****

Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Лабораторна робота №6  
Проектування вбудованих систем

«Фінальний застосунок. ADC, GPIO, I2C, FreeRTOS»

Варіант 17

Виконав Перевірив:

студент групи ІА–13: Гордієнко Н.Ю.

Окаянченко Д.О.

Київ 2025

**Мета роботи:** Навчитись працювати з сенсорами та периферією поєднуючи їх в одному застосунку.

**Посилання на гітлаб-репозиторій:** <https://gitlab.com/david.okayach/designing-embedded-systems>.

## Хід роботи

1. Налаштуємо ADC інтерфейс:

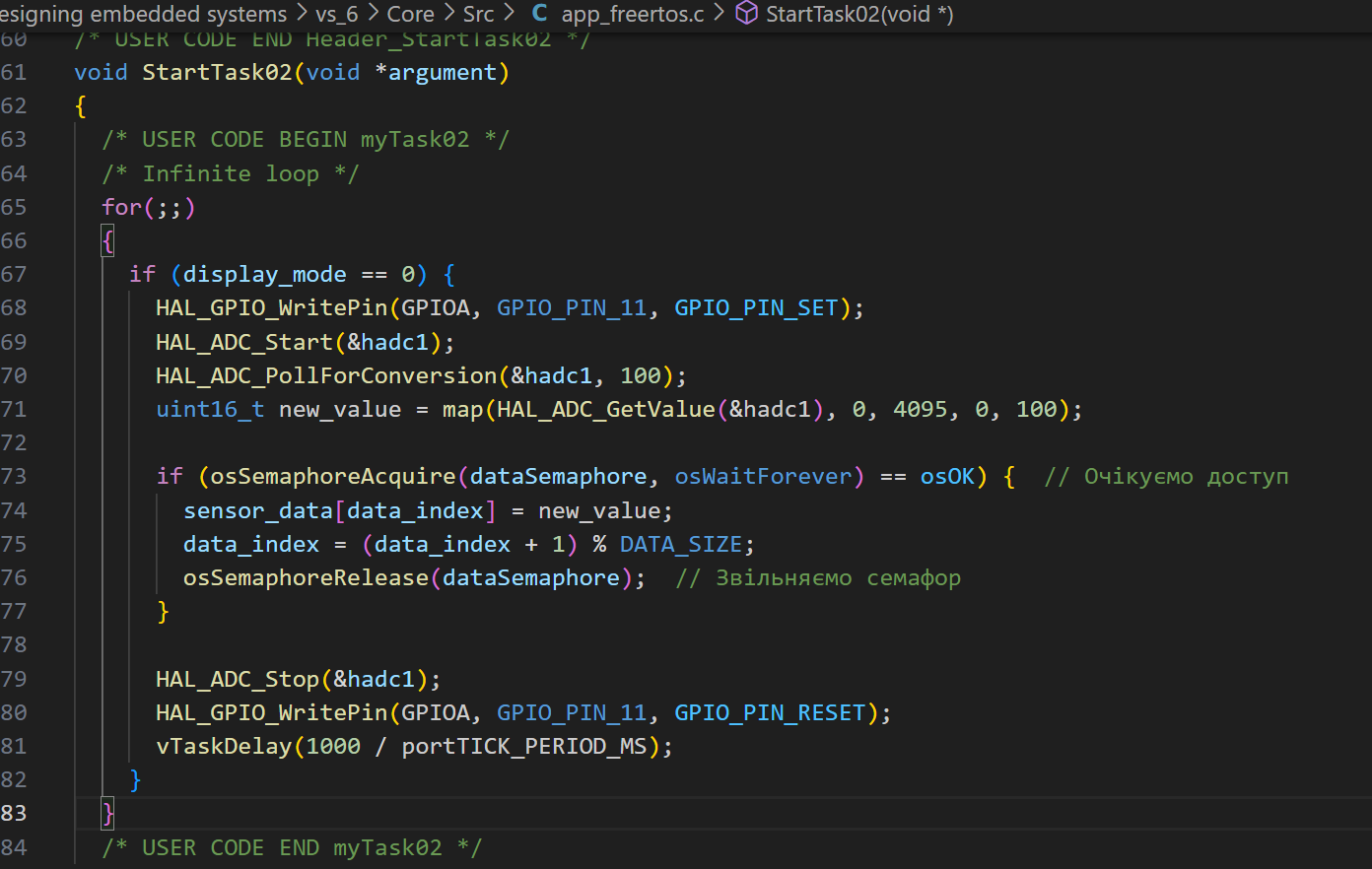
У даній лабораторній роботі розглядається використання FreeRTOS для реалізації багатопотокового керування у вбудованих системах. Основною метою є створення застосунку, який буде взаємодіяти з периферійними пристроями мікроконтролера, такими як ADC (аналогово-цифровий перетворювач), GPIO (загальне введення/виведення), I2C (послідовний інтерфейс обміну даними) та LCD-дисплей.

