

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗАТВЕРДЖУЮ:

Декан факультету комп'ютерних
інформаційних технологій


Ігор ЯКИМЕНКО
“ 29 ” 2025 р.

ЗАТВЕРДЖУЮ:

Проректор з науково-педагогічної
роботи


Віктор ОСТРОВЕРХОВ
“ 29 ” 2025 р.

РОБОЧА ПРОГРАМА

з дисципліни “Системи реального часу”

ступінь вищої освіти – бакалавр

галузь знань – 12 “Інформаційні технології”

спеціальність – 122 “Комп'ютерні науки”

освітньо-професійна програма – “Штучний інтелект”

Кафедра інформаційно-обчислювальних систем і управління

Форма навчання	Курс	Семестр	Лекції (год.)	Лабораторні заняття (год.)	ІРС (год.)	Тренінг (год.)	Самост. робота студ. (год.)	Разом (год.)	Залік (сем.)
Денна	4	7	30	14	3	6	97	150	7

19.08.2025
[Signature]

Тернопіль – ЗУНУ
2025

Робочу програму склав доцент кафедри ІОСУ, к.т.н. Ігор МАЙКІВ

Робоча програма затверджена на засіданні кафедри інформаційно-обчислювальних систем і управління, протокол №1 від 26 серпня 2025 р.

В.о. завідувача кафедри, к.т.н.



Надія ВАСИЛЬКІВ

Гарант освітньо-професійної
програми “Штучний інтелект”
к.т.н., доцент



Василь КОВАЛЬ

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ “СИСТЕМИ РЕАЛЬНОГО ЧАСУ”

1. Опис дисципліни “Системи реального часу”

Дисципліна “Системи реального часу”	Галузь знань, спеціальність, СВО	Характеристика навчальної дисципліни
Кількість кредитів – 5	Галузь знань – 12 “Інформаційні технології”	Статус дисципліни: вибіркова дисципліна Мова навчання: Українська
Кількість залікових модулів – 3	Спеціальність – 122 “Комп’ютерні науки”	Рік підготовки: 4 Семестр: 7
Кількість змістових модулів – 2	Освітньо-професійна програма – “Штучний інтелект”	Лекції: 30 год. Лабораторні заняття: 14 год.
Загальна кількість годин – 150	Ступінь вищої освіти – бакалавр	Самостійна робота: 97 год. Тренінг: 6 год. Індивідуальна робота: 3 год.
Тижневих годин – 10, з них аудиторних – 3 год.		Вид підсумкового контролю: Залік

2. Мета і завдання дисципліни

“Системи реального часу”

2.1. Мета вивчення дисципліни

Метою дисципліни “Системи реального часу” є надання студентам систематичних знань і практичних навичок необхідних для проектування вбудованих систем (ВС) різноманітного призначення з урахуванням технічних та економічних обмежень. Навчальна програма та тематичний план курсу зосереджені на вивченні:

- архітектури процесорів, зокрема ARM архітектури;
- розроблення програмного забезпечення (ПЗ) ВС на мовах асемблера та C;
- реалізація алгоритмів керування та контролю у вбудованих системах;
- базових навиків створення програмного забезпечення ВС з використанням операційних систем реального часу (ОСРЧ), зокрема FreeRTOS;
- системних підходів до створення рішень для різних областей застосування.

2.2. Завдання вивчення дисципліни

Завдання дисципліни “Системи реального часу” – сформулювати у студентів знання та практичні навички проектування та програмування систем на базі мікроконтролерів (МК) з урахуванням обмежень продуктивності, енергоспоживання та надійності.

Курс ознайомлює з базовими принципами побудови вбудованих систем, на базі ARM-мікроконтролерів, організацією вводу-виводу, системою переривань, таймерами та послідовними інтерфейсами (UART, SPI, I2C).

Особлива увага приділяється ознайомленню із ОСРЧ, зокрема FreeRTOS та їх практичному застосуванню: створенню та плануванню задач, управлінню пам'яттю, використанню черг, семафорів, м'ютексів і програмних таймерів.

Формує навик системного аналізу.

2.3. В результаті вивчення дисципліни студенти повинні:

- знати принципи побудови ВС, архітектуру ARM-процесорів та особливості роботи з периферійними пристроями;
- вміти проектувати та реалізовувати програми для МК STM32 мовою C з використанням базових периферійних модулів (GPIO, таймери, UART, SPI, I2C) для взаємодії із сенсорами та виконавчими механізмами;
- вміти розробляти та налагоджувати прототипи прикладних ВС на основі FreeRTOS, інтегруючи сенсори та комунікаційні модулі.

