrozumienie w zakresie treści wykładu. Zawiera pytania problemowe otwarte o zróżnicowanej punktacji. Ocena końcowa z kolokwium: poniżej 50% liczby punktów możliwych do uzyskania - 2,0; od 50% - 3,0; od 60% - 3,5; od 70% - 4,0; od 80% - 4,5; od 90% - 5,0.

Ocena końcowa z laboratorium jest średnią arytmetyczną ważoną ocen za realizację zadań podstawowych i dodatkowych (przygotowanie do kolejnych zadań, zachowanie, zaangażowanie, utrwalanie umiejętności) i ocen za sprawozdania indywidualne lub zespołowe, zamykające zadania. Wagę określa się na zajęciach wprowadzających. Ocena za sprawozdanie indywidualne lub zespołowe może być też weryfikowana w trybie odpowiedzi ustnej i/lub pisemnej. Zadania dodatkowe weryfikują umiejętności przy ubieganiu się o zaliczenie laboratorium lub podwyższenie oceny. Mogą obejmować kolokwium pisemne lub ustne. Skala dla ocen końcowych: jak dla zaliczenia wykładu. Zaliczenie poprawkowe laboratorium obejmuje część praktyczną i kolokwium pisemne lub ustne.

Treści programowe

Wykład: Architektura standardowych mikrokontrolerów i mikrokomputerów. Zasada działania CPU. Zasada działania i zastosowania wbudowanych modułów sprzętowych. Środowisko programistyczne, programatory i systemy docelowe. Programowanie ISP. Programowanie mikrokontrolerów w języku assemblera. Organizacja stosu i wywoływanie podprogramów. System przerwań i jego zastosowania. Zasada działania, inicjalizacja i tryby pracy portów równoległych. Aplikacje i obsługa programowa typowych układów wprowadzania danych do mikrokontrolera. Podstawowe aplikacje mikrokontrolerów i mikrokomputerów z obsługą urządzeń wyjściowych. Obciążalność portów. Aplikacje z podglądem stanu portów równoległych. Współpraca mikrokontrolerów i mikrokomputerów z terminalem znakowym. Komunikacja z komputerem PC.

Laboratorium: Wprowadzenie do środowiska programistycznego i podstawy programowania w języku assemblera. Instrukcje standardowych mikrokontrolerów i ich efektywne wykorzystanie. Rejestry robocze, rejestry funkcyjne i pamięci. Wewnętrzny transfer danych. Warunkowe i bezwarunkowe zmiany kolejności wykonywania rozkazów. Uruchamianie programów arytmetyczno-logicznych. Wprowadzanie opóźnień programowych. Aplikacje z wykorzystaniem stosu. Uruchamianie programu

głównego i podprogramów z wykorzystaniem narzędzi symulacyjnych. Programowanie mikrokontrolera w systemie docelowym. Praca z zestawem ewaluacyjnym. Uruchamianie portów równoległych mikroprocesora i mikrokomputera. Obsługa wejść dwustanowych w trybie przeglądania. Obsługa przycisków i klawiatur. Zastosowania trybu wyjściowego portów do sterowania urządzeń zewnętrznych. Wizualizacja stanu portów równoległych z wykorzystaniem diod LED i wyświetlaczy. Uruchamianie wbudowanych modułów ze wsparciem sprzętu pomiarowego. Uruchamianie komunikacji stacji podrzędnej z komputerem PC.

Tematyka zajęć

Architektura i właściwości mikrokontrolerów i mikrokomputerów. Środowisko programistyczne i praca mikrokontrolera w systemie docelowym. Język assemblera w zastosowaniach. Narzędzia wspierające uruchomienie sprzętu i oprogramowania. Aplikacje z przetwarzaniem danych. Korzystanie z trybu przeglądania stanu urządzeń i systemu przerwań. Tryby pracy i podstawowe zastosowania portów równoległych. Tryby pracy i podstawowe zastosowania portów szeregowych. Rozszerzanie sprzętowe systemu.

Metody dydaktyczne

Wykład z prezentacją multimedialną, wspomagany dyskusją problemową i przykładami na tablicy, zawierający aktywizujące studentów odniesienia do obserwowanych przez nich lub prognozowanych doświadczalnych wyników zadań laboratoryjnych.

Laboratorium: realizacja praktycznych zadań problemowych podanych przez prowadzącego i weryfikacja wyników z wykorzystaniem środowiska programistycznego i zestawów uruchomieniowych, dyskusja porównawcza końcowych rozwiązań, ewentualne wprowadzanie zadań wymagających współpracy dwóch lub kilku zespołów. Interaktywne różnicowanie zadań z uwzględnieniem postępów studentów.

Literatura

Podstawowa:

1. Andrzej Pawluczuk: Sztuka programowania mikrokontrolerów AVR. Podstawy. Wyd. BTC, Warszawa 2006
2. Andrzej Pawluczuk: Sztuka programowania mikrokontrolerów AVR. Przykłady. Wyd. BTC, Warszawa 2007
3. Rafał Baranowski: Mikrokontrolery AVR ATmega w praktyce. Wyd. BTC, Warszawa 2005
4. ATmega16A. 8-bit AVR Microcontroller with 16K Bytes In-System Programmable Flash. Datasheet, Atmel Corporation 2014

Uzupełniająca:

1. Paweł Hadam: Projektowanie systemów mikroprocesorowych. Wyd. BTC, Warszawa 2004
2. ATmega128A. 8-bit AVR Microcontroller Datasheet Complete. Atmel Corporation 2015
3. ATmega8A, mega AVR Data Sheet. 2020 Microchip Technology Inc.
4. Jacek Bogusz: Lokalne interfejsy szeregowe w systemach cyfrowych. Wyd. BTC, Warszawa 2004

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	84	3,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	54	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	30	1,00