

Procesory Sygnałowe (PS)

<http://ii.pw.edu.pl/kowalski/dsp>

Prowadzący:	mgr inż. Henryk A. Kowalski p.308 kowalski@ii.pw.edu.pl								
Numer wersji:	2								
Semestr wprowadzenia:	03L								
Opis:	Studia I stopnia – ISI (obieralny), etap B4								
Wymiar godzinowy zajęć (JD):	<table><thead><tr><th>W</th><th>C</th><th>L</th><th>P</th></tr></thead><tbody><tr><td>2</td><td>-</td><td>1</td><td>1</td></tr></tbody></table>	W	C	L	P	2	-	1	1
W	C	L	P						
2	-	1	1						
ECTS:	5								
Klasy programowa:	OT – Przedmioty Obieralne techniczne SYKI - Systemy Komputerowe (I) SYKIC - Systemy Komputerowe (I-C)								
Klasy tematyczne:	SYK - Systemy Komputerowe TUPS - Techniki i Układy Przetwarzania Sygnałów								
Wymagane przedmioty poprzedzające:	ARKO - Architektura Komputerów lub SYKO - Systemy komputerowe								
Zalecane przedmioty poprzedzające:	SM - Systemy Mikroprocesorowe lub TM- Technika Mikroprocesorowa lub TMIK – Podstawy Techniki Mikroprocesorowej								
Forma zaliczenia:	Egzamin								
Semestr wzorcowy (zalecany):	7								
Aktualizacja:	27.02.2014								

Słowa kluczowe: Procesor sygnałowy, architektura typu Harvard, RISC, arytmetyka stałoprzecinkowa, potokowanie, system wbudowany, system operacyjny czasu rzeczywistego, układy SOC, procesory wielordzeniowe, zintegrowane środowisko programowania.

Cel przedmiotu:

Praktyczne zapoznanie studentów z budową i zastosowaniem nowoczesnych procesorów sygnałowych.

Przegląd najważniejszych architektur procesorów sygnałowych oraz sposobów ich programowania.

Pozwala na zaznajomienie się z przeznaczonym dla tych procesorów narzędziami programowymi, językami assemblerowymi i wysokiego poziomu oraz systemami operacyjnymi czasu rzeczywistego.

Umożliwia praktyczne zweryfikowanie nabytej wiedzy podczas pracy w środowisku sprzętowo-programowym wiodących producentów.

Treść wykładu:

Wprowadzenie (2h):

pojęcia podstawowe, modyfikowane architektury typu Harvard, różnice w stosunku do procesorów ogólnego przeznaczenia, typowe obszary zastosowań.

Budowa procesorów sygnałowych (2h):

reprezentacja danych arytmetycznych, organizacja szyn, organizacja pamięci, podstawowe bloki funkcjonalne, układy peryferyjne, jednostka wspomagająca uruchamianie.

Języki assemblerowe (2h):

specjalizowane tryby adresowania, ortogonalność, specjalizowane instrukcje, mnożenie, przesłania równoległe, pętle sprzętowe, formaty instrukcji.

Budowa procesora sygnałowego wybranej rodziny (2h):

procesory rodziny TMS320C2000, organizacja rdzenia C28x, budowa rdzenia CLA, jednostka FP, układ generacji sygnału zegarowego, układ CSM, układ Watchdog, układ generacji sygnału RESET, wykonywanie Bootowania procesora, organizacja układu przerwań, układ PIE, moduł wejścia-wyjścia GPIO

Budowa modułów peryferyjnych procesora sygnałowego wybranej rodziny (4h):

organizacja modułów: CPU Timer, ePWM – podmoduł TB, CC, AQ, DB, PC, TZ, ET, DC), HRPWM, ADC, TEMP i COMP, eCAP oraz modułu SCI/UART

Programowanie procesora sygnałowego w języku wysokiego poziomu (2h):

specjalizowane struktury językowe, reprezentacja danych arytmetycznych, powiązania pomiędzy językiem C i assemblerem, rozkazy, procedury, skoki, przerwania, sekcje, pakiet programowy controlSUITE, pakiety typu „Firmware Development Package”

Zintegrowane środowisko programowania Code Composer Studio (2h)

Definiowanie projektów w CCS, importowanie projektów, perspektywa C/C++, okno C/C++ Project, budowanie projektu, linker, okno Console, tworzenie pliku definicji konfiguracji sprzętowej, perspektywa Debug, okno Disassembly, pułapki, okno Watch, okno Memory,

Narzędzia sprzętowe i programowe, debugowanie (2h):

narzędzia programowe generowania kodu: kompilatora języka C, assembler, symulator,.

sprzętowe zestawy uruchomieniowe, emulatory sprzętowe, wspomaganie debugowania w środowisku CCS.