3. Програма навчальної дисципліни “Системи реального часу”

Змістовий модуль 1. Основи програмування вбудованих систем

Тема 1. Поняття вбудованих систем. Приклади застосування

Визначення вбудованих систем, їх основні характеристики. Відмінність від універсальних обчислювальних пристроїв. Приклади застосування у різних сферах (промисловості, автомобілебудуванні, побутовій техніці, IoT).

Тема 2. Архітектура ARM процесорів. Розроблення ПЗ на мові C для ARM

Архітектура ARM Cortex-M (реєстри, режими роботи, енергоспоживання). Цикл розроблення програмного забезпечення мовою C для ARM МК. Використання середовища STM32CubeIDE та стандартів CMSIS. Бібліотеки HAL як інструмент для роботи з периферією МК.

Тема 3. Порти вводу-виводу. Система переривань ARM процесорів

Принципи роботи портів вводу-виводу (GPIO) у МК. Режими налаштування GPIO (вхід, вихід, альтернативні функції, аналоговий режим) приклади використання. Система переривань, структура таблиці векторів переривань, основи її роботи. Типи переривань (апаратні, зовнішні, внутрішні). Пріоритети переривань та механізм їх опрацювання. Контролер переривань NVIC. Взаємодія GPIO та системи переривань при опрацюванні подій.

Тема 4 Системний таймер. Таймери загального призначення

Принципи роботи системного таймера SysTick. Використання SysTick для реалізації затримок і керування задачами. Таймери загального призначення, Структура та режими роботи таймерів (лічильник, ШІМ, вимірювання частоти й тривалості імпульсів). Приклади застосування таймерів (опрацювання періодичних подій, керування двигунами, генерація сигналів).

Тема 5. Послідовні інтерфейси UART, SPI, I2C

Принципи роботи послідовних інтерфейсів. Інтерфейс UART, технічні особливості та режими роботи (асинхронна/синхронна передача, формат повідомлення, швидкість передачі даних та інше). Інтерфейс SPI, технічні особливості та режими роботи (“ведучий-підлеглий”, синхронізація та інше). Інтерфейс I2C технічні особливості та режими роботи (адресація пристроїв, ACK/NACK, напівдуплексний обмін даними). Приклади використання послідовних інтерфейсів для взаємодії із сенсорами та периферією.

Змістовий модуль 2. Операційні системи реального часу

Тема 6. Операційні системи реального часу. Структура FreeRTOS

Поняття ОСРЧ, їх відмінність від звичайних ОС. Основні вимоги до ОСРЧ (детермінованість, швидкодія, передбачуваність реакції). Архітектура та ключові компоненти FreeRTOS (ядро, задачі, планувальник). Типи планування задач у FreeRTOS (пріоритетне, витісняюче, кооперативне). Приклади застосування FreeRTOS у вбудованих системах.

Тема 7. Керування пам'яттю, задачами, чергами

Керування пам'яттю у FreeRTOS. Статичне та динамічне виділення пам'яті. Задачі у FreeRTOS. Створення, стан, пріоритети, переключення контексту між задачами. Черги у FreeRTOS. Передача повідомлень, синхронізація між задачами. Приклади обміну даними між сенсорами та інтерфейсами (SPI, I2C).

Тема 8. Опрацювання переривань, програмні таймери, мютекси, семафори

Опрацювання переривань у FreeRTOS. Програмні таймер, їх використання для відкладених і періодичних подій. Принципи роботи синхронізаційних примітивів. М'ютекси. Синхронізація доступу до спільних ресурсів. Запобігання конфліктам. Семафори (двійкові та лічильникові), їх роль у координації задач та управлінні подіями.

