



Технології програмування спеціалізованих процесорів

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Другий (магістерський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>123 Комп'ютерна інженерія</i>
Освітня програма	<i>Комп'ютерна інженерія</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>очна(денна)</i>
Рік підготовки, семестр	<i>1 курс, весняний</i>
Обсяг дисципліни	<i>5 кредитів , 150 год. Лекцій 36 годин, лабораторних 18 годин</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен</i>
Розклад занять	<i>Згідно розкладу на весняний семестр поточного навчального року за адресою http://rozklad.kpi.ua</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: професор кафедри ОТ, д.т.н., доцент Клименко Ірина Анатоліївна ikliryna@gmail.com.</i>
Розміщення курсу	<i>Лекційний матеріал: https://classroom.google.com/c/NDYxMjYwMTM2Nzc0?cjc=dqopnyr</i>

Програма навчальної дисципліни

Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Спеціалізовані процесори використовуються в технологіях Інтернету Речей (IoT), убудованих системах, розумних системах. Відомі апаратні системи для автомобільної промисловості, промислової автоматизації, високопродуктивних обчислень, штучного інтелекту машинного навчання. Відомі рішення для застосування в медичних додатках, фитнес-трекерах, в додатках, де виникає необхідність оброблення великих об'ємів даних в високою швидкістю, мультимедійних додатках, іграх з високою якістю і мультимедійним контентом, тощо.

– **Мета дисципліни** – вивчення теоретичних та практичних основ технології створення спеціалізованих процесорів на базі систем на кристалі (SoC) Розроблення власних модулів та розширень для спеціалізованих процесорів на базі технології SoC для вирішення користувацьких задач.

Технологія створення спеціалізованих процесорів SoC дозволяють інтегрувати відразу декілька необхідних функцій в одну систему на одному чипі. На сьогодні будь який сучасний процесор, універсальний або спеціалізований, побудований за технологією SoC, яка інтегрує арифметичні розширювачі, графічні прискорювачі, убудовані кеші, інтерфейси, пам'ять. Це нова ера розвитку процесорних ядер, що має на меті інтенсивне підвищення продуктивності обчислень для забезпечення сучасних користувацьких потреб. В курсі розглядаються, як основні аспекти

технологій розроблення SoC з використанням сучасних програмованих технологій на FPGA, так і питання використання відомих SoC процесорів в різноманітних технологічних та технічних системах, IoT, та побуті.

Вивчення дисципліни спрямовано на оволодіння студентами наступними фаховими компетенціями:

- здатність ідентифікувати, класифікувати та формулювати вимоги до технічного та програмно-апаратного забезпечення комп'ютерів;
- знання фундаментальних концепцій, парадигм і основних принципів функціонування обчислювальних засобів інженерії програмного забезпечення;
- здатність до використання теоретичних, логічних та арифметичних основ побудови сучасних комп'ютерів і вміння їх застосовувати при рішенні професійних завдань;
- здатність розробляти окремі компоненти комп'ютерних систем, в тому числі, з використанням сучасних систем автоматизації проектування;
- здатність розробки та використання елементів архітектурозалежного рівня операційних систем на основі знання загальних принципів організації та функціонування операційних систем;
- здатність брати участь у командній роботі по проектуванню комп'ютерів та окремих елементів комп'ютерних систем;
- здатність формулювати та забезпечувати вимоги щодо надійності комп'ютерних систем у відповідності з вимогами замовника, технічним завданням та стандартами.