Для ефективної обробки та відображення інформації використовується багатопоточність, що дозволяє паралельно виконувати декілька завдань:

* Читання даних з сенсорів за допомогою ADC
* Обробка натискань кнопок для зміни режиму відображення
* Відображення отриманої інформації на LCD

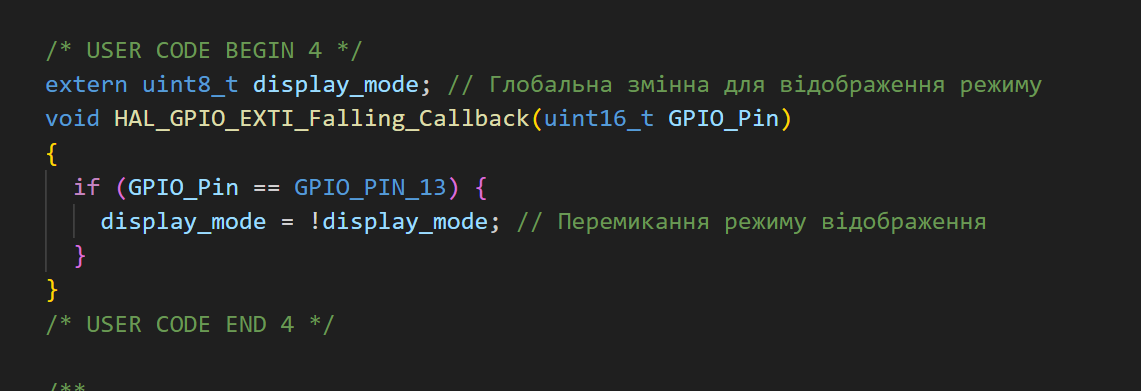
З метою забезпечення узгодженого доступу до спільних ресурсів застосовуються механізми синхронізації у FreeRTOS. Зокрема, використовується семафор (Semaphore) для захисту буфера даних, що зберігає виміряні значення сенсора. Це запобігає конфліктам при одночасному доступі до спільної пам’яті з різних потоків.

У рамках цієї лабораторної роботи було створено застосунок, що зчитує дані з потенціометра, який підключений до аналогово-цифрового перетворювача (ADC) мікроконтролера. Оскільки потенціометр видає значення в діапазоні від 0 до 4095, застосунок перетворює ці дані у шкалу від 0 до 100, що робить їх більш зручними для візуального сприйняття.