4. Структура залікового кредиту з дисципліни “Системи реального часу”

Денна форма навчання

Тема	Кількість годин					
	Лекції	Лабораторні заняття	Інд. робота	Тренінг	Самостійна робота	Контрольні заходи
Змістовий модуль 1. Основи програмування вбудованих систем						
Тема 1. Поняття вбудованих систем. Приклади застосування	2		1	3	7	Опитування під час заняття
Тема 2. Архітектура ARM процесорів. Розроблення ПЗ на мові C для ARM	4	2			11	Опитування під час заняття
Тема 3. Порти вводу-виводу. Система переривань ARM процесорів	4	2			11	Опитування під час заняття
Тема 4 Системний таймер. Таймери загального призначення	4	2			11	Опитування під час заняття
Тема 5. Послідовні інтерфейси UART, SPI, I2C	4	2			14	Опитування під час заняття
Змістовий модуль 2. Операційні системи реального часу						
Тема 6. Операційні системи реального часу. Структура FreeRTOS	4	2	2	3	14	Опитування під час заняття
Тема 7. Керування пам'яттю, задачами, чергами	4	4			14	Опитування під час заняття
Тема 8. Опрацювання переривань, програмні таймери, мютекси, семафори	4	-			15	Опитування під час заняття
Разом	30	14	3	6	97	

5. Тематика лабораторних занять

1. Встановлення та робота з інтегрованим середовищем проектування STM32-Cube. Робота із портами вводу-виводу загального призначення.
2. Робота із послідовним інтерфейсом UART. Обмін даними із комп'ютером.
3. Система переривань мікроконтролерів STM32. Опрацювання переривань.
4. Таймери мікроконтролерів STM32 та робота із ними.
5. Послідовні інтерфейси I2C, SPI. Робота із сенсорами та опрацювання даних сенсорів.
6. Розгортання ОС реального часу FreeRTOS. Створення прикладного ПЗ для роботи на FreeRTOS.

6. Самостійна робота

Студенти проводять дослідження в області проектування та реалізації систем реального часу та готують результати дослідження у вигляді реферату. Студенти можуть самостійно запропонувати та погодити з викладачем тему дослідження, або обрати одну із запропонованих за напрямками.

Загальні теми - дозволяють охопити базові поняття:

1. Архітектура вбудованих систем: класифікація та основні компоненти.
2. Порівняння мікроконтролерів і мікропроцесорів у вбудованих системах.
3. Використання ARM Cortex-M у сучасних вбудованих рішеннях.
4. Інтерфейси вводу-виводу (UART, SPI, I2C): особливості та застосування.
5. Енергоефективність у вбудованих системах: методи оптимізації.
6. Периферійні модулі мікроконтролерів та їхнє використання.
7. Системи реального часу: відмінності від звичайних обчислювальних систем.
8. Методи налагодження та тестування вбудованих систем.
9. Використання емуляторів і симуляторів у розробці вбудованих програм.
10. Сучасні засоби розробки ПЗ (STM32CubeIDE, Keil uVision, IAR).

FreeRTOS та операційні системи реального часу - дозволяють дати практичні аспекти застосування FreeRTOS:

11. Основи операційних систем реального часу (RTOS).
12. Архітектура FreeRTOS: ядро, завдання, планувальник.
13. Планувальники завдань у FreeRTOS: типи та принципи роботи.
14. Механізми синхронізації в FreeRTOS: семафори, м'ютекси, події.
15. Використання черг і повідомлень у FreeRTOS для міжпроцесорної комунікації.
16. Управління пам'яттю в FreeRTOS.

17. Таймери та відкладені функції (програмні таймери) у FreeRTOS.
18. Приклад розробки багатозадачної системи з використанням FreeRTOS.
19. FreeRTOS у промислових застосуваннях: приклади використання.
20. Порівняння FreeRTOS з іншими RTOS (Zephyr, ThreadX, VxWorks).
21. Організація паралельного опитування кількох сенсорів у FreeRTOS.

Спеціалізовані теми та застосування - дозволяють розглянути сучасне застосування систем реального часу (IoT, медицина, безпека, автомобільна промисловість та інші):

22. Архітектура шини AMBA та її роль у вбудованих системах.
23. Використання DMA у вбудованих системах.
24. Організація обміну даними через UART, SPI та I2C у реальному часі.
25. Вбудовані системи в автомобільній промисловості: стандарт AUTOSAR.
26. Використання вбудованих систем у медичних приладах.
27. Вбудовані системи в IoT-пристроях: виклики та перспективи.
28. Безпека у вбудованих системах: загрози та методи захисту.
29. Використання ОС у мікроконтролерах для розумних пристроїв.
30. Вбудовані системи для керування роботами та дронами.
31. Перспективи розвитку вбудованих систем: RISC-V та нові архітектури.
32. Побудова системи контролю руху об'єкта на основі даних акселерометра.
33. Виявлення падіння із використанням сенсорів руху та FreeRTOS.

