

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Львівський національний університет імені Івана Франка**  
**Факультет електроніки та комп'ютерних технологій**  
**Кафедра радіоелектронних і комп'ютерних систем**

**Затверджено**

На засіданні  
кафедри радіоелектронних  
і комп'ютерних систем  
факультету електроніки та комп'ютерних  
технологій  
Львівського національного університету  
імені Івана Франка  
(протокол № *1/24* від *28.08.* 2023 р.)

Завідувач кафедри:

  
\_\_\_\_\_ Ігор ОЛЕНИЧ

**Силабус із навчальної дисципліни**  
**“Мікроконтролери (на базі GL Embedded Starter Kits)”**  
**що викладається в межах ОПП**  
**“Інженерія програмного забезпечення”**  
**першого (бакалаврського) рівня вищої освіти для здобувачів з**  
**спеціальності 121 – Інженерія програмного забезпечення**

Львів 2023 р.

<b>Назва дисципліни</b>	Мікроконтролери (на базі GL Embedded Starter Kits)
<b>Адреса викладання дисципліни</b>	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, вул. Драгоманова 50, м. Львів, 79005, вул. Ген. Тарнавського 107, м. Львів, 79011
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет електроніки та комп'ютерних технологій Кафедра радіоелектронних і комп'ютерних систем
<b>Галузь знань, шифр та назва спеціальності</b>	12 – інформаційні технології 121 – Інженерія програмного забезпечення
<b>Викладачі дисципліни</b>	Бойко Ярослав Васильович, канд. фіз.-мат. наук, доцент кафедри радіоелектронних і комп'ютерних систем
<b>Контактна інформація викладачів</b>	<a href="mailto:yaroslav.boйко@lnu.edu.ua">yaroslav.boйко@lnu.edu.ua</a> , <a href="https://electronics.lnu.edu.ua/employee/bojko-ya-v">https://electronics.lnu.edu.ua/employee/bojko-ya-v</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю): ауд. 101, корпус факультету електроніки та комп'ютерних технологій, вул. Драгоманова 50, м. Львів
<b>Сторінка курсу</b>	<a href="https://e-learning.lnu.edu.ua/course/view.php?id=4584">https://e-learning.lnu.edu.ua/course/view.php?id=4584</a>
<b>Інформація про дисципліну</b>	Дисципліна “Мікроконтролери (на базі GL Embedded Starter Kits)” належить до циклу професійної та практичної підготовки за блоками вибіркового навчання спеціальності 121 – Інженерія програмного забезпечення для освітньої програми “Інженерія програмного забезпечення”, яка викладається в шостому семестрі в обсязі 5,5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
<b>Коротка анотація дисципліни</b>	Вивчення основ програмування мікроконтролерів у межах курсу здійснюється на базі мікроконтролерів STM32, доповнених платами виробництва компанії GlobalLogic, які розширюють комунікаційні можливості та способи відображення інформації.
<b>Мета та цілі дисципліни</b>	<i>Мета:</i> формування у студентів знань про архітектуру мікроконтролерів та програмованих систем на кристалі, принципів створення програмного забезпечення мікроконтролерних систем. <i>Цілі:</i> формування у студентів практичних навичок проектування інформаційно-вимірювальних мікроконтролерних систем, використання спеціалізованих програмних інструментів для програмування мікроконтролерів.
<b>Література для вивчення дисципліни</b>	1) Carmine Noviello. <i>Mastering STM32: A step-by-step guide to the most complete ARM Cortex-M platform, using the official STM32Cube, 2nd Edition.</i> – Leanpub, 2022. – 910 p. 2) Trevor Martin. <i>The Designer's Guide to the Cortex-M Processor Family, 3rd Edition.</i> – Elsevier, 2022. – 650 p. 3) Cem Unsalan, Huseyin Deniz Gurhan, Mehmet Erkin Yucel. <i>Embedded System Design with ARM Cortex-M Microcontrollers: Applications with C, C++ and MicroPython.</i> – Springer, 2022. – 576 p.