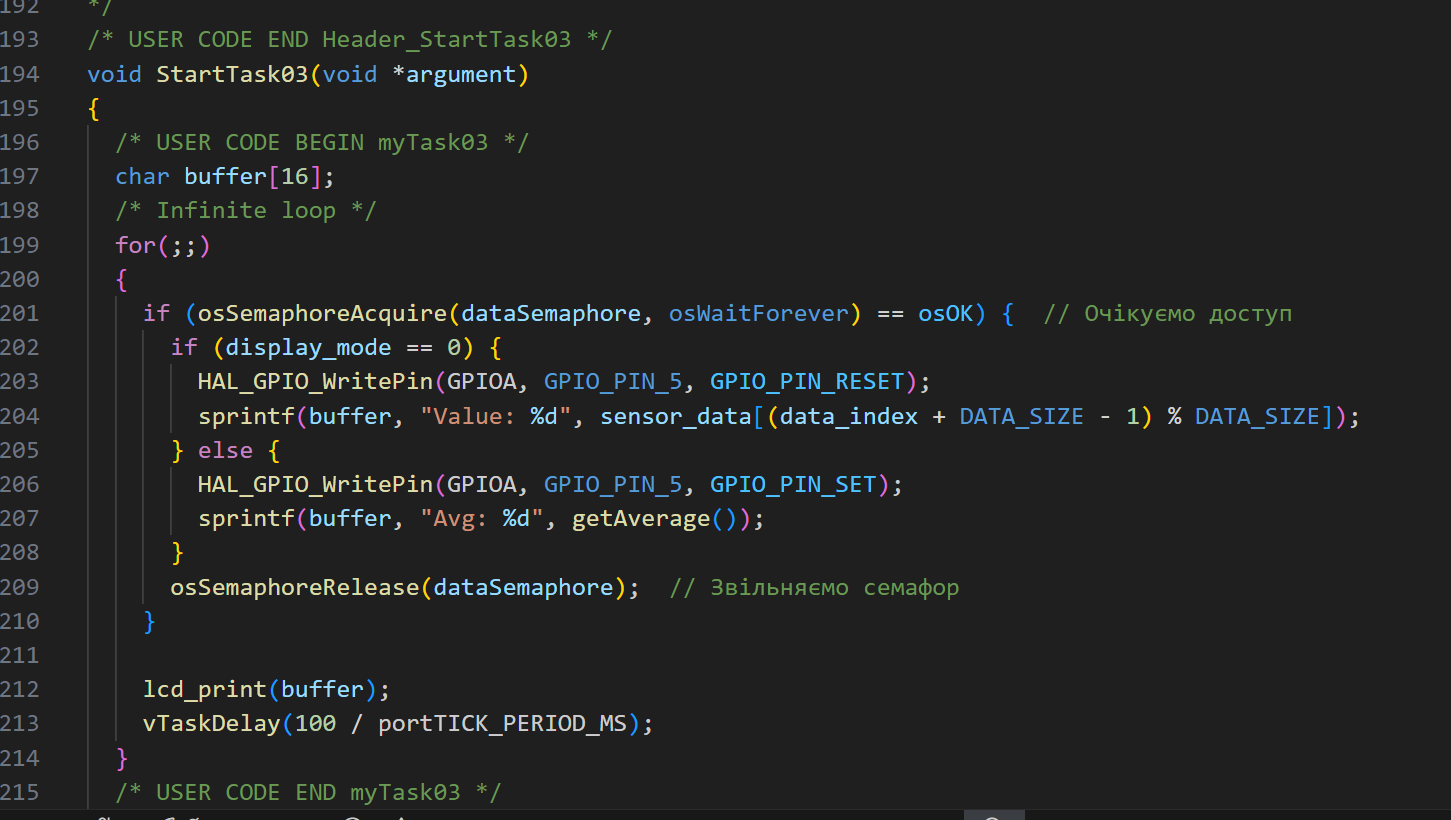


Щосекунди застосунок зчитує поточне значення потенціометра та зберігає його в масив. Це дозволяє вести історію вимірювань, що згодом використовується для обчислення середнього значення.

Якщо режим відображення активний, система працює в стандартному режимі, при цьому жовта лампа блимає, сигналізуючи про оновлення даних. На екрані відображається останнє отримане значення потенціометра. Дані збираються кожну секунду, про що свідчить відповідне щосекундне блимання жовтої лампи.



Якщо користувач натискає кнопку, відбувається перемикання режиму відображення, про що свідчить вимкнення жовтої лампи та ввімкнення зеленої лампи. Замість поточного значення потенціометра, застосунок переходить у режим відображення середнього значення, яке обчислюється на основі всіх збережених вимірювань у масиві. Дані не збираються, поки застосунок не буде повернуто до режиму відображення поточного значення потенціометра.



Таким чином, користувач може відстежувати як миттєве значення потенціометра, так і його середній рівень за певний проміжок часу. Це може бути корисним для аналізу стабільності сигналу або оцінки змін у налаштуваннях потенціометра.

