

УДК (UDC) 004.428:004.9:654

**Лупандін
Антоній Володимирович***магістр комп'ютерних наук
Харківський національний університет імені В.Н. Каразіна,
майдан Свободи, 4, Харків, 61022, Україна
e-mail: anlupandin@icloud.com;
<https://orcid.org/0009-0002-7591-5152>***Хрусьов
Максим Михайлович***кандидат фізико-математичних наук, завідувач кафедри комп'ютерних
систем та робототехніки, старший дослідник, доцент
Харківський національний університет імені В.Н. Каразіна,
майдан Свободи, 4, Харків, 61022, Україна
e-mail: maksym.khruslov@karazin.ua;
<https://orcid.org/0000-0001-9639-9340>***Копейкін
Володимир Васильович***старший викладач кафедри комп'ютерних систем та робототехніки
Харківський національний університет імені В.Н. Каразіна,
майдан Свободи, 4, Харків, 61022, Україна
e-mail: volodymyr.kopieikin@karazin.ua; vertexyz@gmail.com;
<https://orcid.org/0009-0009-6845-7211>*

Проектування та перспективи створення модульної GSM-бібліотеки для MicroPython з підтримкою NB-IoT

Актуальність. У контексті зростаючої популярності Інтернету речей (IoT) все більшої актуальності набуває створення програмного забезпечення, сумісного з обмеженими апаратними ресурсами. Технологія NB-IoT забезпечує енергоефективний зв'язок для автономних пристроїв, однак її інтеграція у середовищі MicroPython залишається недостатньо опрацьованою через обмеження наявних GSM-бібліотек. Проблематика ефективної взаємодії мікроконтролерів з GSM-модулями потребує розробки адаптованих інструментів.

Мета. Метою роботи є аналіз сучасних бібліотек для взаємодії з GSM-модулями з акцентом на NB-IoT-проекти та формулювання вимог до нової бібліотеки, оптимізованої для середовища MicroPython.

Методи дослідження. Застосовано порівняльний аналіз функціональних можливостей, архітектури та сумісності п'яти поширених бібліотек: pySerial, GSMComm, Adafruit FONA, TinyGSM, GSM (Arduino). Дослідження здійснювалося за критеріями підтримки AT-команд, енергоефективності, розширюваності, обробки помилок та придатності до MicroPython.

Результати. Встановлено, що жодна з досліджених бібліотек не забезпечує повної підтримки MicroPython-середовища без обмежень. Основні недоліки полягають у відсутності модульності, низькому рівні адаптації до NB-IoT та обмеженій обробці помилок. На основі аналізу сформульовано концепцію нової бібліотеки з модульною архітектурою, адаптованою до MicroPython, з окремими модулями для AT-команд, роботи з мережею, обробки помилок та інтеграції з IoT-протоколами (MQTT, HTTP). Запропоновано структурну модель бібліотеки та UML-діаграми її компонентів.

Висновки. Проведене дослідження підтверджує потребу у створенні нової спеціалізованої GSM-бібліотеки, що поєднувала б гнучкість, енергоефективність і простоту інтеграції з MicroPython для застосування в NB-IoT-пристроях. Запропоновані архітектурні рішення орієнтовані на зниження навантаження на апаратну частину, забезпечення стабільної комунікації та спрощення розробки IoT-проектів.

Ключові слова: GSM, NB-IoT, MicroPython, AT-команди, модульна архітектура, IoT, енергоефективність, pySerial, GSMComm, Adafruit FONA, TinyGSM, Arduino.

Як цитувати: Лупандін А. В., Хрусьов М. М., Копейкін В. В. Перспективи створення спеціалізованої gsm-бібліотеки для micropython у проектах NB-IoT. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2025. вип. 65. С.57-66. <https://doi.org/10.26565/2304-6201-2025-65-05>

How to quote: A. Lupandin, M. Khruslov, V. Kopieikin, "Design and Prospects for Developing a Modular GSM Library for MicroPython with NB-IoT Support", *Bulletin of V. N. Karazin Kharkiv*

National University, series Mathematical modelling. Information technology. Automated control systems, vol. 65, pp. 57-66, 2025. <https://doi.org/10.26565/2304-6201-2025-65-05> [in Ukrainian]

Вступ

У сучасних умовах розробки пристроїв Інтернету речей (IoT) на передній план виходить використання енергоефективних та надійних технологій зв'язку. Однією з таких технологій є NB-IoT (Narrowband IoT), що забезпечує довгострокове з'єднання з низьким енергоспоживанням, що є особливо важливим для автономних пристроїв з обмеженими ресурсами.