	<p>4) Majid Pakdel. <i>Advanced Programming with STM32 Microcontrollers: Master the Software Tools Behind the STM32 Microcontroller</i>. – Elektor Verlag, 2020. – 216 p.</p> <p>5) Cem Unsalan, Duygun E. Barkana, H. Deniz Gurhan. <i>Embedded Digital Control with Microcontrollers: Implementation with C and Python</i>. Wiley. – 2021. – 364 p.</p> <p>6) Dong-Jin Lim. <i>Control Systems Engineering: Design and Implementation using Arm Cortex-M Microcontrollers</i>, Dr. Dong-Jin Lim, 2021. – 449 p.</p> <p>7) Mazidi, Muhammad Ali &amp; Chen, Shujen &amp; Ghaemi, Eshragh. <i>STM32 Arm Programming for Embedded Systems: Using C Language with STM32 Nucleo</i>. MicroDigitalEd, 2018. – 378 p.</p> <p>8) Joseph Yiu. <i>The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors, 3rd Edition</i>. Newnes, 2014. – 477 p.</p> <p>9) Mark Fisher. <i>ARM Cortex M4 Cookbook</i>. Packt Publishing, 2016. – 463 p.</p> <p>10) Agus Kurniawan. <i>STM32 Nucleo-32 Development Workshop</i> PE Press, 2018. – 194 p.</p>
<b>Обсяг курсу</b>	Загальний обсяг: 165 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 101 год.
<b>Очікувані результати навчання</b>	<p>У результаті вивчення даного курсу студент буде:</p> <p><b>знати:</b></p> <ul style="list-style-type: none"> <li>• систематику та основні тенденції розвитку мікроконтролерної техніки;</li> <li>• теоретичні основи та практичні аспекти архітектури мікроконтролерів;</li> <li>• програмні інтерфейси мікроконтролерів;</li> <li>• призначення та функції інструментальних засобів для створення програмного забезпечення мікроконтролерів;</li> <li>• основи роботи з периферією МК;</li> <li>• порядок розробки та відлагодження програмного забезпечення МК:</li> </ul> <p><b>вміти:</b></p> <ul style="list-style-type: none"> <li>• використовувати інструментальні програмні засоби для розробки програмного забезпечення МК;</li> <li>• застосовувати бібліотеки функцій, що надаються системами розробки МК;</li> <li>• вирішувати питання організації взаємодії МК з периферійними пристроями;</li> <li>• реалізовувати проекти з використанням МК як елементами ІоТ, робототехніки тощо.</li> </ul> <p>Після вивчення даного курсу «Мікроконтролери (на базі GL Embedded Starter Kits)» здобувачі набудуть таких Загальних і Фахових компетентностей та Програмних результатів навчання:</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК02. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК04. Здатність спілкуватися іноземною мовою як усно, так і письмово.</p> <p>ЗК05. Здатність вчитися і оволодівати сучасними знаннями.</p> <p>ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p>ФК14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.</p> <p>ФК15. Здатність розробляти архітектури, модулі та компоненти програмних систем.</p> <p>ФК17. Здатність дотримуватися специфікацій, стандартів, правил і</p>

	<p>рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>ФК25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p> <p>ФК26. Здатність до алгоритмічного та логічного мислення.</p> <p>ФК27. Здатність використовувати для розробки програмного забезпечення перспективні засоби та технології, зокрема, науки про дані, штучного інтелекту, IoT, вбудованих систем тощо.</p> <p>ПРН06. Вміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.</p> <p>ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.</p> <p>ПРН11. Вибирати вихідні дані для проектування, куруючись формальними методами опису вимог та моделювання.</p> <p>ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p> <p>ПРН15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.</p> <p>ПРН17. Вміти застосовувати методи компонентної розробки програмного забезпечення.</p> <p>ПРН18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.</p>
<b>Ключові слова</b>	Методи обчислень, інтерполяція функції, чисельне розв'язування рівнянь та систем рівнянь.
<b>Формат курсу</b>	Очний
<b>Теми</b>	<ol style="list-style-type: none"> <li>1. Вступ. Класифікація та огляд мікроконтролерних систем.</li> <li>2. Мікроконтролери виробництва компанії STMicroelectronics – загальна характеристика.</li> <li>3. Проектування та розробка програмного забезпечення для мікроконтролерів STM32.</li> <li>4. Загальна характеристика STM32F407.</li> <li>5. Центральний процесор і підсистема пам'яті STM32F407.</li> <li>6. GPIO</li> <li>7. Базова архітектура переривань STM32F407.</li> <li>8. Системні ресурси.</li> <li>9. Аналогові блоки.</li> <li>10. Спеціалізована периферія.</li> <li>11. Цифрові блоки з фіксованими функціями.</li> <li>12. Апаратні та програмні засоби налагодження STM32F407.</li> </ol>
<b>Підсумковий контроль, форма</b>	Залік у кінці семестру
<b>Пререквізити</b>	<p>Для вивчення даного курсу студентам потрібні базові знання з курсів:</p> <ul style="list-style-type: none"> <li>- Основи програмування,</li> <li>- Архітектура комп'ютерів і комп'ютерна електроніка,</li> <li>- Комп'ютерні інформаційні мережі та системи,</li> <li>- Вступ в інженерію програмного забезпечення.</li> </ul>
<b>Навчальні методи та</b>	Інформаційні методи (лекція, бесіда, ілюстрація, демонстрація); дедуктивні методи на основі узагальнень;

