

**П.С. Волковий, аспірант**  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»

## Організація маршрутизації, зберігання та обробки даних з датчиків з використанням IoT-платформи

*Впровадження нових інформаційно-вимірювальних систем на виробництві зазвичай вимагає визначення структур та алгоритмів зберігання даних, отриманих з вимірювальних перетворювачів. При подальшій обробці такої інформації, а також при додатковій її передачі на перетворювачі виконавчих засобів, які здатні керувати процесом виробництва, реагуючи на виявлені дефекти матеріалу, потрібно приділяти увагу оптимізації відповідних структур та алгоритмів, оскільки їх вибір може суттєво впливати на затримки розповсюдження, обробки та використання інформації з вимірювальних перетворювачів. Окрім своєчасної та надійної передачі інформації з вимірювальних перетворювачів на перетворювачі виконавчих засобів, існує ще низка типових вимог до реалізації інформаційно-вимірювальних систем, таких як доступність вимірювальної інформації для аналізу оператором через спеціалізований графічний користувацький інтерфейс та збереження зроблених вимірів. Графічний інтерфейс користувача зазвичай потрібен для конфігурації інформаційно-вимірювальної системи, а також дозволяє аналізувати відхилення вимірів з метою аналізу ефективності виявлення дефектів у контрольованому матеріалі, дає можливість виявляти та усувати у певних випадках складові похибки, які спричинені неоптимальною реалізацією або конфігурацією інформаційно-вимірювальної системи. У роботі проаналізовано використання IoT-сервера для вирішення описаних вище задач, зроблено огляд існуючих IoT-платформ та описано процес адаптації такої платформи на прикладі інтеграції в інформаційно-вимірювальну систему, призначену для безконтактного вимірювання товщини матеріалу.*

**Ключові слова:** інформаційно-вимірювальна система; інтернет-технології; датчики-перетворювачі; накопичення даних; обробка даних; виробничий контроль.

**Постановка проблеми в загальному вигляді.** Процес обробки вимірювальної інформації важливих технологічних параметрів на виробництві містить такі етапи:

1. Отримання вимірювальних даних з використанням необхідної кількості вимірювальних датчиків-перетворювачів;
2. Аналіз вимірювальної інформації та прийняття узагальненого рішення;
3. Виконання певних дій за розробленим алгоритмом залежно від прийнятого рішення.

Пристрої, функціонал яких включає вимірювання значень параметрів матеріалу, їх первинне перетворення в необхідний нормуючий рівень сигналу для подальшої передачі і наступної обробки, можна вказати як датчики-перетворювачі вимірювальної системи. Пристрої, функціонал яких включає виконання певних дій, які впливають на процес виробництва, можна називати виконавчими засобами. Залежно від необхідних умов для вимірювання та контролю, зв'язок між датчиками-перетворювачами та виконавчими засобами, через проміжний блок обробки даних та створення керуючого сигналу, може бути реалізовано різним чином.

Існує декілька підходів реалізації зв'язку між датчиками, блоком обробки інформації та виконавчими засобами залежно від технічного завдання.

У випадках, коли вимірювання та подальший виконавчий вплив системи вимагають затримки між здійсненням виміру та реакцією виконавчого засобу менш ніж 10 мс, зазвичай використовується прямий провідний зв'язок та низькорівневі протоколи зв'язку для забезпечення стабільної затримки.

У випадках, коли допустима затримка є більшою, з'являється можливість централізованої організації системи вимірювання та контролю у вигляді додавання проміжного пристрою для об'єднання, маршрутизації та оптимізації надходження інформації. Такий підхід дозволяє: зберігати інформацію для подальшого аналізу з метою покращення надійності роботи системи; здійснювати складні обчислення на основі даних, отриманих з групи датчиків для прийняття рішення; зменшити складність налаштування системи у випадку наявності багатьох взаємозв'язаних датчиків-перетворювачів та виконавчих засобів, а також полегшити її конфігурацію.