Модулі GSM з підтримкою NB-IoT вже набули широкого застосування, однак їх використання з обмеженими ресурсами мікроконтролерів у середовищі MicroPython досі недостатньо опрацьоване. У цій роботі проведено аналіз існуючих інструментів (бібліотек), виявлено їхні недоліки та визначено напрями створення спеціалізованого інструментарію для NB-IoT-комунікації в середовищі MicroPython.

1. Огляд бібліотек

1.1. Бібліотека pySerial

pySerial — це широко поширена бібліотека Python, яка забезпечує послідовний зв'язок між різними платформами, включаючи Windows, Linux і macOS. Незважаючи на те, що pySerial не є специфічним для GSM, він служить базовим рівнем для багатьох бібліотек GSM та IoT, керуючи низькорівневими послідовними взаємодіями, необхідними для надсилання та отримання AT-команд модулям GSM.

Розроблена Крісом Ліхті, pySerial активно розробляється з 2001 року. Вона підтримується за ліцензією з відкритим вихідним кодом, заохочуючи внески спільноти та широке використання як в академічних, так і в комерційних проєктах.

Ця бібліотека сумісна з різними платформами, що забезпечує широке використання. Конфігурація послідовного порту дозволяє детально налаштувати такі параметри, як швидкість передачі даних, парність, стоп-біти та параметри тайм-ауту. Підтримує як блокуючі, так і неблокуючі операції читання/запису. Легко інтегрується з іншими бібліотеками та фреймворками Python, слугуючи серверною частиною для комунікаційних модулів вищого рівня.

pySerial побудовано навколо класу Serial, який інкапсулює всі функції, пов'язані з послідовним зв'язком. Ключові компоненти включають Serial Class, який керує життєвим циклом послідовного порту. Підтримка потоків дозволяє одночасно виконувати операції читання та запису, що може бути корисно для складніших сценаріїв взаємодії. Бібліотека діє як міст зв'язку для надсилання AT-команд і отримання відповідей від модулів GSM.

Серед переваг цієї бібліотеки сумісність із широким спектром послідовних пристроїв, простий API, що скорочує час навчання для розробників, широке тестування, велика база користувачів та документація. Недоліки включають відсутність специфічної підтримки GSM, необхідність додаткової розробки специфічних GSM-функцій.[1]

1.2. Бібліотека GSMComm

GSMComm — це бібліотека .NET, ретельно створена для GSM-зв'язку, яка пропонує повний набір функцій для надсилання SMS, здійснення дзвінків, керування з'єднаннями для передачі даних і керування конфігураціями мережі. Створений для розробників, що працюють в екосистемі .NET, GSMComm оптимізує взаємодію з модулями GSM шляхом абстрагування низькорівневої обробки команд AT.

GSMComm розроблена спільнотою і підтримується на таких платформах, як GitHub, завдяки чому стала популярною серед розробників завдяки надійному функціоналу і простоті інтеграції з програмами на платформі .NET.

Бібліотека полегшує надсилання та отримання SMS-повідомлень. Дозволяє ініціювати та завершувати голосові виклики через модулі GSM. Підтримує встановлення з'єднань для передачі даних і керування ними, включаючи налаштування конфігурацій APN. Надає інструменти для моніторингу потужності сигналу та стану мережі. Використовує події для сповіщення додатків про вхідні повідомлення, стан викликів та інші дії модуля GSM. Реалізує комплексні механізми виявлення помилок і звітування для забезпечення надійної роботи. GSMComm побудовано

навколо об'єктно-орієнтованого дизайну, розділяючи різні функції GSM на окремі класи для кращої організації та зручності обслуговування.

Клас `GsmCommMain` діє як основний інтерфейс для ініціалізації зв'язку, надсилання команд і керування операціями GSM.

Клас `SMS` інкапсулює функції, пов'язані з SMS, зокрема надсилання та отримання повідомлень. Клас `Call` керує операціями голосового виклику, дозволяючи розробникам ініціювати та завершувати виклики програмно. Клас `Network` виконує завдання, пов'язані з мережею, наприклад конфігурацію APN і моніторинг потужності сигналу. `Events` надає обробники подій для реагування на асинхронну діяльність модуля GSM, покращуючи швидкість реагування програм [2].

Ця бібліотека ідеальна для додатків, які потребують масового надсилання SMS-повідомлень і автоматизованої обробки SMS-повідомлень. Підходить для систем, яким потрібно автоматизувати операції голосових дзвінків, наприклад для автоматизованих операторів або систем відстеження дзвінків. `GSMComm` дозволяє додаткам керувати з'єднаннями для даних для модулів GSM із підтримкою Інтернету, що полегшує розгортання IoT. Також надає допомогу в розробці інструментів для моніторингу якості мережі та стану модуля GSM у режимі реального часу.