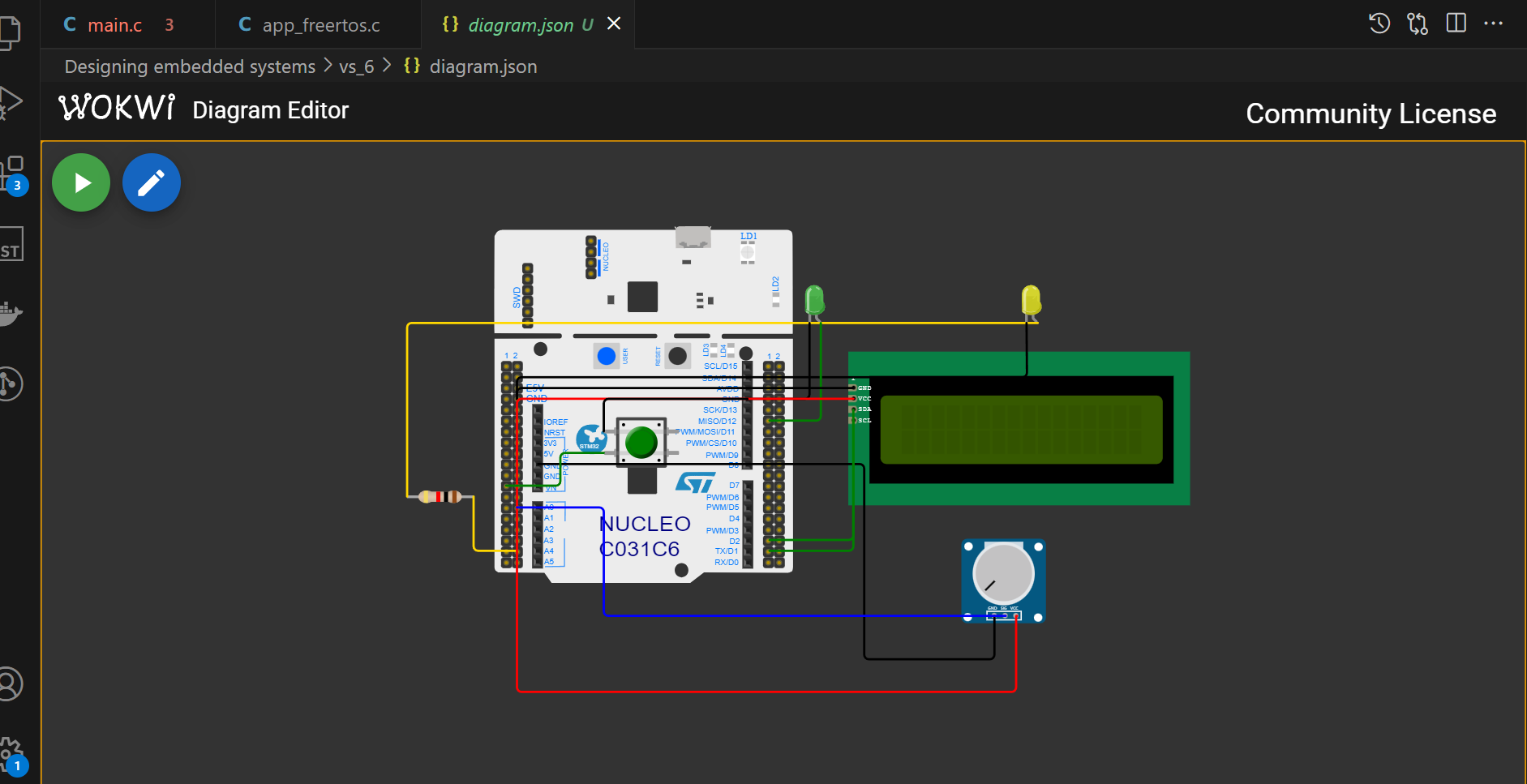
Як видно з коду, у цьому застосунку використано можливості паралельного виконання завдань, що дозволяє мікроконтролеру одночасно зчитувати дані з потенціометра, реагувати на натискання кнопки та оновлювати інформацію на LCD-дисплеї. Завдяки використанню FreeRTOS кожне з цих завдань виконується у власному потоці, що забезпечує ефективний розподіл обчислювальних ресурсів та мінімізацію затримок під час роботи системи.