У відповідності до вищезазначеного, студенти отримають наступні результати навчання:

- сформулюють систему знань і умінь в області розроблення вузлів процесора та модулів IP ядер;
- оволодіють знаннями фундаментальних концепцій, парадигм і основних принципів розроблення IP ядер та процесорних ядер;
- отримають поняття та кваліфікаційні основи професії програміст апаратного забезпечення на мові високого рівня опису апаратури Verilog VHDL;
- навчатимуться контролювати дотримуватись стандартів розробки вбудованих систем та ПЗ для них та використовувати уніфіковані принципи та методи виявлення та попередження помилок;
- ознайомляться з основними принципами верифікації та тестування модулів вбудованих систем на рівні Hardware, зокрема автоматизованого тестування;
- ознайомляться з основними принципами побудови, налаштування та тестування оточення вбудованих систем на інфраструктурному та мережному рівнях IoT;
- ознайомляться з принципами розвертання та функціонування операційної системи Linux на процесорах для вбудованих систем;
- налаштувати операційну систему Linux на сучасних спеціалізованих процесорах.
- розробляти власні модулі для процесорного ядра.
- розробляти цифрові пристрої (спеціалізовані обчислювачі, арифметичні розширювачі) на FPGA та інтегрувати їх в систему під керуванням ОС Linux.
- працювати з інтерфейсами та датчиками.
- навчатимуться складати технічну документацію;
- отримають досвід командної роботи;
- отримають досвід відстоювання власних рішень у професійній дискусії та представлення результатів власних розробок;
- отримають досвід використання компетентностей комп'ютерної інженерії на практиці;
- використання Git та GitHub для контролю версій та роботи в команді розробників;

Кредитний модуль забезпечує наступні компетентності і програмні результати освітньо-практичної програми першого (бакалаврського) рівня вищої освіти (ОПП): ФК1, ФК2, ФК3, ФК6, ФК12, ФК20, ФК23: ПРН1, ПРН2, ПРН5, ПРН7, ПРН14, ПРН15, ПРН16, ПРН17.

1. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

При вивченні дисципліни «Технології тестування (QA) вбудованих систем» доцільно використовувати знання, отримані при вивченні попередніх дисциплін: «ОС Linux», «Комп'ютерна логіка», «Теорія електричних кіл та сигналів», «Комп'ютерна електроніка» (базові знання), «Програмування», «Алгоритми і структури даних», «Технології програмування на C для вбудованих систем», «Програмування Python» (базові знання), «Системне програмування» (базові знання), «Операційні системи» (базові знання), «Комп'ютерні мережі», «Іноземна мова».

Дисципліна є базовою для курсів: «Інтелектуальні Real Time системи», «Інтернет речей IoT», «Devops», «Керування проектами», «Системне програмування. Розробка системних програм», «Комп'ютерне моделювання», «Організація обчислювальних процесів», «Системне програмне забезпечення», «Комп'ютерні мережі», «Комп'ютерні системи», «Технології проектування комп'ютерних систем», «Дослідження і проектування паралельних систем», «Технології програмування паралельних комп'ютерних систем», «Вбудовані системи», «Сучасні технології програмування», «Програмування системи реального часу», «IoT».

2. Структура кредитного модуля

Вступ

Розділ 1. Введення в архітектуру сучасних процесорів. Особливості архітектурної та функціональної організації сучасних RISC та CISC процесорів.

Тема 1.1. Сучасна класифікація процесорів за функціональним призначенням. Універсальні та спеціалізовані процесори. Огляд сучасної елементної бази. Системи на кристалі (SoC). Багатоядерні процесори.

Тема 1.2. Архітектура Фон-Неймана в сучасних процесорних ядрах. Архітектура AVR, як флагман сучасної елементної бази для процесорних ядер, орієнтованих на встановлення універсальних операційних систем.

Розділ 2. Технології програмування сучасних SoC

Тема 2.2. Тенденції розвитку технологій розроблення процесорів в умовах 4-ї Індустріальної революції.

Розділ 2. Сучасні технології проектування та розроблення спеціалізованих процесорів (SoC). Розробка засобів на програмовних логічних інтегральних схемах (ПЛІС).

Тема 2.1. Життєвий цикл процесу проектування та розроблення апаратних засобів. Уніфікований процес розробки мікросхем.