Щодо переваг, бібліотека `GSMComm` пропонує широкий спектр операцій GSM із коробки, зменшуючи потребу в додаткових впровадженнях. Об'єктно-орієнтований дизайн сприяє розширюваності та зручності обслуговування завдяки структурованій ієрархії класів. Використовує надійність і функції `.NET Framework`, отримуючи переваги від його інструментів і бібліотек. Абстракції високого рівня зменшують складність взаємодії GSM для розробників, підвищуючи продуктивність, завдяки зручному API.

Серед недоліків бібліотеки можна зазначити її обмеженість середовищем `.NET`, що унеможливує використання в інших мовах програмування або платформах, наприклад, `MicroPython`. Крім того, `GSMComm` може бути надмірно складною для простих додатків.»

«Порівняно з поширенішими бібліотеками, такими як `pySerial`, `GSMComm` має меншу спільноту, що обмежує доступність підтримки та додаткових ресурсів. Також важливо враховувати, що бібліотека потребує середовища виконання `.NET`, що часто є неприйнятним для пристроїв з обмеженими ресурсами, таких як мікроконтролери.

1.3. Бібліотека Adafruit FONA

Adafruit FONA Library — це спеціалізована бібліотека, розроблена для серії GSM-модулів Adafruit FONA. Реалізований в Arduino, він забезпечує високорівневі абстракції для загальних функцій GSM, спрощуючи взаємодію з апаратним забезпеченням FONA та дозволяючи швидко розробку проектів з підтримкою GSM.

Розроблена компанією Adafruit Industries, відомим постачальником обладнання та програмного забезпечення з відкритим кодом, бібліотека FONA ретельно задокументована та підтримується численними прикладами та посібниками. Він оптимізований для власних модулів GSM Adafruit, але може бути адаптований для використання з сумісним обладнанням.

Adafruit FONA Library дозволяє легко надсилати та отримувати SMS-повідомлення

Adafruit FONA Library полегшує ініціювання та завершення голосових викликів програмним шляхом. Підтримує встановлення з'єднань даних GPRS для доступу до Інтернету. Надає методи для налаштування параметрів APN і керування параметрами мережі. Містить функції для моніторингу стану батареї та керування живленням модуля GSM. Керує сигналами готовності терміналу даних (DTR) для керування модулем. Забезпечує керування параметрами аудіо, увімкнення та вимкнення функцій мікрофона та динаміка. Вбудовані функції забезпечують функції високого рівня, які абстрагують базові AT-команди, зменшуючи потребу в реалізації команд вручну.

Бібліотека Adafruit FONA побудована на основі Arduino, використовуючи класи та методи для інкапсуляції функцій GSM. Клас `Adafruit_FONA` — центральний клас, який надає методи для SMS, викликів і мережевих операцій. Похідні класи від цього є спеціалізовані класи, які розширюють базову функціональність для підтримки різних версій модуля FONA. Службові функції включає допоміжні функції для аналізу відповідей, керування підключеннями та обробки помилок. Каталог прикладів містить численні приклади ескізів, що демонструють різні випадки використання, допомагаючи розробникам зрозуміти та ефективно використовувати бібліотеку [3].

Бібліотека ідеально підходить для любителів і освітніх проєктів, які потребують можливостей GSM, таких як SMS-сповіщення, віддалені виклики та реєстрація даних. Підходить для створення пристроїв Інтернету речей, яким потрібно спілкуватися через мережі GSM, що забезпечує підключення до Інтернету у віддалених місцях. Сприяє розробці носимих технологій, які використовують GSM для зв'язку та передачі даних. Дозволяє створювати системи віддаленого моніторингу та керування, які покладаються на GSM для передачі даних і отримання команд.

Перевагою може бути те, що Adafruit FONA Library розроблена спеціально для модулів FONA від Adafruit, що забезпечує оптимізовану продуктивність і сумісність, але це має зворотню сторону, про котру буде написано нижче. API високого рівня абстрагує складність взаємодії GSM, дозволяючи розробникам зосередитися на логіці програми, а не на обробці команд низького рівня. Бібліотека пропонує розширену документацію, включаючи докладні посилання на функції, приклади використання та навчальні посібники, що полегшує впровадження. Активні форуми спільноти та канали підтримки покращують досвід розробників, надаючи допомогу та сприяючи спільному розвитку. Плавню інтегрується з середовищем розробки Arduino, використовуючи переваги інтуїтивно зрозумілого інтерфейсу та розгалуженої екосистеми.