<p><b>техніки, які будуть використовуватися під час викладання курсу</b></p>	<p>евристичні методи (проблемна лекція); інтерактивні методи (дискусія).</p>
<p><b>Необхідне обладнання</b></p>	<p>Для проведення лекційних занять: комп'ютер (мінімальні характеристики: процесор з архітектурою x86_64, 8ГБ оперативної пам'яті, 512ГБ HDD/SSD), доступ до мережі Internet, засоби мультимедіа (в т.ч. проектор). Для проведення лабораторних занять: Комп'ютери (мінімальні характеристики: процесор з архітектурою x86_64, 8ГБ оперативної пам'яті, 512ГБ HDD/SSD). Набори GL Embedded Starter Kit, генератори сигналів, осцилографи. Необхідне програмне забезпечення включає в себе ОС Linux, середовище розробки STM32CubeIDE. LMS Moodle.</p>
<p><b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b></p>	<p>Оцінювання проводиться за 100-бальною шкалою. Бали нараховуються за таким співвідношенням:</p> <ul style="list-style-type: none"> <li>• Змістовий модуль 1: до 14 балів за виконання модульного завдання – 50 тестових завдань.</li> <li>• Змістовий модуль 2: до 14 балів за виконання модульного завдання – 50 тестових завдань..</li> <li>• Виконання лабораторних робіт: до 72 балів.</li> </ul> <p>Підсумкова максимальна кількість балів 100. Звітність за курс – залік у кінці семестру.</p> <p><b>Змістовий модуль</b> – самостійна робота студента у вигляді проходження тестових завдань у системі електронного навчання Moodle. Питання до тестового завдання вибираються випадковим чином із банку завдань. Тематика завдань подана у розділі <b>Питання до модульного контролю</b>.</p> <p><b>Академічна доброчесність:</b> Очікується, що роботи студентів будуть оригінальними дослідженнями чи міркуваннями. Списування та втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності у виконанні завдань є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману. Жодні форми порушення академічної доброчесності не толеруються.</p> <p><b>Відвідання занять</b> є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні зайняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів робіт, передбачених курсом.</p> <p><b>Література.</b> Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.</p> <p><b>Політика виставлення балів.</b> Враховуються бали, набрані при поточному контролі та бали за виконання лабораторних робіт. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими</p>

	<p>мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.</p> <p><b>Оцінювання лабораторних робіт</b> (12 лабораторних робіт, максимальна кількість балів: 72) відбувається шляхом оцінки роботи студента під час проведення лабораторної роботи в аудиторії та захисту звіту по виконаній лабораторній роботі (0-6 балів за одну роботу).</p> <p>Бали оцінювання лабораторних робіт нараховуються за наступним співвідношенням:</p> <p>6 – студент в повному обсязі володіє навчальним матеріалом, має повне розуміння розглянутої теми, надає правильні відповіді на запитання по темі, код програми функціонує відповідно до завдання, містить стандартні засоби тестування та коментарі;</p> <p>5 – студент в повному обсязі володіє навчальним матеріалом, має повне розуміння розглянутої теми, надає правильні відповіді на запитання по темі, код програми функціонує відповідно до завдання;</p> <p>4 – студент достатньо розуміє розглянутий матеріал та принципи написаного ним коду програми, присутні неточності та незначні помилки у відповідях на запитання по темі, код програми функціонує відповідно до завдання (або з несуттєвими недоліками);</p> <p>3 – студент не досить добре розуміє розглянутий матеріал та написаний ним код програми, вагається та надає неточні/не конкретні відповіді на запитання по темі, код програми функціонує неточно, або з помірними недоліками;</p> <p>2 – студент погано розуміє розглянутий матеріал та написаний ним код програми, студент в більшості надає помилкові відповіді на питання по темі, код програми функціонує з суттєвими недоліками;</p> <p>1 - студент погано розуміє розглянутий матеріал та написаний ним код програми, код програми не функціонує належним чином;</p> <p>0 - студент зовсім не засвоїв розглянутий матеріал, написаний ним код програми не відповідає темі/не функціонує взагалі.</p> <p><b>Оцінювання змістових модулів</b> (2 змістових модулі, 28 балів сумарно) — за результатами проходження студентом тестів.</p> <p><b>Критерії оцінювання результатів неформальної освіти:</b>  Нарахування балів відбувається за написання студентом тез доповідей на конференціях, наукових статей, участь у діяльності наукових гуртків, участь у наукових семінарах та круглих столах, конкурсах, участь у заходах неформальної освіти за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Prometheus тощо), курсах на провідних ІТ компаніях за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів відповідної активності до вимог результатів навчання з навчальної дисципліни.</p>
<p><b>Питання до заліку</b></p>	<p>Визначення і сфери застосування МК — особливості МК як обчислювальних систем, типи мікропроцесорів, застосування в системах автоматичного контролю.</p> <p>Типи вбудованих систем — мікрокомп'ютерні та мікроконтролерні вбудовані системи.</p>