Реферат повинен складатися з наступних розділів:

1. Вступ (актуальність теми, мета дослідження).
2. Основна частина (аналіз вибраної теми, теоретичні основи (залежно від обраної теми), процес проведення дослідження).
3. Висновки.
4. Список використаних джерел.
5. Додатки (за необхідності).

7. Тренінг з дисципліни

Тематика: Розроблення прототипу модуля для опрацювання даних сенсорів. Завдання, які необхідно виконати під час тренінгу:

1. Дослідження:

- виконати аналіз однієї із можливих областей застосування модуля виходячи із набору доступних сенсорів;
- визначити ключові функції, які потрібно реалізувати в застосунку.

2. Розроблення алгоритму роботи пристрою та програми для опрацювання сенсорних даних, який включає наступні етапи:

- отримати (зчитати) дані з обраних сенсорів, використовуючи HAL-драйвери STM32;
- виконати попередню фільтрацію та опрацювання сигналів: усереднення, цифрова фільтрація, компенсація шуму;
- реалізувати алгоритм виявлення ключових подій або параметрів, наприклад: рух, зміна орієнтації у просторі, перевищення порогового значення;
- оцінити коректність алгоритму шляхом виводу даних у STM32CubeMonitor, UART-консоль або на вбудований дисплей Discovery Kit.

3. Налаштування плати та інтеграція з FreeRTOS:

- за допомогою STM32CubeMX / STM32CubeIDE згенерувати базовий проект для обраної апаратної платформи;
- створити набір задач (tasks) для паралельного зчитування даних із сенсорів та подальшого їх опрацювання;
- налаштувати взаємодію між задачами через черги або семафори (наприклад, окреме завдання для сенсора й окреме для обробки/візуалізації результатів);
- згенерувати прошивку у форматі .elf / .hex для програмування плати.

4. Демонстрація:

- запрограмувати модуль та продемонструвати роботу прототипу (наприклад: визначення положення плати в просторі, вимірювання рівня освітленості, контроль температури чи тиску).

8. Засоби оцінювання та методи демонстрування результатів навчання

У процесі вивчення дисципліни “Системи реального часу” використовуються наступні засоби оцінювання та методи демонстрування результатів навчання, а саме:

- оцінювання результатів виконання лабораторних робіт;
- поточне тестування та опитування;
- оцінювання виконання завдань тренінгу;
- оцінювання виконання завдань самостійної роботи.

9. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни “ Системи реального часу” визначається як середньозважена величина, залежно від питомої ваги кожної складової залікового кредиту:

Модуль 1		Модуль 2	Модуль 3
40 %	40%	5 %	15 %
Поточне оцінювання	Модульний контроль	Тренінг	Самостійна робота
Виконання та захист лабораторних робіт (6 лабораторних робіт)	Модульна контрольна робота: тест, включає 50 запитань. Студенти проходять on-line у системі Moodle	Виконання завдань під час тренінгу відповідно до п.7 робочої програми	Виконання завдання для самостійної роботи

Модуль №1 включає дві складові за якими оцінюють знання студента:

1. **“Поточне оцінювання”** - оцінка визначається як середнє арифметичне оцінок, отриманих за виконання та захист лабораторних робіт.

ECTS, бали	Критерії оцінювання
А 90 – 100	Виконані всі завдання лабораторної роботи. Реалізований код програми є працездатним, оптимізованим та структурованим. Правильно виконані всі етапи роботи (налаштування середовища, програмування, відлагодження, перевірка результатів). Звіт оформлено акуратно з поясненням принципів роботи програми. Студент впевнено відповідає на запитання викладача, надає повні пояснення, вміє узагальнити результати та відповідає на додаткові запитання. Звіт оформлено акуратно з поясненням принципів роботи програм та без помилок.
В 85 – 89	Виконані всі основні завдання та частина додаткових. Результати працездатні, хоча присутні дрібні неточності у коді чи описі. Студент демонструє добре розуміння теми та пояснює ключові рішення. Звіт містить незначні недоліки в оформленні.
С 75 – 84	Виконані всі основні завдання, але без додаткових. Програма працює, але з обмеженнями (відсутні додаткові перевірки, код програми не оптимізований, відсутня обробка виняткових ситуацій). Частково зроблений аналіз отриманих результатів. У звіті є недоліки в оформленні або опис.
Д 65 – 74	Виконана лише частина основних завдань. Представлені рішення містять суттєві недоліки в коді або логіці роботи, але студент демонструє базове розуміння теми. Робота виконана за зразком, без власних пояснень. Аналіз та висновки неповні або неточні. У звіті є помилки чи пропуски.
Е 60 – 64	Виконано мінімальну частину базових завдань з допомогою викладача. Отримані результати обмежено працездатні. Аналіз та висновки відсутні або поверхневі. Звіт має значні недоліки.
FX 35 – 59	Більшість завдань не виконано. Код програми або відсутній, або непрацездатний. Студент не може пояснити свої дії та не розуміє суті лабораторної роботи. Звіт відсутній або не відповідає вимогам.