Серед недоліків варто зазначити, що бібліотека розроблена виключно для модулів FONA від Adafruit, що зменшує її корисність при роботі з іншим обладнанням GSM і потребує додаткових змін для сумісності. Залежність від Arduino Framework може бути перешкодою для розробників, які використовують інші екосистеми або мови програмування.

Крім того, бібліотека не підтримує інші платформи, зокрема MicroPython, що обмежує її застосування у різноманітних проєктах. Незважаючи на достатність функцій для базових операцій GSM, вона не підтримує розширені функції, такі як інтеграція з MQTT або комплексну обробку помилок, необхідні для складних проєктів IoT.

1.4. Бібліотека TinyGSM

TinyGSM — це легка бібліотека GSM, створена для плат Arduino та ESP8266/ESP32. Вона підтримує широкий спектр модулів GSM і надає такі функції, як SMS, HTTP і MQTT зв'язок, що робить її дуже універсальною для різних додатків IoT. TinyGSM розроблено для ефективної роботи, мінімізуючи накладні витрати на пам'ять і обробку даних, що має вирішальне значення для мікроконтролерів з обмеженими ресурсами.

TinyGSM, розроблена Володимиром Шиманським, стала популярною в спільнотах Arduino та ESP8266/ESP32 завдяки легкій архітектурі та широкій сумісності з GSM-модулями. Вона підтримується на GitHub із активним внеском спільноти, що покращує набір функцій і сумісність.

Бібліотека полегшує надсилання та отримання SMS-повідомлень із підтримкою об'єднаних повідомлень. Вона дозволяє ініціювати та завершувати голосові виклики програмно. Підтримує встановлення з'єднань GPRS для доступу до Інтернету та передачі даних. Дозволяє надсилати HTTP-запити GET і POST, полегшуючи інтеграцію веб-служб. Забезпечує функції клієнта MQTT для публікації та підписки на обмін повідомленнями, необхідних для програм IoT. Містить методи налаштування конфігурації APN і керування параметрами мережі. Пропонує інструменти для моніторингу потужності сигналу та стану мережі. Оптимізовані енерговитрати, підходить для пристроїв IoT з живленням від батареї. Сумісна із широким спектром модулів GSM, включаючи SIM800, SIM900, A6, M95 тощо.

TinyGSM використовує модульну та розширювану архітектуру, розділяючи функціональні можливості на окремі класи та модулі для сприяння зручності обслуговування та масштабованості. Основний клас TinyGsm надає доступ до функцій GSM, слугуючи основним інтерфейсом для розробників. Також є спеціалізовані класи, які обробляють особливості зв'язку для різних модулів GSM, забезпечуючи сумісність і оптимізовану продуктивність. Існують виділені класи, які керують зв'язками HTTP і MQTT, абстрагуючись від базових деталей протоколу.

Бібліотека включає допоміжні функції для аналізу відповідей, керування підключеннями та обробки помилок.

Дозволяє розробникам налаштовувати такі параметри, як швидкість передачі даних, тайм-аути та параметри живлення відповідно до конкретних установок апаратного забезпечення.[4]

Добре підходить для створення пристроїв IoT, які потребують підключення до Інтернету для передачі даних, віддаленого моніторингу та контролю. Віддалене ведення журналу даних дозволяє створювати системи віддаленого журналювання, які надсилають дані датчиків до

хмарних служб через HTTP або MQTT. Спрощує надсилання SMS-сповіщень і сповіщень для систем моніторингу та оповіщення. Підтримуються програми, які вимагають функцій голосового виклику, наприклад віддалена допомога або автоматична обробка викликів. Інтегрує модулі GSM у налаштування розумного будинку для дистанційного керування та обміну даними.

Серед переваг це легкий дизайн оптимізований для мікроконтролерів з обмеженими ресурсами, що забезпечує ефективну продуктивність і низьке використання пам'яті. Модульна архітектура полегшує включення лише необхідних компонентів, підвищуючи гнучкість і оптимізуючи ресурси. Також бібліотека сумісна із широким спектром модулів GSM, що збільшує його застосовність у різноманітних проєктах і налаштуваннях обладнання. Надає API високого рівня, через що має зручні методи роботи GSM, зменшуючи складність впровадження. Отримує переваги від постійних внесків і підтримки з боку активної спільноти, що забезпечує постійні вдосконалення та оновлення. Пропонує велику документацію та численні приклади, допомагаючи розробникам зрозуміти та ефективно використовувати бібліотеку.