Центральний блок такої системи відповідає за приймання та накопичення даних з датчиків-перетворювачів, їх обробку і передачу виконавчим засобам сигналу керування. Оскільки для вирішення поставленого завдання затримка в одну секунду є допустимою, було обрано саме такий варіант системи. Для зменшення вартості та складності налагодження системи було обрано варіант реалізації зв'язку між

блоком керування, датчиками-перетворювачами та виконавчими засобами за допомогою мережі «Інтернет». Для збільшення надійності зберігання даних та можливості запобігання наслідків збоїв (наприклад, зникнення електроенергії на підприємстві) до системних вимог було додано можливість розгортання бази даних у хмарному середовищі та можливість використання розподілених баз даних. Надалі створений продукт будемо називати IoT-платформою.

Проблемою, вирішення якої описано в цій статті, є реалізація безконтактної системи для зв'язку між виконавчими засобами та датчиками-перетворювачами, з можливістю обробки та візуалізації інформації в реальному часі через мікропроцесорний блок для задачі вимірювання та контролю товщини матеріалу в процесі виробництва.

**Аналіз останніх досліджень та публікацій, на які спирається автор.** Опис типових вимог до обробки даних в інформаційно-вимірювальній системі розглянуто у публікації [1].

Огляд технічних характеристик різних протоколів зв'язку, що використовуються в сфері IoT разом з результатами порівнянь впливу вибору протоколу зв'язку на параметри IoT платформи, вказано у роботі [2]. Більш детальна інформація щодо стандартів та відмінностей окремих протоколів міститься у джерелах [3–9]. У [10] наявна інформація щодо стандартів для безпроводних інформаційно-вимірювальних систем та аналіз застосовності таких систем на виробництві.

**Метою статті** є розробка IoT-платформи (рис. 1), що є частиною інформаційно-вимірювальної системи для моніторингу товщини рулонного матеріалу на виробництві. Функціональність вказаної вище платформи має враховувати: накопичення даних із зареєстрованих датчиків-перетворювачів, контроль виконавчих засобів, обробку та відображення отриманих даних у режимі реального часу.

Для досягнення мети необхідно описати вирішення таких завдань на прикладі застосування IoT-платформи з інформаційно-вимірювальною системою безконтактного вимірювання товщини різних матеріалів:

- дослідити існуючі IoT-протоколи зв'язку між датчиками-перетворювачами / виконавчими засобами, а також IoT-фреймворки;
- обрати програмний стек, який найліпше відповідає поставленому завданню;
- розгорнути створюваний IoT-сервер та створити графічний інтерфейс для візуалізації процесу вимірювання та контролю;
- додати до датчиків-перетворювачів та блоку обробки інформації функцію комунікації з IoT-сервером.

**Викладення основного матеріалу.** Створювана IoT-платформа має відповідати таким вимогам:

1. Надійність збереження даних. Втрата частини вимірів може призвести до хибної оцінки якості продукції за вимірюваним технологічним параметром матеріалу, що є недопустимим;
2. Достатня масштабованість – система має мати можливість обробляти дані одночасно з великої кількості датчиків-перетворювачів, використовуючи обмежену кількість ресурсів;
3. Невелика надлишковість протоколу комунікації. Це дозволить зменшити навантаження на IoT-платформу і на підключені пристрої, що впливає на вартість пристроїв системи та потенціал платформи до масштабування;
4. Захищеність протоколу комунікації – у незареєстрованих пристроїв не має бути доступу до зчитування та запису інформації, оскільки це може призвести до перехоплення процесу контролю за виробництвом;
5. Візуалізація – у користувачів має бути можливість у режимі реального часу отримувати інформацію щодо процесу контролю через web-інтерфейс.

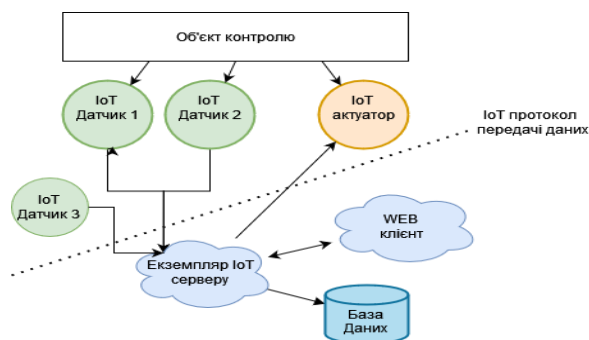


Рис. 1. Структурна схема взаємодії компонентів розроблюваної системи з датчиками та актуаторами (виконавчими засобами)