Тема 2.3. Огляд сучасних САПР

Розділ 3. Особливості програмування спеціалізованих процесорів з використанням програмовних інтегральних мікросхем (FPGA)

Тема 3.1. Архітектура SoC

Тема 3.2. Основні етапи та особливості розроблення

Тема 3.3. Стандартний потік проектування

Тема 3.4. Архітектурно орієнтоване проектування

Розділ 4. Практичні аспекти розроблення спеціалізованого процесора

Тема 4.1 Огляд плати для розробки DE1-SoC

Тема 4.2 Використання системи автоматизації проектування Intel Quartus Prime

Тема 4.3 Розробка модулів для взаємодії з зовнішніми пристроями

Тема 4.4 Розробка модулів для виконання арифметичних операцій

Тема 4.5 Розробка модулів запам'ятовуючих пристроїв

Тема 4.6 Розробка модулів системи керування

Розділ 5. Розроблення вбудованого пристрою на базі спеціалізованого ARM процесору

Тема 5.1. Особливості збирання, налагодження та використання операційних систем типу Linux на процесорах AVR для реалізації вбудованих систем на базі спеціалізованих процесорів.

Тема 5.2. Збирання виконуваних програм завантажувальника (u-boot), ядра (Kernel), файлової системи (RootFs) для процесора AVRv7.

Тема 5.3. Етапи розвертання операційної системи. Процеси ініціалізації архітектури в просторі ядра операційної системи на прикладі архітектури процесорів ARM CORTEX.

Тема 5.4. Прошивка процесора. Робота з HareWare.

3. Література

1. Клименко І.А., Ткаченко В.В. «Технології програмування спеціалізованих процесорів. Частина 1. Технології програмування SoC: Навчальний посібник : методичні вказівки для виконання лабораторних робіт для здобувачів другого освітнього рівня (магістра) за освітньо-професійною програмою «Комп'ютерна інженерія» за спеціальністю 123 «Комп'ютерна інженерія» / І.А. Клименко, В.В. Ткаченко. – Київ : КПІ ім. Ігоря Сікорського, 2022. – XX с. (В роботі)
2. Архітектура комп'ютерів. Процесори: Теорія та лабораторний практикум. Навчальний посібник для здобувачів ступеня бакалавра за спеціальністю 123 «Комп'ютерна інженерія» / уклад.: І.А. Клименко, В.А. Таранюк, А.В. Каплунов, В.В. Ткаченко – Київ : КПІ ім. Ігоря Сікорського, 2022. – 92 с., Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 1 від 02.09.2022 р.).
3. Жабін В.І., Жуков І.А., Ткаченко В.В., Клименко І.А.. Прикладна теорія цифрових автоматів. : Навчальний посібник.- К.: Видавництво «НАУ Друк», 2009.- 492 с.