Відносно недоліків варто зазначити, що деякі розширені функції або підтримка окремих модулів можуть мати недостатню документацію, що ускладнює впровадження у складних проєктах. Також абстрагування низькорівневих функцій обмежує детальний контроль, що може бути критичним для спеціалізованих застосувань.

Крім того, керування залежностями та конфігураціями для різних GSM-модулів може бути досить складним, особливо в проєктах з кількома типами обладнання. Бібліотека TinyGSM орієнтована переважно на платформи Arduino та ESP8266/ESP32 і має обмежену сумісність із середовищем MicroPython, що значно зменшує її придатність для відповідних проєктів.

1.5. Бібліотека GSM

Бібліотека GSM — це вбудована бібліотека в екосистемі Arduino, що забезпечує основні функції GSM. Вона підходить для звичайних програм, які вимагають простих операцій GSM, таких як надсилання SMS або здійснення дзвінків. Хоча вона не така багата функціями, як спеціалізовані бібліотеки, такі як TinyGSM або Adafruit FONA Library, вона пропонує простий інтерфейс для базової взаємодії GSM.

Бібліотека GSM, що входить до складу Arduino IDE, підтримується спільнотою Arduino. Вона розроблена для бездоганної роботи з Arduino GSM Shield та іншими сумісними GSM-модулями, забезпечуючи легкість інтеграції та розгортання для базових проєктів GSM.

Бібліотека дозволяє надсилати та отримувати SMS-повідомлення з базовими функціями. Облегшує ініціювання та завершення голосових викликів програмним шляхом. Підтримує встановлення з'єднань GPRS для доступу до Інтернету, хоча й з обмеженими можливостями порівняно з більш просунутими бібліотеками. Надає методи налаштування конфігурацій APN і керування параметрами мережі. Містить основні інструменти для моніторингу потужності сигналу та стану мережі. Дозволяє запитувати інформацію про SIM-картку та керувати нею, наприклад перевіряти стан SIM-картки та отримувати номери телефонів. GSM розроблена для роботи з Arduino GSM Shields без коробки, що спрощує процес налаштування для початківців.

Бібліотека GSM використовує процедурну та зрозумілу архітектуру, що відповідає простоті моделі програмування Arduino. Центральний клас – клас GSM, надає методи для операцій GSM, слугуючи основним інтерфейсом для розробників. Бібліотеку оптимізовано для Arduino GSM Shield, що забезпечує безперебійний зв'язок і апаратну сумісність. Містяться допоміжні функції для аналізу відповідей і керування з'єднаннями, хоча й з обмеженою глибиною порівняно з більш спеціалізованими бібліотеками [5].

Ідеально підходить для проєктів, які потребують надсилання та отримання SMS-повідомлень, таких як системи сповіщень або боти для обміну повідомленнями. Підходить для додатків, які потребують автоматизації голосових викликів, наприклад для автоматизованих операторів або систем відстеження викликів. Забезпечує базове підключення до Інтернету для пристроїв IoT, полегшуючи передачу даних і віддалений моніторинг. Служить чудовим навчальним засобом для початківців, які вивчають зв'язок GSM у середовищі Arduino.

Перевагами є простий API який узгоджується з моделлю програмування Arduino, що робить його доступним для початківців і простих проєктів. Швидке розгортання сприяє швидкому створенню прототипів і розгортанню основних функцій GSM без тривалого налаштування. Бібліотека плавно інтегрується з середовищем розробки Arduino, використовуючи переваги його

інтуїтивно зрозумілого інтерфейсу та розгалуженої екосистеми. Розроблено спеціально для Arduino GSM Shields, що забезпечує оптимізовану продуктивність і сумісність.

До недоліків бібліотеки належать обмежений набір функцій, які охоплюють лише базові можливості GSM, без підтримки більш розширених операцій, таких як HTTP чи MQTT. Бібліотека також не підходить для масштабованих або складних IoT-проєктів, що вимагають розширених можливостей.

Крім того, GSM-бібліотека є апаратно-залежною, оптимізованою для роботи з Arduino GSM Shield, що знижує гнучкість її використання з іншим обладнанням. Відсутність повноцінних механізмів обробки помилок і обмежена документація можуть спричинити труднощі в реалізації складніших завдань.

2. Порівняння архітектури бібліотек

Розуміння структурного дизайну бібліотек GSM має вирішальне значення, оскільки це безпосередньо впливає на їх зручність використання, розширюваність і продуктивність. У цьому підрозділі розглядаються архітектурні компоненти розглянутих бібліотек, підкреслюючи подібності та відмінності.