Насамперед потрібно обрати протокол комунікації між платформою та пристроями. Наразі поширеним серед IoT-пристроїв є використання протоколів HTTP, MQTT, CoAP та AMQP. Розглянемо кожен з них окремо:

**HTTP** є протоколом зв'язку прикладного рівня. Оскільки він заснований на TCP, надійність збереження цілісності даних під час доставки є високою. Також є підтримка TLS. Проте цей протокол не пристосований до взаємодії з виконавчими засобами, бо запити можуть відправлятися лише з клієнта до сервера, що призведе до необхідності постійної відправки запитів керуючим пристроєм. Мінімальний розмір HTTP пакетів – 62 байти. Проте реальний надлишковий розмір зазвичай значно більший.

**MQTT** [3] також використовує за основу протокол TCP. На відміну від HTTP, цей протокол використовує модель «publisher-subscriber», тому він набагато краще підходить для взаємодії з виконавчими засобами. Керуючий пристрій може відправити один запит, який підпише його на отримання визначеної у запиті інформації, яка буде надходити за наявності. Надлишковий розмір MQTT пакета варіюється від 44 до 52 байт залежно від типу пакета. Є підтримка TLS. Варто зазначити, що існують також модифікації MQTT/UDP з меншою надійністю, але і меншим надлишковим розміром пакетів.

**CoAP** [4] є альтернативним протоколом, що поєднує особливості HTTP та MQTT. На відміну від попередніх протоколів він використовує UDP як протокол транспортного рівня, що значно зменшує надлишковий розмір пакетів, який в середньому сягає лише 31 байта. Цей протокол також підтримує модель «publisher-subscriber», подібну до MQTT. З негативних сторін є відсутність механізму гарантованої доставки інформації. Проте існує концепція повідомлень, на які сервер має відправити відповідь з підтвердженням отримання повідомлення. У зв'язку з RESTful архітектурою цього протоколу, CoAP має дещо обмежену швидкість доставки отриманих повідомлень порівняно з MQTT [5].

**AMQP** [6] є одним з найбільш придатних до розширення протоколів. На відміну від MQTT, який підтримує 3 QoS рівні [7], AMQP підтримує лише 2. Разом з моделлю «publisher-subscriber» AMQP також підтримує запити та має велику кількість налаштувань [8], що суттєво збільшує надлишковий розмір пакетів. Як протокол транспортного рівня AMQP використовує TCP.

Оскільки для поставленого завдання потрібна потокова передача даних, з відносно невеликими затримками, було обрано MQTT як протокол, що забезпечує малу надлишковість пакетів, підключення типу «publisher-subscriber», дозволяє додавати нові пристрої, а також дає можливість отримати значну швидкість передачі інформації з датчиків-перетворювачів мікропроцесорному керуючому блоку та виконавчим засобам (актуаторам – засоби, які приводять у рух матеріал та саме положення перетворювачів при його скануванні в процесі виробництва). Використання TCP та підтримка QoS 0 забезпечує захист від втрат пакетів, але є і можливість використання UDP у випадку жорсткіших вимог щодо затримок повідомлень та кількості датчиків разом з перетворювачами та актуаторами, що використовують обмежений інформаційний канал.

**Вибір IoT-фреймворку.** При виборі IoT-фреймворку для розробки додатків та сервісів «Інтернету речей» сучасних інтернет-технологій у вимірювальній техніці необхідно враховувати його компоненти, функціонал та можливість масштабування. IoT-фреймворк – це набір протоколів, інструментів та стандартів, що забезпечують структуру для розробки та розгортання IoT-рішень. Він зазвичай включає апаратне та програмне забезпечення, мережеві елементи (IoT-протоколи), управління пристроями, безпеку, управління даними, розробку додатків та хмарну платформу.

Існують як відкриті (open-source), так і пропрієтарні IoT-фреймворки. Відкриті фреймворки надають доступ до вихідного коду, що дозволяє вносити необхідні модифікації та будувати IoT-додаток з більшим доступом. Пропрієтарні ж фреймворки мають вже встановлену платформу, з якої можна розпочинати роботу.