Класифікація та порівняльний аналіз на основі базових архітектур і виробників -- Intel 8051, Atmel AVR, Microchip PIC, ARM, Renesas H8, SuperHTM, M16C. Motorola/Freescale/NXP families.

Історія розвитку індустрії мікроконтролерів — від мікропроцесорів Intel 4004 (1971), TI TMS1802NC (1972), Intel 8051 (1976) до сучасного стану елементної бази та комплектування (Cypress PSoC, STM32).

Гарвардська архітектура та архітектура Фон-Неймана -- відмінності у розподілі адресних просторів та особливостях функціонування мікропроцесора.

Вузькоспеціалізовані системи — цифрові сигнальні процесори (DSP), інтегральні схеми для специфічного застосування (ASIC).

Системи-на-чипі (SoC) — мікроконтролерні та мікропроцесорні SoC, програмовані SoC (PSoC).

МК як складова IoT — визначення IoT та роль мікроконтролерної техніки. Edge/Fog- обчислення. Розширення STM32Cube.AI.

Загальна характеристика, продуктова лінійка, серії — загальні основи будови MCU STM32 (ядро процесора ARM Cortex M, статична оперативна пам'ять, флеш-пам'ять, інтерфейс налагодження, периферійні пристрої). Огляд характеристик: ширина слів для даних, розрядність, архітектура, конвеєр, набір інструкцій, організація пам'яті програм, буфер передвибірки, середній розмір інструкції, тип переривань, затримка реагування на переривання, режими управління енергоспоживанням, налагоджувальний інтерфейс. Серії STM32 H7, F7, F4, F3, F2, F1, F0, G0, L5, L4, L1, L0, W, J (відмінності, застосування).

Представлення периферійних блоків — підблок, який містить кінцеві автомати, лічильники, комбінаторну та послідовну логіку (призначений для виконання завдань, які не потребують участі процесора); підблок налаштування та управління периферією, яке здійснюється програмою через регістри, з'єднані з внутрішньою шиною, що розділена з іншими ресурсами MCU.

Типові структури програм та особливості програмування мікроконтролерів. Низько- та високорівневі засоби програмування МК. Синтаксис та приклади асемблерного коду для STM32. Cortex Microcontroller Software Interface Standard (CMSIS).

Особливості застосування мови C. Переозначені типи даних. Системи числення (коротке повторення). Компілятори. MISRA C.

Хід виконання програм. Опитування. Переривання. Багатозадачні системи. Середовище розробки STM32CubeIDE. Основні можливості (підтримка STMicroelectronics ST-LINK та Segger J-Link, імпорт проектів з Atollic TrueSTUDIO та AC6 System Workbench) та етапи роботи в середовищі (вибір мікроконтролера STM32, конфігурація Pinout, інтегрованої периферії, таймерів і проміжного програмного забезпечення, створення проекту та генерація коду ініціалізації).

Архітектура, властивості, блок-діаграма. Базовий мікропроцесор ARM Cortex M4:

- Повнофункціональний набір інструкцій ARMv7-M, оптимізований для вбудованих додатків

- Модуль обчислень з рухомою комою (FPU)
- Блок захисту пам'яті (MPU)
- Вкладений векторний контролер переривання (NVIC)
- Можливості відстеження, точки зупинки та JTAG
- Розширена архітектура шин мікроконтролерів (AMBA)
- Вдосконалена високопродуктивна шина (AHB5, AHB-Lite)

Кількісні характеристики (частоти, розміри пам'яті різних типів). Внутрішні шини. Системні тактові сигнали. Тактові сигнали периферійних пристроїв.