Більшість бібліотек GSM мають основну структуру, що складається з таких компонентів:

- Командний інтерфейс керує надсиланням і отриманням AT-команд через послідовні канали зв'язку. Зазвичай інкапсульована в клас, який обробляє послідовні операції низького рівня.
- Парсер команд інтерпретує відповіді від модуля GSM, перетворюючи їх у дані, які можна виконати, або створюючи відповідні винятки. Реалізовано як частину командного інтерфейсу або як окремий модуль для обробки конкретних командних відповідей.
- Менеджер підключень обслуговує такі пов'язані з мережею функції, як підключення до стільникової мережі, керування налаштуваннями APN і моніторинг якості сигналу. Часто розробляється як окремий клас або інтегрується в основні функції бібліотеки.
- Інтеграції полегшує інтеграцію з сервісами вищого рівня, такими як брокери MQTT, HTTP-сервери або хмарні платформи (наприклад, Blynk). Реалізовано як додаткові модулі або класи, які розширюють основні функції.
- Моделі визначає структури даних, що представляють різні сутності та команди в контексті зв'язку GSM. Часто прості класи даних або структури, які інкапсують параметри команд і відповіді.
- Утиліти надають допоміжні функції для таких завдань, як перетворення даних, керування станом і журналювання. Реалізовано як окремі модулі для сприяння повторному використанню коду та модульності.
- Обробка помилок реалізує спеціальні винятки для ефективного керування та реагування на помилки зв'язку. Визначається як спеціальні класи винятків, які створюються на основі конкретних умов помилки.

Для кращого сприйняття у таблиці 1 представлено порівняльний аналіз основних структурних блоків бібліотек. Приведено перелік бібліотек й основних компонентів. Кожна існуюча бібліотека має власний набір переваг і обмежень, що впливає на їх придатність для різних програм.

Попередньо підсумовуючи можна зазначити, що `pySerial` служить низькорівневою бібліотекою послідовного зв'язку без специфічних для GSM абстракцій. `GSMComm` і `TinyGSM` пропонують комплексні структури з усіма основними компонентами, що робить їх придатними для складних програм.[6]

Бібліотека Adafruit FONA забезпечує спеціалізовану структуру, оптимізовану для конкретного обладнання, з частковими реалізаціями деяких компонентів. GSM (бібліотека Arduino) пропонує спрощену структуру, орієнтовану на основні функції GSM, придатну для простих проєктів.

Таблиця 1. Порівняльний аналіз структурних блоків бібліотек
Table 1. Comparative analysis of library structural blocks

Бібліотека	Інтерфейс команд	Парсер команд	Менеджер з'єднань	Інтеграції	Моделі	Утиліти	Обробка помилок
pySerial	Так	Ні	Ні	Ні	Ні	Мінімальний	Ні
GSMComm	Так	Так	Так	Так	Так	Мінімальний	Так
Adafruit FONA Library	Так	Частково	Частково	Ні	Так	Так	Так
TinyGSM	Так	Так	Так	Так	Так	Так	Так
GSM (Arduino Library)	Так	Частково	Частково	Ні	Ні	Мінімальний	Мінімальний

3. Концепція та структура проєктування спеціалізованої бібліотеки для MicroPython

У цьому розділі буде представлено концепцію майбутньої розробки бібліотеки послідовних команд, оптимізованої для застосування в середовищах MicroPython. Виходячи з результатів попереднього аналізу наявних рішень і виявлених функціональних обмежень, буде сформульовано основні підходи до проєктування, архітектурні принципи, методи реалізації, а також запропоновано сценарії тестування, які дозволять ефективно реалізувати поставлені цілі.

У межах запропонованого підходу бібліотека спиратиметься на принципи об'єктно-орієнтованого програмування: кожен функціональний блок (робота з АТ-командами, обробка помилок, підключення до мережі) буде інкапсульовано у відповідні класи, реалізуючи спадкування та поліморфізм. Це забезпечить уніфікований інтерфейс взаємодії з різними GSM-модулями та спростить підтримку коду.

Модульна архітектура бібліотеки дозволить (рис.1):

- повторно використовувати компоненти в різних проєктах;
- ізолювати зміни, що спростить обслуговування та оновлення;
- безболісно додавати підтримку нових модулів;
- тестувати окремі частини системи незалежно.



Рис. 1 UML-діаграма пакетів
Fig. 1 UML package diagram

Структура коду передбачає розділення на логічні каталоги (рис. 2): ядро, реалізації модулів, утиліти, тести та приклади. Це забезпечить зрозумілу організацію проєкту, спростить навігацію та командну розробку.

З архітектурної точки зору, проєкт передбачатиме чітке розділення базової логіки обробки АТ-команд і управління мережею від реалізацій, специфічних для окремих GSM-модулів. Це забезпечить стабільність, гнучкість і тривалий життєвий цикл рішення.