Для забезпечення коректного доступу до спільних даних між потоками використано механізм семафорів. Це дозволяє уникнути ситуацій, коли одні потоки змінюють дані в той самий момент, коли інші їх зчитують. Наприклад, потік, що відповідає за запис нових значень потенціометра, спочатку отримує блокування на змінення буфера (семафор), а після запису звільняє його, дозволяючи іншим потокам зчитати оновлені дані без помилок.

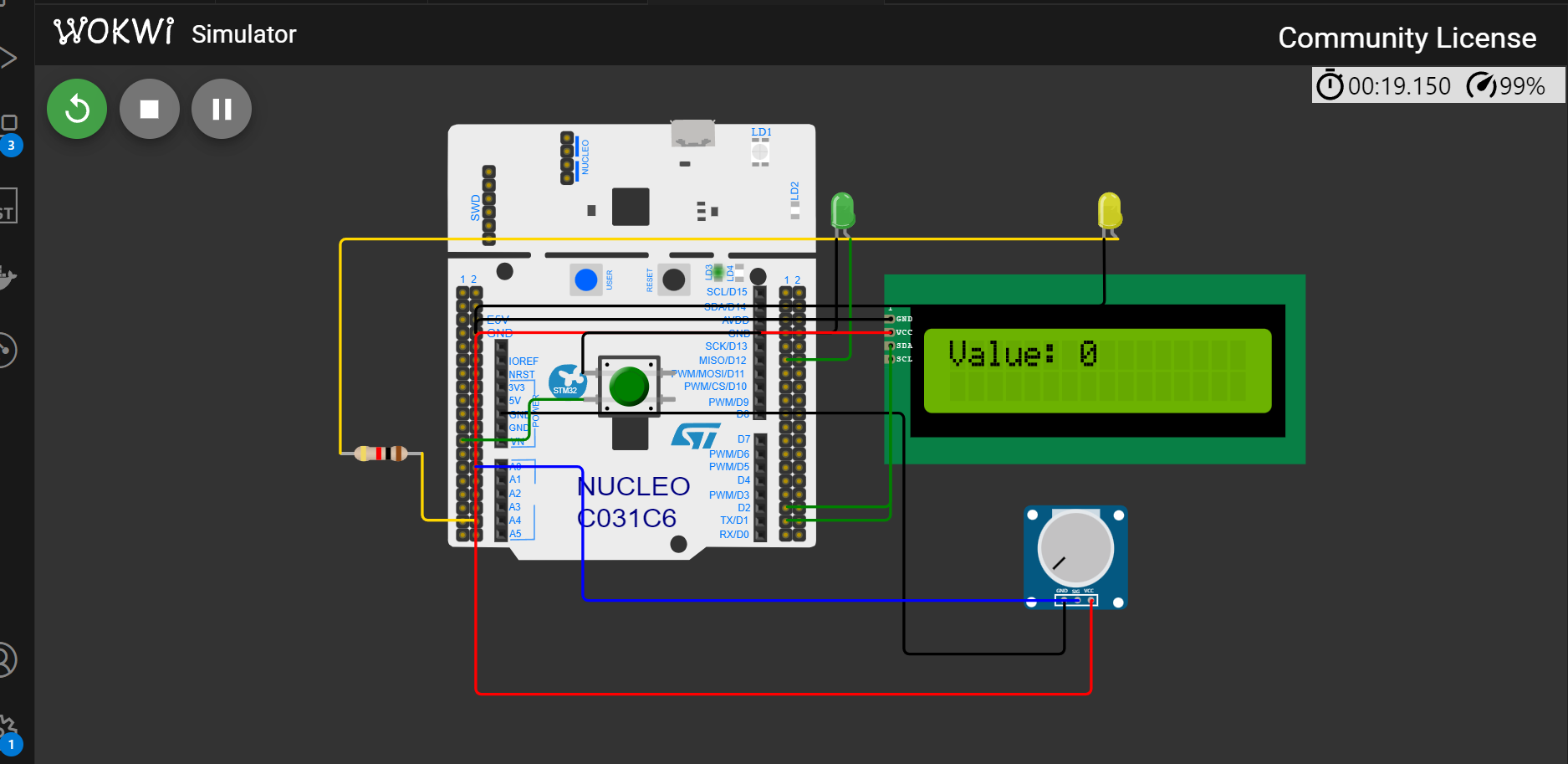
Окрім семафорів, у FreeRTOS існують черги повідомлень, які дозволяють потокам передавати дані між собою, уникаючи зайвого використання спільної пам’яті. У цьому проєкті буфер для збереження вимірювань є простою чергою, яка дозволяє зберігати останні 10 значень потенціометра.