4.2. Додаткова література

4. Жабін В.І., Клименко І.А., Стіренко С.Г. Арифметичні та управляючі пристрої цифрових ЕОМ: Навчальний посібник. – К.:БЕК +, 2008. –
5. Клименко І.А. Класифікація та архітектурні особливості програмованих мультипроцесорних систем-на-кристалі // Проблеми інформатизації та управління: Зб.наук.пр.– К.: Вид-во нац. авіац. ун-ту «НАУ-друк», 2012.– Вип.1(36). – С 90 – 103.
6. Клименко І.А. Тенденції застосування сучасної елементної бази для побудови високопродуктивних обчислювальних систем // Проблеми інформатизації та управління: Зб.наук.пр. – К.: Вид-во нац. авіац. ун-ту «НАУ-друк», 2010.– Вип.1(29). – С 90 – 103.
7. DE1-SoC User Manual (rev.F/rev.G Board) Terasic Technology Inc. [Електронний ресурс]. Режим доступу: <https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&No=836&PartNo=4>
8. Complex Programmable Logic Device (CPLD) [Електронний ресурс]. Режим доступу: <https://www.xilinx.com/products/silicon-devices/cpld/cpld.html>
9. Field Programmable Gate Array (FPGA) [Електронний ресурс]. Режим доступу: <https://www.xilinx.com/products/silicon-devices/fpga/what-is-an-fpga.html>
10. INTEL FPGAS [Електронний ресурс]. Режим доступу: <https://www.intel.com/content/www/us/en/products/programmable/fpga.html>
11. INTEL CYCLONE FPGAS AND CYCLONE V SOC DEVICES [Електронний ресурс]. Режим доступу: <https://www.intel.com/content/www/us/en/products/programmable/cyclone-series.html>
12. DE1-SoC User Manual (rev.F/rev.G Board) Terasic Technology Inc. [Електронний ресурс]. Режим доступу: <https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&No=836&PartNo=4>
13. Altera Monitor Program Tutorial for ARM [Електронний ресурс]. Режим доступу: ftp://ftp.intel.com/pub/fpgaup/pub/Intel_Material/14.1/Tutorials/Altera_Monitor_Program_ARM.pdf
14. Cyclone V Hard Processor System Technical Reference Manual [Електронний ресурс]. Режим доступу:

https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/cyclone-v/cv_54001.pdf

4.3. Інформаційні ресурси

15. Курс відеолекцій – Курс «Технології програмування спеціалізованих процесорів» на Google Classroom <https://classroom.google.com/c/NDYxMjYwMTM2Nzc0?cjc=dqopnyr>

4. Лабораторні роботи

Мета лабораторних робіт – придбання практичних навичок роботи з сучасним обладнанням та ознайомлення з відомими технологіями створення SoC що базуються на використанні продуктів компанії Altera та розроблення продукту з власною прошивкою для вирішення заданої користувацької задачі.

Під час вивчення курсу використовується налагоджувальна плата DE1-SoC на базі ALTERA System-on Chip (SoC) FPGA, яка містить вбудований двоядерний процесор Cortex-A9 на базі ARM і масив програмованої логіки. Плата DE1-SoC базується на процесорному ядрі Cortex-A9, оптимізованому для встановлення ОС Linux, поєднує різноманітні інтерфейси, периферію, високошвидкісну пам'ять DDR3, аудіо- та відео-інтерфейси, Ethernet, можливість програмування на інтегрованій FPGA.

Під час вивчення курсу також можуть бути використані платформи BeagleBone Black, який базується на SoC Texas Instruments AM335x (Sitara) та Raspberry Pi.

Теми лабораторних робіт

Лабораторна робота 1. Вступне заняття. Тестування.

Лабораторна робота 2. Розвертання оточення та інсталяція програмного забезпечення для роботи з платою для розробки DE1-SoC

Лабораторна робота 3. Розробка модулів для взаємодії з зовнішніми пристроями

Лабораторна робота 4. Розробка модулів для виконання арифметичних операцій

Лабораторна робота 5. Розробка модулів запам'ятовуючих пристроїв

Лабораторна робота 6. Розробка модулів для відліку системного часу

Лабораторна робота 7. Розробка модулів системи керування

Лабораторна робота 8. Збирання спеціалізованого процесорного ядра. Підключення модулів.

5. Самостійна робота студента

Види самостійної роботи:

- підготовка до аудиторних занять, (0,5 годин x 18 лекцій = 9 годин);
- підготовка та оброблення проведення розрахунків за первинними даними, отриманими на практичних заняттях, виконання практичних робіт розв'язок задач (2 години x 8 лабораторних робіт = 16 годин);
- виконання модульної контрольної роботи (2 МКР x 4 години = 8 годин).

Політика та контроль

6. Політика навчальної дисципліни (освітнього компонента)

Для досягнення мети навчальної дисципліни слід зосередитись в лекційному матеріалі на особливостях побудови функціонального рівня комп'ютера, процесора та інших його компонентів. Особливу увагу необхідно приділити особливостям проектування комп'ютерів із застосуванням сучасної елементної бази.