2. **“Модульний контроль”** - оцінка визначається за результатами складання тесту. Тест включає 50 питань теоретичного характеру, які охоплюють основні теми дисципліни “Системи реального часу”. Загальне число балів, що отримав студент, показує наскільки він має глибокі, міцні і системні знання із дисципліни.

Виконання тестових завдань оцінюється за 100-бальною шкалою, тому одна правильна відповідь на тестове завдання оцінюється кількістю балів, яка розраховується наступною формулою:

$$K = 100/N,$$

де: K - кількість балів за 1 питання; N - кількість питань у тесті.

Загальна кількість балів за весь тест оцінюється наступною формулою:

$$Z = K * P,$$

де: Z - загальна кількість балів; K - кількість балів за 1 запитання; P - кількість правильних відповідей на тестові завдання.

Модуль №2 - “Тренінг” – оцінка визначається, як середнє арифметичне оцінок, отриманих за виконання окремих завдань тренінгу, відповідно п. 7 даної робочої програми.

ECTS, бали	Критерії оцінювання
A 90 – 100	Завдання тренінгу виконані в повному обсязі та реалізовані на високому рівні з урахуванням всіх вимог. Показана креативність або оптимізація, що покращує результат. Студент упевнено та аргументовано пояснює прийняті рішення, відповідає на запитання. Демонструє впевнене володіння матеріалом. Продемонстровано якісний кінцевий програмний продукт/прототип.
B 85 – 89	Виконано всі основні завдання. Присутні незначні недоліки у деталях або оформленні. Результат відповідає вимогам та є цілком працездатним.
C 75 – 84	Виконано більшість завдань. Деякі вимоги реалізовані поверхово. Представлений результат працездатний, але має обмежену функціональність.
D 65 – 74	Завдання виконано частково. Присутні суттєві недоліки в реалізації або якості результату. Досягнуто мінімально необхідного рівня. Реалізована основна ідея, але без глибокої деталізації.
E 60 – 64	Виконано лише базову частину завдань. Робота відповідає мінімальним вимогам та демонструє обмежений рівень знань і практичних умінь.
FX 35 – 59	Завдання виконано частково, значна частина ключових етапів відсутня. Результат неповний або не відповідає встановленим вимогам. Демонстрація роботи програми/прототипу обмежена або неможлива.

Модуль №3 - “Самостійна робота” – оцінка визначається, як середнє арифметичне оцінок, отриманих під час виконання завдання самостійної роботи, а саме: 1) представлення проведеного дослідження з обраної теми; 2) аналіз отриманих результатів.

ECTS, бали	Критерії оцінювання
A 90 – 100	Робота виконана повністю, тема розкрита всебічно. Використано достатню кількість джерел, є чітка структура (вступ, основна частина, висновки). Аналіз результатів глибокий та аргументований, зроблено власні висновки. Оформлення відповідає вимогам.
B 85 – 89	Тема досліджена на високому рівні, використано якісні джерела. Присутня логічна структура та аналіз, проте глибина висновків або критичний огляд обмежені. Невеликі недоліки в оформленні.
C 75 – 84	Виконано основний обсяг роботи, тема розкрита частково. Аналіз поверховий, висновки загальні. Використано обмежену кількість джерел, але робота залишається цілісною та зрозумілою.
D 65 – 74	Тема досліджена неповністю, значна частина матеріалу подана описово без належного аналізу. Висновки слабкі або нечіткі. Є помітні недоліки в структурі та оформленні.
E 60 – 64	Представлено лише базову інформацію по темі, відсутній достатній аналіз. Висновки формальні. Робота відповідає мінімальним вимогам.
FX 35 – 59	Виконано частину завдання. Тема розкрита поверхово, аналіз практично відсутній. Структура нечітка, джерела використані обмежено. Робота не відповідає більшості вимог