Серед популярних IoT-фреймворків можна виокремити такі:

1. ThingsBoard:
  - **ліцензія:** FOSS;
  - **реалізація:** серверна частина побудована з використанням Java;
  - **підтримка протоколів:** підтримує широкий спектр протоколів, враховуючи MQTT, CoAP, HTTP, LwM2M, NB IoT, LoRaWAN, SigFox;
  - **внутрішній обмін даними:** підтримує декілька механізмів, таких як RabbitMQ та Apache Kafka;
  - **масштабованість:** забезпечує горизонтальну масштабованість, що дозволяє лінійно збільшувати кількість підтримуваних серверних запитів та пристроїв при додаванні нових серверів у кластерному режимі;
  - **бази даних:** для вбудованого функціоналу зберігання даних підтримує PostgreSQL, а також NoSQL бази даних;
  - **GUI:** дозволяє легко розширювати інтегрований графічний користувацький інтерфейс за допомогою плагінів JS React;
  - **додатковий функціонал:** включає адміністрування користувачів, інтеграцію сповіщень (SMS/Slack), збір та зберігання телеметричних даних, візуалізацію даних, управління пристроями та

активами, систему правил для обробки даних (Rule Engine), управління сигналами тривоги, підтримку великої кількості клієнтів, шифрування пакетів для MQTT з використанням HTTPS, управління обліковими даними пристроїв;

2. Magistrala:

- **ліцензія:** FOSS, продовження проєкту Mainflux;
- **реалізація:** серверна частина побудована з використанням Golang;
- **модульність:** відрізняється значною модульністю серверної частини системи, що мінімізує надмірні витрати при масштабуванні проєкту;
- **особливості:** Magistrala є однією з відкритих платформ, яка дозволяє розгорнути власні IoT-рішення;

3. Хмарні платформи: AWS IoT, Azure IoT, IBM Watson IoT, Google IoT Core. Ці платформи надають комплексні пропріетарні рішення, враховуючи управління пристроями, обробку даних, аналітику та можливості інтеграції.

Також існують інші відомі IoT-фреймворки та бібліотеки, як відкриті, так і пропріетарні, такі як OpenHab, DeviceNive, Eclipse IoT. Частина з них є менш популярною та більш складною в налаштуванні, тоді як інструменти, такі як Eclipse IoT не є фреймворками, і здебільшого потрібні для спрощення реалізації свого фреймворку.

При виборі фреймворку для конкретного проєкту, як у випадку з інформаційно-вимірною системою для контролю товщини рулонного матеріалу, важливо враховувати такі вимоги, як надійність збереження даних, масштабованість, низька надмірність протоколу комунікації, захищеність протоколу та можливості візуалізації даних у реальному часі. Важливою є підтримка MQTT.

Після огляду відповідних характеристик було обрано ThingsBoard, оскільки він дозволяє швидко прототипувати GUI для відображення даних з описаної вище системи контролю і відповідає вище переліченим вимогам.

**Структурний опис інформаційної системи для вимірювання товщини рулонних матеріалів у реальному часі в процесі виробництва.** Огляд IoT-фреймворків було зроблено відносно оптимальності використання їх саме з системою, яка призначена для безконтактного вимірювання товщини рулонних матеріалів (рис. 2, а).