Для виконання лабораторних робіт та модульних контрольних робіт встановлюються дедлайни. Виконання лабораторних робіт в рамках встановлених термінів оцінюються балами за протокол.

Модульна контрольна робота виконується самостійно за індивідуальним завданням, на виконання МКР встановлюються дедлайни, МКР не приймається поза встановлені терміни.

Окремі теми лекційних занять супроводжуються короткими експрес-тестами (на 15 хвилин), які включають матеріал вивченої теми та питання, які задані на самостійне вивчення. Бали отримані за тест входять в семестрову рейтингову оцінку. Поточні тести не перескладаються.

В дистанційному режимі навчання допускається семестровий контроль автоматом за умови отримання від 60 до 100 балів за поточні роботи за семестр, за умови виконання всіх лабораторних робіт та МКР.

Виконання лабораторних робіт та МКР є обов'язковими для допуску до семестрового контролю автоматом. Для допуску до семестрового контролю автоматом, оцінка, яку студент може отримати за виконання кожної лабораторної роботи – $(3+(7 \times 3)+(4 \times 3)+6)$ балів (всього 42 бали) та за кожну модульну контрольну роботу – по 9 балів (всього 18 балів), див. таблицю оцінювання семестрових робіт, розділ 8. Таким чином мінімальна оцінка, яку може отримати студент, щоб зарахувати навчальний курс автоматом = 60 балів, максимальна – 100 балів за виконання всіх поточних робіт за семестр.

Модульна контрольна робота не переписується за умови негативної оцінки, негативна оцінка за МКР (менше ніж 9 балів (<60%)) не дає права отримати іспит автоматом, але вважається такою, що була написана і може бути зарахована для допуску до семестрового іспиту.

Якщо студент набрав менше ніж 60 балів за виконання поточних робіт, при цьому він захистив всі лабораторні роботи та МКР на мінімальні бали, студент пише екзаменаційну роботу, яка оцінюється в 40 балів. Бали за екзаменаційну роботу додаються до балів за лабораторні роботи та МКР і складають семестрову рейтингову оцінку.

7. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Розподіл навчального часу за видами занять і завдань з дисципліни згідно з робочим навчальним планом. На кредитний модуль виділено 150 годин та 5 кредитів.

Семестровий рейтинг студента з кредитного модуля розраховується, виходячи із 100-бальної шкали. Рейтинг складається з балів, що студент отримує за виконання 8 лабораторних робіт R_L , двох модульних контрольних робіт $R_{МКР}$ та екзамену R_E .

Максимальна кількість балів за лабораторні роботи складає 60 балів, тобто $R_L = 60$.

Узагальнені критерії оцінювання лабораторних робіт наступні:

- своєчасність підготовки протоколу до лабораторного заняття, повнота виконання теоретичного або практичного завдання в протоколі, протокол вчасно викладений на GitLab;
- коректність функціонування розроблених моделей на програмному або апаратному забезпеченні, демонстрація власного репозиторію на GitLab з матеріалами лабораторної роботи та, наявність комітів;
- опитування за тематикою лабораторної роботи для зарахування практичної частини роботи, захист одержаних в роботі результатів, відповіді на додаткові теоретичні запитання викладача, повнота оформлення звіту/протоколу по роботі на GitLab.

Детальний підхід до оцінювання кожної лабораторної роботи наведений в таблиці 1.

Максимальна кількість балів за МКР $R_{МКР} = 2 \times 15 = 30$ балів.

МКР1 проводиться у вигляді автоматизованого тестування. Тест складається із 60 питань

$R_{МКР_2} = 0,25 \times 60 = 15$ балів

Критерії оцінювання МКР2 за чотирма рівнями:

- правильна та змістовна відповідь з поясненнями в термінах предметної області : 13 – 15 балів;

- правильна відповідь, неповні пояснення: 11 – 12 балів;
- відповідь містить помилки: 9 – 10 балів;
- відповідь містить суттєві помилки, немає пояснень: 4-8 балів;
- немає відповіді: 0 балів.