Архітектура та набір команд мікропроцесора ARM Cortex-M4. Регістри — загального призначення та спеціальні (program status, exception mask, control registers). Регістри FPU. Статусні прапорці, Q-статусний прапорець. Біти GE. Набір інструкцій: інструкції переміщення в межах процесора, інструкції доступу до пам'яті, арифметичні операції, логічні операції, інструкції зсуву та обертання, операції перетворення даних, операції опрацювання бітових полів, інструкції опрацювання виняткових ситуацій. Розширення DSP. SIMD-розширення. Множення і MAC-інструкції. FPU-інструкції.

Набори інструкцій Thumb (16-розрядні інструкції) Thumb-2 (змінна довжина інструкцій). Pipelining, розгалуження, передбачення розгалужень. Фіксований адресний простір пам'яті. Технологія Bit-Banding та її використання. Вирівняний та невирівняний доступ до пам'яті. Стекова пам'ять. Модуль захисту пам'яті (MPU).  
Типи пам'яті — Flash, EEPROM, SRAM.

Відображення периферії STM32 і дескриптори HAL. Конфігурування GPIO.

Режими GPIO (High Impedance Analog, High Impedance Analog, Resistive Pull Up, Open Drain, Drives Low, Open Drain, Drives High, Strong Drive, Resistive Pull Up & Down. Регістр значення виводів, регістр встановлення, регістр скидання, регістр напрямку GPIO. Особливості програмування GPIO. Переривання від GPIO. Альтернативні функції GPIO.

Джерела переривань — апаратні та програмні переривання. Exceptions (винятки) — Reset, NMI, Hard Fault, Memory Management Fault, Bus Fault, Usage Fault, SVCCall, Debug Monitor, PendSV, SysTick. Послідовність входжень винятків і стекування. Виконання обробників винятків. Інтерфейс обробників переривань. Алгоритм опрацювання (життєвий цикл) переривань. Регістри вмикання переривань. Регістр програмного перемикавання переривань. Спеціальні регістри винятків і маскування переривань. Відображення таблиці векторів в пам'яті та її переміщення. Пріоритети. Затримки переривань. Блок-діаграма NVIC. Зовнішні лінії та NVIC. Дозвіл переривань (використання STM32CubeMX).

Система керування живленням. Режими зниженого енергоживлення. Режим сну. System control register (SCR). Умови виходу. Контролер переривань пробудження. Інструкції WFI, WFE. Розробка програм із зниженим енергоспоживанням.  
Таймери, головні внутрішні осцилятори, високо- і низькочастотні таймери. Таймери загального призначення. Режим захоплення. Режим лічильника



	<p>зовнішніх подій. Режим збігу. Регістри керування таймерами. Структура модуля таймерів/лічильників. Сторожовий таймер. Модуль годинника реального часу. Регістри керування модулем RTC.</p> <p>Система введення/виведення (огляд, основи використання GPIO).</p> <p>Аналогово-цифровий перетворювач. Основи successive approximation (SAR) АЦП. Модуль HAL_ADC. Режими перетворення: Single Channel, Single Conversation, Scan Single Conversion, Single Channel, Continuous Conversion, Scan Continuous Conversion, Injected Conversion, Dual.</p> <p>Цифрово-аналоговий перетворювач. Модуль HAL_DAC. Ручне керування ЦАП та використання таймера.</p> <p>Controller Area Network. Загальна інформація, Керуючі регістри контролерів CAN. Регістри фільтрації повідомлень. Топологія мережі CAN, область застосування, протоколи високого рівня.</p> <p>Direct Memory Access. Модуль HAL_DMA. Синхронізація DMA. Пересилання з пам'яті в пам'ять. Пакутна передача. Підтримка модулем DMA периферійних пристроїв. Пересилання несуміжних даних. Регістри керування модулем DMA. Передавання даних в режимі переривання. Передавання Peripheral-to-Peripheral. Використання STM32CubeMX для конфігурування DMA. Аналіз продуктивності DMA.</p> <p>Широтно-імпульсна модуляція та її режими (mode 1, mode 2). Використання STM32CubeMX для конфігурування режимів PWM. Одноімпульсний режим. Режим енкодера.</p> <p>Послідовні комунікаційні блоки та їх режими.</p> <p>I2C. Загальна характеристика інтерфейсу. Режими роботи інтерфейсу I2C мікроконтролерів сімейства STM32. Опис виводів і керуючих регістрів.</p> <p>SPI. Передача даних в режимі ведучого та веденого. Переривання SPI.</p> <p>UART, Характеристика модулів UART MCU STM32F407. Конфігурування та передавання даних.</p> <p>Винятки, які стосуються помилок. Послідовність опрацювання помилок Cortex-M та порядок викликів ARM. Регістри Cortex-M для аналізу помилок CFSR, HFSR, DFSR, MMFAR, BFAR.</p> <p>Програмні засоби налагодження коду середовища STM32CubeIDE — Expressions, Memory Monitors, Watchpoints, Instruction Stepping Mode, Registers View. Debug Port (DP), Access Port (AP), and Debug Access Port (DAP). Інтерфейс трасування. Режими та події налагодження. Модулі Flash patch and breakpoint (FPB), Data watchpoint and trace (DWT), Instrumentation trace macrocell (ITM). АНВ access port (АНВ-AP). ARM Semihosting.</p> <p>Зовнішні налагоджувачі. ST-LINK, SEGGER J-Link, Інтерфейси JTAG та SWD./</p>
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