Бібліотека матиме модульну архітектуру, що забезпечуватиме масштабованість, повторне використання коду та зручну інтеграцію з різними GSM/NB-IoT модулями.



Рис. 2 Структура розробленої бібліотеки

Fig. 2 The structure of the developed library

Основні структурні елементи бібліотеки будуть наступними:

`gsm_nbiot_lib/` — кореневий пакет, який об'єднуватиме всі внутрішні модулі.

`core/` — міститиме реалізацію основної функціональності: обробку АТ-команд (`at_command.py`), парсинг відповідей (`command_parser.py`), керування мережевими з'єднаннями (`connection.py`), обробку помилок (`errors.py`) та абстракції модулів (`base_module.py`).

`module/` — включатиме реалізації для конкретних пристроїв, таких як SIM7020 і SIM7000.

`integration/` — надаватиме засоби інтеграції з платформами Blynk, MQTT і HTTP.

`utils/` — включатиме допоміжні інструменти: системи журналювання, конфігураційні засоби та службові функції.

`models/` — визначатиме структури даних для АТ-команд і відповідей.

`main.py` — слугуватиме демонстраційним скриптом запуску та використання бібліотеки.

`tests/` — включатиме модульні тести для ключових компонентів із використанням фреймворків `pytest` і `unittest`.

`docs/` — зберігатиме документацію, згенеровану засобами Sphinx і доступну в онлайн-форматі.

`examples/` — міститиме практичні сценарії використання функціоналу бібліотеки.

Файли конфігурації проєкту — `README.md`, `pyproject.toml`, `requirements.txt`, `.env`, `LICENSE` та інші файли, які забезпечуватимуть управління проєктом, налаштування середовища та юридичну базу.

Таке розділення функціональностей на ізольовані пакети забезпечить чітке зонування відповідальності кожного модуля, що, у свою чергу, мінімізуватиме внутрішні залежності, знижуватиме ризики каскадних помилок і спрощуватиме налагодження та супровід системи. Передбачатиметься також можливість цільового тестування кожного компонента або зовнішньої інтеграції, що дозволить досягти повного покриття тестами та своєчасного виявлення потенційних проблем.

Під час тестування будуть використовуватися методи імітації апаратної взаємодії, що дасть змогу перевіряти логіку бібліотеки незалежно від фізичного обладнання. Завдяки модульній побудові, бібліотеку можна буде легко доповнювати новими можливостями або підтримкою додаткових GSM-модулів без значних змін в існуючій кодовій базі.

Висновки

Проведене дослідження виявило істотний брак універсальних і водночас оптимізованих рішень для взаємодії з GSM-модулями у середовищі MicroPython, особливо в контексті використання технологій NB-IoT. Аналіз поширених бібліотек (pySerial, GSMComm, Adafruit FONA, TinyGSM, GSM для Arduino) засвідчив, що хоча вони й реалізують базовий GSM-функціонал, їх ефективність на малопотужних мікроконтролерах обмежена апаратною залежністю, відсутністю модульності, обмеженою обробкою помилок або несумісністю з MicroPython.

Це зумовило потребу у створенні нової спеціалізованої бібліотеки для MicroPython, яка відповідає б таким ключовим вимогам:

- повна інтеграція з середовищем MicroPython;
- модульна архітектура для масштабованості та повторного використання;
- розвинені механізми обробки помилок і відновлення зв'язку;
- підтримка сучасних IoT-протоколів (MQTT, HTTP);
- енергоефективність для автономних пристроїв;
- наявність документації та засобів тестування.

Розроблена бібліотека реалізує ці вимоги завдяки чітко структурованій архітектурі, заснованій на принципах об'єктно-орієнтованого програмування та модульності. Її структура охоплює ядро для базової логіки, окремі модулі для специфічних GSM-пристроїв, утиліти, інтеграційні компоненти та моделі для уніфікації даних. Такий підхід забезпечує легкість масштабування, повторне використання коду та зручність супроводу.

Таким чином, запропонована бібліотека закладає основу для формування нового стандарту взаємодії з GSM-модулями у середовищі MicroPython та відкриває перспективи для розробки високонадійних IoT-систем з підтримкою NB-IoT.