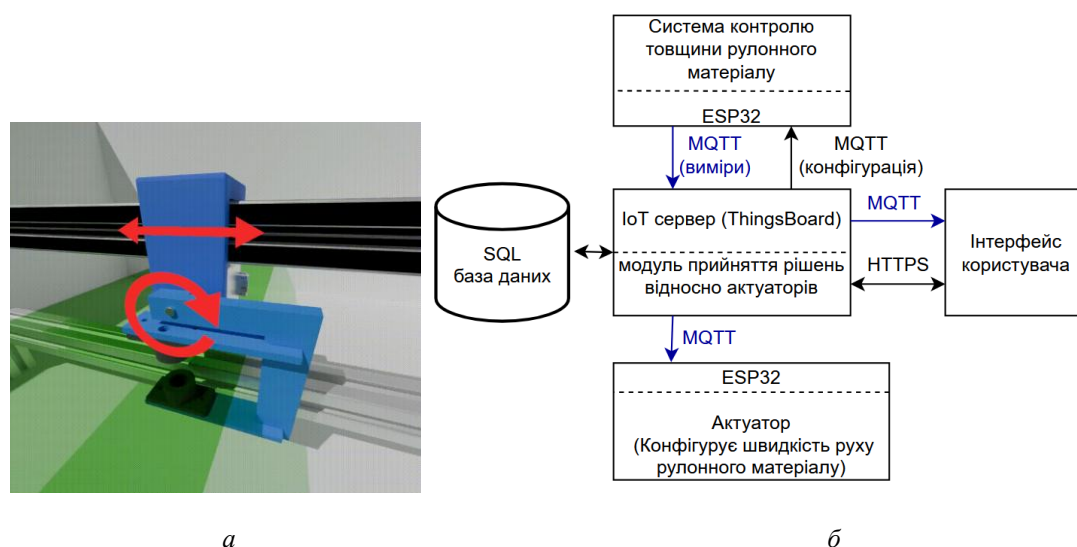


Рис. 2. Система контролю товщини рулонних матеріалів:  
 а) модель розробленого пристрою сканування матеріалу;  
 б) схема передачі вимірної інформації з використанням IoT-сервера

Система, для якої було обрано IoT-фреймворк, має мати можливість проводити вимірювання товщини рулонного матеріалу з різним переміщенням в обох горизонтальних напрямках датчиків відносно самої поверхні полотна. Результати вимірів мають бути достатніми для створення наближеної 3D-моделі поверхні контрольованої плівки. Отримані виміри на виробництві можуть бути використані для зупинки виробничого процесу у разі виявлення суттєвих відхилень, оцінки якості плівки для сортування продукту або пошуку та видалення дефектних частин плівки з рулону. В такому разі зручним інтерфейсом користувача має бути такий, що включає 3D-реконструкцію вигляду плівки за збереженими вимірами та дозволяє швидко розрахувати, на якій частині плівки є суттєві відхилення у товщині.

Для використання системи вимірювання товщини рулонного матеріалу (рис. 2, б), було використано мікросхему ESP32. Завдання даного датчика та системи полягає в агрегації зроблених вимірів (для збільшення пропускної можливості IoT-сервера) та формуванні MQTT-пакетів. Вимірювання товщини відбуваються за визначенням зміни коефіцієнта проходження ультразвукового сигналу крізь матеріал при зміні пікових значень амплітуди хвиль за їх певного кута падіння. Таким чином, для проведення вимірів товщини матеріалу потрібно вимірювати коефіцієнт проходження ультразвукового сигналу під різними кутами падіння хвиль до поверхні полотна. Щоб надати IoT-серверу можливість збирати як дані про вимірювання параметра полотна, так і результуючу інформацію, а також балансувати між затримкою та кількістю надісланих вимірів за секунду, кожен ESP32 підключається з використанням MQTT, через яку можна налаштувати зміст та кількість пакетів даних.

Як інтерфейс користувача було обрано розширення інтерфейсу ThingsBoard плагіном для візуалізації зроблених вимірів. ThingsBoard має власний web-інтерфейс, тому було створено плагін з використанням Three JS для тривимірної репрезентації зроблених вимірів.

**Висновки з дослідження і перспективи подальших досліджень у цьому напрямі.** Використання IoT-платформи дозволило ефективно реалізувати маршрутизацію, зберігання і обробку вимірів з датчиків-перетворювачів. Розглянуті основні протоколи передачі даних між IoT-пристроями та IoT-сервером дозволили зробити вибір та у подальшому інтеграцію IoT-платформи до системи безконтактного вимірювання та контролю товщини рулонних матеріалів. Інтерфейс користувача можна використовувати з можливістю візуалізації зроблених вимірів. Запропонований безконтактний метод вимірювання товщини рулонного матеріалу на базі IoT-сервера дає можливість проводити безперервний моніторинг параметра в процесі виробництва.