**Схема курсу “Мікроконтролери (на базі GL Embedded Starter Kits)”  
для студентів спеціальності 121 – Інженерія програмного забезпечення**

<b>Тиж.</b>	<b>Тема, план, короткі тези</b>	<b>Форма діяльності (заняття)* *лекція, самостійна, дискусія, групова робота)</b>	<b>Література.** * Ресурси в інтернеті</b>	<b>Завдання, год</b>	<b>Термін виконання</b>
1,2	Вступ. Класифікація та огляд мікроконтролерних систем.	Лекція	[1], [2], [3], [5], Сайт курсу	3	кінець поточного тижня
1,2	Ознайомлення із засобами програмування STM32F407 – STM32CubeIDE. Ознайомлення із схемним редактором для конфігурування периферії STM32CubeMX	Лабораторна робота	Сайт курсу	3	кінець поточного тижня
2,3	Мікроконтролери виробництва компанії STMicroelectronics – загальна характеристика	Лекція	[1–7], [9], Сайт курсу	4	кінець поточного тижня
3,4	Реалізація проекту програмного і апаратного (PWM) керування яскравістю світлодіода.	Лабораторна робота	Сайт курсу	4	кінець поточного тижня
4,5	Проектування та розробка програмного забезпечення для мікроконтролерів STM32	Лекція	[1], [2–8], [10], Сайт курсу	4	кінець поточного тижня
5	Керування блоком PWM за допомогою USART	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
6	Загальна характеристика STM32F407	Лекція	[1–5], [9], [10], Сайт курсу	2	кінець поточного тижня
6	Вимірювання аналогових сигналів з відображенням на LCD-дисплеї	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
7	Центральний процесор і підсистема пам'яті STM32F407	Лекція	[1], [2], [8], Сайт курсу	2	кінець поточного тижня
7	Приклади використання шини CAN	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
8,9	GPIO	Лекція	[1], [6], [10], Сайт курсу	4	кінець поточного тижня
8,9	Опрацювання сигналів MEMS-пристроїв	Лабораторна робота	Сайт курсу	4	кінець поточного тижня
10,11	Базова архітектура переривань STM32F407	Лекція	[1], [2], [5], Сайт курсу	4	кінець поточного тижня
10	Керування LED за допомогою механізму переривань	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
12	Системні ресурси	Лекція	[1], [2], [7], [10],	2	кінець поточного тижня
11,12	Організація мережної взаємодії за	Лабораторна	Сайт курсу	4	кінець

	протоколом Ethernet	робота			поточного тижня
13	Аналогові блоки	Лекція	[1], [5], [8], [10],	2	Кінець поточного тижня
13	Реалізація генераторів сигналів різної форми	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
14	Спеціалізована периферія	Лекція	[1], [2], [3], Сайт курсу	2	кінець поточного тижня
14	Реалізація простої системи вимірювання частоти сигналу	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
15	Цифрові блоки з фіксованими функціями	Лекція	[1], [2--8] Сайт курсу	2	кінець поточного тижня
15	Вимірювання температури цифровим датчиком	Лабораторна робота	Сайт курсу	2	кінець поточного тижня
16	Апаратні та програмні засоби налагодження (debugging) STM32F407	Лекція	[1], [2], [9], Сайт курсу	2	кінець поточного тижня
16	Приклади використання DMA	Лабораторна робота	Сайт курсу	2	кінець поточного тижня