СПИСОК ЛІТЕРАТУРИ

1. PySerial 3.0 Documentation [Електронний ресурс] // PySerial : офіційний сайт. URL: <https://pythonhosted.org/pyserial> (дата звернення: 26.03.2025).
2. GSMComm for .NET 2.0 [Електронний ресурс] // Arponic : сайт спільноти розробників. URL: <https://gsmcomm-for-net-2-0.arponic.com> (дата звернення: 28.03.2025).
3. Adafruit FONA Library 1.0 Documentation [Електронний ресурс] // Arduino : документація бібліотек. URL: <https://docs.arduino.cc/libraries/adafruit-fona-library/> (дата звернення: 05.05.2025).
4. TinyGSM Documentation [Електронний ресурс] // Arduino : документація бібліотек. URL: <https://docs.arduino.cc/libraries/tinygsm/> (дата звернення: 15.05.2025).
5. GSM Documentation [Електронний ресурс] // Arduino : документація бібліотек. URL: <https://docs.arduino.cc/libraries/gsm/> (дата звернення: 17.05.2025).
6. MicroPython documentation [Електронний ресурс] // MicroPython : офіційна документація. URL: <https://docs.micropython.org/en/latest/> (дата звернення: 21.05.2025).

REFERENCES

1. PySerial 3.0 Documentation, PySerial, 2024. [Online]. Available: <https://pythonhosted.org/pyserial>. [Accessed: Mar. 26, 2025].

2. GSMComm for .NET 2.0, Apponic, 2024. [Online]. Available: <https://gsmcomm-for-net-2-0.apponic.com>. [Accessed: Mar. 28, 2025].
3. Adafruit FONA Library 1.0 Documentation, Arduino, 2024. [Online]. Available: <https://docs.arduino.cc/libraries/adafruit-fona-library/>. [Accessed: May 5, 2025].
4. TinyGSM Documentation, Arduino, 2024. [Online]. Available: <https://docs.arduino.cc/libraries/tinygsm/>. [Accessed: May 15, 2025].
5. GSM Documentation, Arduino, 2024. [Online]. Available: <https://docs.arduino.cc/libraries/gsm/>. [Accessed: May 17, 2025].
6. MicroPython Documentation, MicroPython, 2024. [Online]. Available: <https://docs.micropython.org/en/latest/>. [Accessed: May 21, 2025].

Lupandin Antonii*Master of Computer Sciences**V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine**e-mail: anlupandin@icloud.com;**<https://orcid.org/0009-0002-7591-5152>***Khruslov Maksym***PhD in Physical and Mathematical Sciences, Head of the Department of Computer Systems and Robotics, Senior Researcher, Associate Professor**V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine**e-mail: maksym.khruslov@karazin.ua;**<https://orcid.org/0000-0001-9639-9340>***Kopieikin Volodymyr***Senior Lecturer at the Department of Computer Systems and Robotics**V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine**e-mail: volodymyr.kopieikin@karazin.ua; vertexyz@gmail.com;**<https://orcid.org/0009-0009-6845-7211>*

Design and Prospects for Developing a Modular GSM Library for MicroPython with NB-IoT Support

Relevance. In the context of the growing popularity of the Internet of Things (IoT), the development of software compatible with limited hardware resources is becoming increasingly relevant. NB-IoT technology provides energy-efficient communication for autonomous devices; however, its integration within the MicroPython environment remains insufficiently addressed due to the limitations of existing GSM libraries. The issue of effective interaction between microcontrollers and GSM modules necessitates the development of adapted tools.

Purpose. The aim of this work is to analyze current libraries for interaction with GSM modules, with an emphasis on NB-IoT projects, and to formulate the requirements for a new library optimized for the MicroPython environment.

Research Methods. A comparative analysis was conducted of the functional capabilities, architecture, and compatibility of five common libraries: pySerial, GSMComm, Adafruit FONA, TinyGSM, and GSM (Arduino). The research was carried out according to criteria such as AT command support, energy efficiency, extensibility, error handling, and suitability for MicroPython.

Results. It was found that none of the analyzed libraries provide full support for the MicroPython environment without limitations. The main drawbacks include the lack of modularity, low adaptability to NB-IoT, and limited error handling. Based on the analysis, the concept of a new library with a modular architecture adapted to MicroPython was formulated, including separate modules for AT commands, network handling, error processing, and integration with IoT protocols (MQTT, HTTP). A structural model of the library and UML diagrams of its components were proposed.

Conclusions. The study confirms the need to create a new specialized GSM library that would combine flexibility, energy efficiency, and ease of integration with MicroPython for use in NB-IoT devices. The proposed architectural solutions aim to reduce the load on hardware, ensure stable communication, and simplify the development of IoT projects.

Keywords: GSM, NB-IoT, MicroPython, AT commands, modular architecture, IoT, energy efficiency, pySerial, GSMComm, Adafruit FONA, TinyGSM, Arduino.