Procesory wielordzeniowe i SOC (2h):

Procesory dwurdzeniowe rodziny TMS320F28M25x, rdzeń TMS320C28x, rdzeń ARM Cortex-M3, zasoby wspólne procesora, podsystem Master, podsystem Control, podsystem Analog, blok Clocking, bloki NMI, blok Reset, blok Voltage Monitor, moduły peryferyjne, programowanie, pakiet programowy controlSUITE, moduł uruchomieniowy H52C1 Concerto controlCARD

Potokowanie (2h)

Wprowadzenie, głębokość potoku, hazard, mechanizm „interlocking”, wpływ skoku, skoki opóźnione, obsługa przerwania, szybkie przerwanie,

Systemy operacyjne czasu rzeczywistego (2h):

pojęcia podstawowe, monitory, strumienie, wektory, tablice, wyjątki, zarządzanie pamięcią, biblioteki.

Przegląd procesorów sygnałowych (2h):

procesory stałoprzecinkowe, zmiennoprzecinkowe, ewolucja procesorów sygnałowych, architektura z podwójnym układem MAC, architektura superskalarna, techniki SIMD, architektura z bardzo długim słowem instrukcyjnym (VLIW).

Podstawowe zagadnienia omawiane są na przykładzie procesorów rodziny TMS320C2000, TMS320C6000 (Texas Instruments), DSP56xxx (Freescale) oraz systemu operacyjnego czasu rzeczywistego DSP/BIOS (Texas Instruments). Prezentowane są również wybrane zagadnienia budowy procesorów sygnałowych innych rodzin oraz firm.

Zakres laboratorium

Zajęcia laboratoryjne obejmują zapoznanie się praktyczne ze środowiskiem programowania CCS oraz z pracą poszczególnych modułów peryferyjnych procesora sygnałowego z zastosowaniem sprzętowego systemu uruchomieniowego czasu rzeczywistego.

- Lab 1: C1 Pierwszy projekt w CCS
C2 Projekt Watchdog
C3 Projekt LED blinking
- Lab 2: C4 Praca z modułem ePWM w trybie Upcount
C5 Praca z modułem HRPWM
- Lab 3: C6 Praca z modułem ADC
C7 Praca z czujnikiem temperatury modułu ADC
C8 Praca z modułem eCAP w trybie Capture
- Lab 4: C9 Praca z modułem eCAP w trybie APWM
C10 Praca z modułem SCI

Każde zajęcia laboratoryjne trwają 175min. bez przerwy. Ćwiczenia laboratoryjne są wykonywane w zespołach dwuosobowych. Ocena rozwiązań ćwiczeń laboratoryjnych odbywa się w trakcie zajęć na podstawie pisemnego sprawozdania i rozmowy z zespołem studenckim realizującym ćwiczenia.

Zakres projektu

Projekt obejmuje napisanie i uruchomienie aplikacji czasu rzeczywistego w języku C według założeń podanych w temacie projektu. Projekt jest realizowany w zespołach kilkuosobowych. Projekt jest uruchamiany z wykorzystaniem środowiska programowania CCS oraz sprzętowego systemu uruchomieniowego czasu rzeczywistego w laboratorium sprzętowym. Projekt musi być zrealizowany do końca semestru. Ocena projektu odbywa się na podstawie działającego oprogramowania oraz przedstawionego pisemnego sprawozdania. Pełne środowisko CCS jest dostępne również dla studentów do pracy w domu. Student może uczestniczyć w prowadzonych co tydzień konsultacjach (w wymiarze 1 godz.). Projekt umożliwia utrwalenie wiedzy nabytej podczas wykładu i zajęć laboratoryjnych oraz nabycie umiejętności pracy zespołowej

Literatura:

- [1] Kowalski HA Procesory DSP dla praktyków, BTC, Legionowo 2011
- [2] Lapsley P, Bier J, Shoham A, Lee EA, DSP processors fundamentals, Architectures and features, Berkeley Design Technology, Inc, 1996. (John Wiley & Sons, 1997)
- [3] Marvin C, Ewers G, A simple approach to digital signal processing, Texas Instruments, 1993. (John Wiley & Sons, 1996) (Marven C, Ewers G, Zarys cyfrowego przetwarzania sygnałów WKŁ, 1999.)
- [4] EDN DSP Directory www.ednmag.com
- [5] Steven W. Smith, The Scientist and Engineer's Guide to Digital Signal Processing, California Technical Publishing San Diego, California, Second Edition, 1999, www.DSPguide.com
- [5] Strona internetowa firmy Texas Instruments <http://www.ti.com/>