Оцінка за МКР2 знижується за:

- некоректне оформлення;
- відсутність коментарів в змістовних термінах;

відсутність пояснень під час розрахунків.

Максимальна кількість балів за екзамен дорівнює $R_E = 40$ балів.

Екзаменаційний білет містить 4 завдання (одне теоретичне і три практичних) за тематикою лекцій та лабораторних робіт, що виконувались в семестрі. Кожне запитання оцінюється від 0 до 10 балів.

Критерії оцінювання кожного запитання за чотирма рівнями:

- правильна та змістовна відповідь: 9 – 10 балів;
- правильна відповідь, неповні пояснення: 6 – 8 балів;
- відповідь містить помилки: 3 – 5 балів;
- немає відповіді або відповідь невірна: 0 балів.

Календарна атестація студентів (на 8 та 14 тижнях семестрів) з дисципліни проводиться за значенням поточного рейтингу студента на час атестації. Якщо значення цього рейтингу не менше 50 % від максимально можливого на час атестації, студент вважається атестованим. В іншому випадку в атестаційній відомості виставляється «неатестовано».

Таблиця 1. Деталізація оцінювання кожної лабораторної роботи

Назва заняття	Форма контролю	Кількість балів	Допуск до іспиту автоматом	Всього балів
Лабораторна робота 1.	Вступний тест (Linux introduction)	4	3	4
Лабораторна робота 2	Виконання завдання (Networking)	4	5	8
	Опитування по QA	3		
	Протокол на GitLab	1		
Лабораторна робота 3	Виконання завдання (Networking)	4	5	8
	Опитування по QA	3		
	Протокол на GitLab	1		
Лабораторна робота 4	Виконання завдання (Networking)	4	5	8
	Опитування по QA	3		
	Протокол на GitLab	1		
Лабораторна робота 5	Виконання завдання (Networking)	4	5	8
	Опитування по QA	3		
	Протокол на GitLab	1		
Лабораторна робота 6	Виконання завдання	6	5	8
	Протокол на GitLab	2		
Лабораторна робота 7	Виконання завдання	6	5	8
	Протокол на GitLab	2		
Лабораторна робота 8	Виконання завдання (Python)	3	5	8

	introduction)			
	Розроблення автотестів	4		
	Протокол на GitLab	1		
Експрес-тести на лекціях	2 x 5	10	4	10
МКР	МКР1 (Тест)	15	9	15
	МКР2	15	9	15
Всього балів		100	60	100

Кількість балів, що отримує студент за семестр визначається «автоматом» за формулою

$$RC = R_{л} + R_{МКР} + R_{\text{Експрес тести}}$$

Максимальна кількість балів за семестр не перевищує $RC = 100$.

З урахуванням одержаної суми балів кінцева оцінка визначається за таблицею 2:

Студент не може отримати оцінку за семестр «автоматом», якщо кількість семестрових балів менше ніж 60 балів. В цьому випадку, студент складає іспит з дисципліни. Необхідною умовою допуску до екзамену студента є виконання і захист всіх лабораторних робіт з сумою балів не менше ніж 30 балів.

Кількість балів, що отримує студент за семестр визначається формулою

$$RC = R_{л} + R_{МКР} + R_{Е}$$

Максимальна кількість балів за семестр не перевищує $RC = 100$.

З урахуванням одержаної суми балів кінцева оцінка визначається наступною таблицею:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

Робочу програму навчальної дисципліни (силабус):

Складено, д.т.н, доцент, професор кафедри ОТ Клименко Ірина Анатоліївна,

Ухвалено кафедрою обчислювальної техніки (протокол № 10 від 25.05.2022 р.)

Погоджено методичною комісією ФІОТ (протокол №10 від 09.06.2022 р.)