Шкала оцінювання

За шкалою ЗУНУ	За національною шкалою	За шкалою ECTS
90-100	Відмінно	A (відмінно)
85-89	Добре	B (дуже добре)
75-84		C (добре)
65-74	Задовільно	D (задовільно)
60-64		E (достатньо)
35-59	Незадовільно	FX (незадовільно, з можливістю повторного складання)
1-34		F (незадовільно, з обов’язковим повторним курсом)

**10. Інструменти, обладнання та програмне забезпечення,
використання яких передбачає навчальна дисципліна**

№	Найменування	Номер теми
1.	Мультимедійний проектор	1-8
2.	Комп'ютери з доступом до мережі Інтернет	1-8
3.	Програмне забезпечення STM32-CubeMX фірми Semiconductor Tomson	3-8
4	Програмне забезпечення STM32-CubeAI фірми Semiconductor Tomson	3-8
5	Програмне забезпечення STM32-CubeIDE фірми Semiconductor Tomson	3-8
6	Плати B-L475E-IOT01A2 та STM32L476 Discovery Kit	3-8, тренінг

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

1. Yifeng Zhu Embedded Systems with ARM Cortex-M Microcontrollers in Assembly Language and C. 4th ed. - E-Man Press LLC. – 2023. – 730 p.
2. Israel Gbati Bare-Metal Embedded C Programming - Packt Publishing. – 2024. – 415 p.
3. Alexander G. Dean Embedded Systems Fundamentals with Arm Cortex-M based Microcontrollers. Practical Approach. 2nd ed. - Arm Education Media. – 2021. – 340 p.
4. Trevor Martin The Designer's Guide to the Cortex-M Processor Family 3rd ed. - Elsevier Ltd. – 2023. – 624 p.
5. Mark Fisher ARM Cortex-M4 Cookbook - Packt Publishing. – 2016. – 276 p.
6. Majid Pakdel Advanced Programming with STM32 Microcontrollers - Elektor International Media – 2020. – 216 p.
7. Brian Amos Hands-On RTOS with Microcontrollers - Packt Publishing. – 2020. – 462 p.
8. Dogan Ibrahim ARM-Based Microcontroller Multitasking Projects Using the FreeRTOS Multitasking Kernel - Elsevier Ltd. – 2020. – 501 p.
9. Warren Gay Beginning STM32 Developing with FreeRTOS, libopenm3, and GCC. 2nd ed. – Apress. – 2024. – 527 p.
10. Злобін Г.Г. Основи алгоритмізації та програмування мовою Сі. Підручник – Каравела. – 2022. – 168 с.
11. STM32CubeIDE basics [Електронний ресурс]. Режим доступу: https://www.st.com/content/st_com/en/support/learning/stm32-education/stm32-moocs/STM32CubeIDE_basics_MOOC.html
12. STM32L4 training MOOC [Електронний ресурс]. Режим доступу: https://www.st.com/content/st_com/en/support/learning/stm32-education/stm32-moocs/getting-familiar-with-stm32l4-in-hands-on-sessions.html
13. STM32L4 workshop MOOC [Електронний ресурс]. Режим доступу: https://www.st.com/content/st_com/en/support/learning/stm32-education/stm32-moocs/stm32l4-hands-on-workshop.html
14. FreeRTOS on STM32 v2 [Електронний ресурс]. Режим доступу: https://www.st.com/content/st_com/en/support/learning/stm32-education/stm32-moocs/freertos-common-microcontroller-software-interface-standard-osv2.html
15. ARM University Program Efficient Embedded Systems Design Education Kit [Електронний ресурс]. Режим доступу: <https://www.arm.com/resources/education/education-kits/efficient-embedded-systems>
16. ARM University Program Real-Time Operating Systems Education Kit [Електронний ресурс]. Режим доступу: <https://www.arm.com/resources/education/education-kits/real-time-operating-systems>
17. Електронний курс з дисципліни «Системи реального часу» для студентів галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки» на платформі Moodle ЗУНУ /Майків І.М. - Тернопіль, 2025 <https://moodle.wunu.edu.ua>