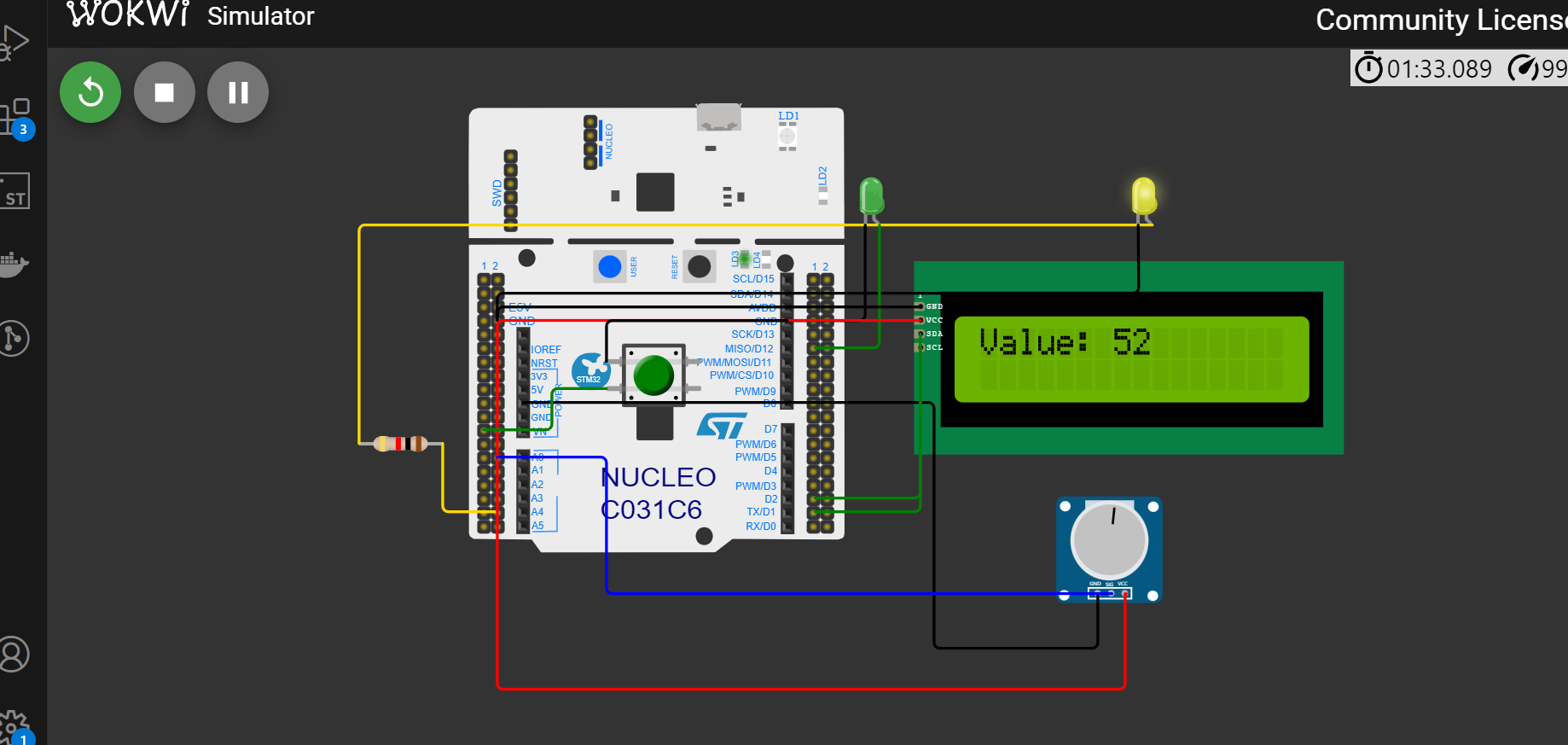
Результат:

Система не запущена:

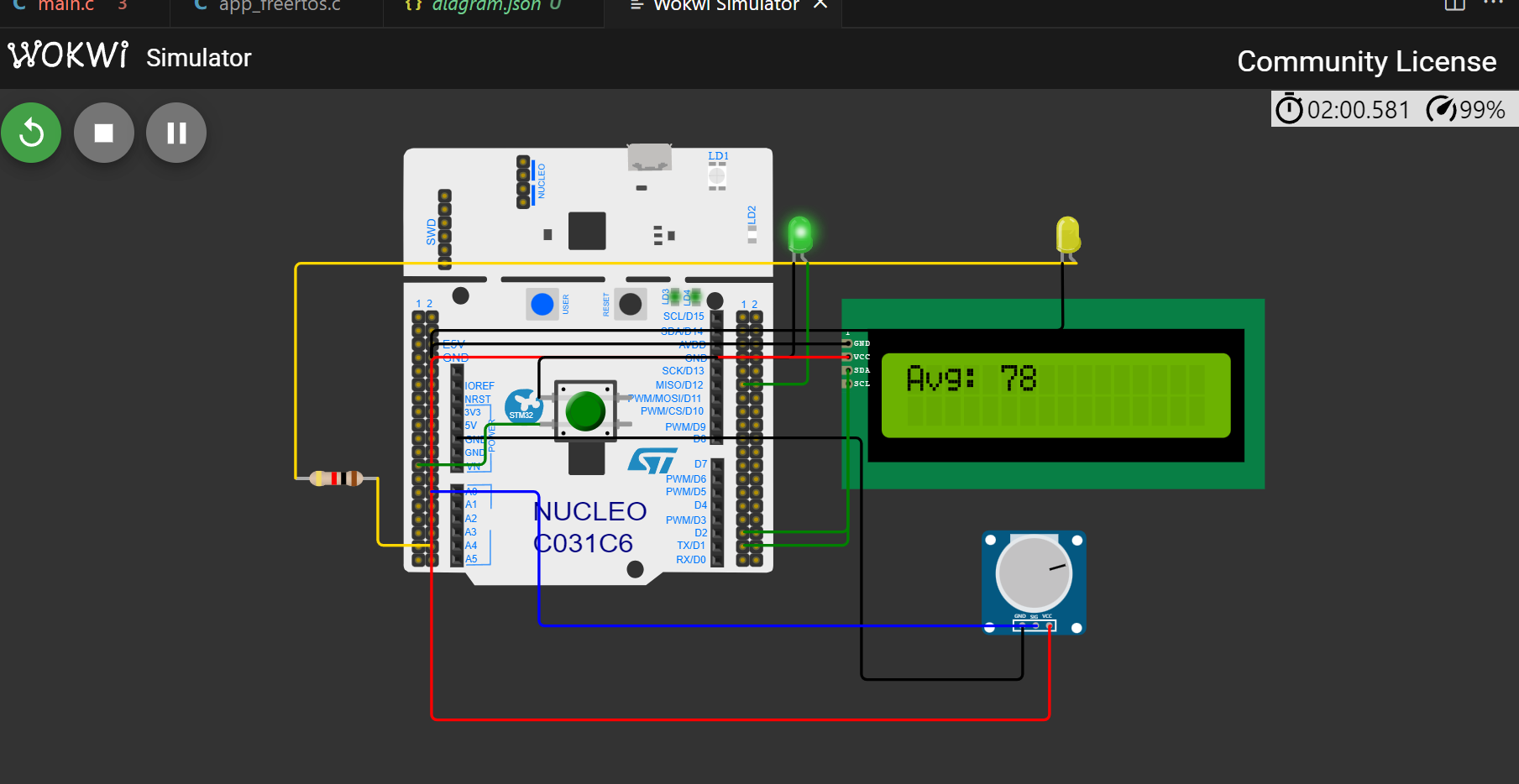


При запуску загорається LCD-дисплей, що виводить поточне значення потенціометра, та починає блимати жовта лампа:

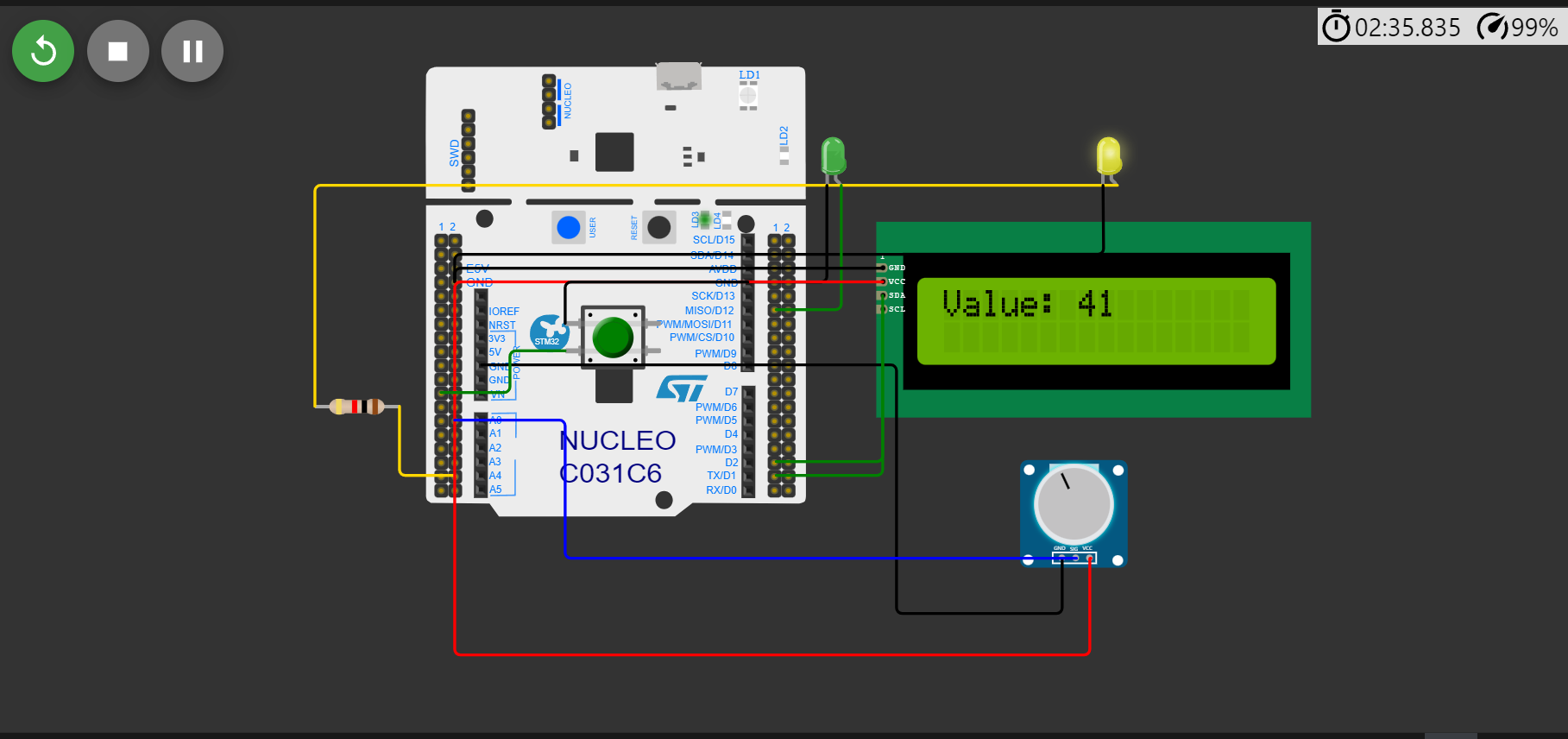




Переключаємо режим на відображення середнього значення:



Повертаємо режим на відображення поточного значенняя потенціометра:



Система працює як заплановано!

**Висновок:** У ході лабораторної роботи було розроблено вбудовану систему, яка дозволяє отримувати та обробляти аналогові дані від потенціометра, використовуючи FreeRTOS. Завдяки багатопоточності мікроконтролер одночасно виконує декілька завдань, що значно підвищує ефективність роботи системи. Реалізація семафорів забезпечує узгоджений доступ до спільних ресурсів, що виключає можливість конфліктів між потоками. Додавання буфера для збереження останніх вимірювань дозволило реалізувати функцію перемикання між миттєвим та середнім значенням потенціометра.