#### References:

1. Seenivasan, D. (2022), «ETL for IoT Data: Integrating Sensor Data into Data Warehouses and Real-Time Analytics», *International Journal of Novel Research and Development*, Vol. 7, No. 10, pp. 482–490, [Online], available at: <https://www.ijnrd.org/papers/IJNRD2210185.pdf>
2. Bayılmış, C., Ebleme, M.A., Çavuşoğlu, Ü. et al. (2022), «A survey on communication protocols and performance evaluations for Internet of Things», *Digital Communications and Networks*, Vol. 8, No. 6, pp. 1094–1104, doi: 10.1016/j.dcan.2022.03.013.
3. *MQTT v5 OASIS standard documentation* (2019), official website, [Online], available at: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html>
4. «MQTT and CoAP», *Efento*, [Online], available at: <https://getefento.com/library/mqtt-and-coap-which-protocol-is-better-for-battery-powered-iot-devices/>
5. Çorak, B.H., Okay, F.Y., Güzel, M. et al. (2018), «Comparative Analysis of IoT Communication Protocols», *Proceedings of the International Symposium on Networks, Computers and Communications (ISNCC)*, Rome, Italy, pp. 1–6, doi: 10.1109/ISNCC.2018.8530963.
6. Ashtari, H. (2023), «AMQP vs. MQTT: 9 Key Differences», *Spiceworks*, [Online], available at: <https://www.spiceworks.com/tech/networking/articles/amqp-vs-mqtt/>
7. «MQTT QoS Levels Explained with Examples for Usage», *Bevywise*, [Online], available at: <https://www.bevywise.com/blog/mqtt-qos-level-use>
8. «Choosing the Best Open-Source IoT Platform for Development: A Comprehensive Guide» (2024), *ZedIoT*, [Online], available at: <https://zediot.com/blog/choosing-the-best-open-source-iot-platform-for-development/>
9. Hmissi, F. and Ouni, S. (2022), «A Survey on Application Layer Protocols for IoT Networks», *International Journal on Advances in Telecommunications*, Vol. 15, No. 1, pp. 11–22, [Online], available at: <http://www.iariajournals.org/telecommunications/tocv15n12.html>
10. Raza, M., Aslam, N., Le-Minh, H. et al. (2018), «A critical analysis of research potential, challenges, and future directives in industrial wireless sensor networks», *IEEE Communications Surveys & Tutorials*, Vol. 20, No. 1, pp. 39–95, doi: 10.1109/COMST.2017.2759725.

**Волковий** Павло Сергійович – аспірант Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», розробник програмного забезпечення, GlobalLogic.

<https://orcid.org/0009-0000-6032-9084>.

Наукові інтереси:

- програмування;
- обробка зображень (computer vision);
- безпроводні технології (Wi-Fi);
- приладобудування.

E-mail: [pavel98volk@gmail.com](mailto:pavel98volk@gmail.com).

Volkovyi P.S

**Implementation of routing, storage, and processing of data from sensors and transducers using an IoT platform**

The introduction of new information and measurement systems in production usually requires the creation of structures and algorithms for storing data obtained from sensors and transducers. During the subsequent processing of such information, as well as when it is passed to actuator transducers, capable of influencing the production process based on detected material defects, it is crucial to select optimal data transfer structures and algorithms, since their choice can significantly influence the transmission and processing of information that is generated on sensors and transducers. In addition to minimizing the transmission time and maximizing the transmission reliability, there is typically a number of additional requirements. Those often include the availability of measurement process control and monitoring through a specialized graphical user interface, and long-term storage of performed measurements for future use. The role of graphical user interface is to configure the information and measurement system, and to do a real-time analysis of measured information and the deviations introduced by measurement, so that one can access the material defect detection effectiveness. It also allows us to determine and eliminate the sources of unexpected measurement errors caused by improper measurement system implementation and configuration. This paper analyzes the use of an IoT server to address the abovementioned tasks, provides the comparison between existing IoT platforms, and describes the process of adapting an IoT platform to an example information and measurement system, designed for contactless material thickness measurement.

**Keywords:** information and measuring system; internet technologies; sensors and transducers; data accumulation; data processing; production control.

Стаття надійшла до редакції 28.08.2025.