Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“____” _____ 2026 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax”

Виконав здобувач вищої освіти
IV курсу, групи КІ-22-1
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Артеменюк В.С.
«____» _____ 2026 р.

Керівник проекту
доктор філософії (PhD)
_____ Усік П.С.
«____» _____ 2026 р.
Рецензент _____
_____

м. Кропивницький

Центральноукраїнський національний технічний університет
Факультет *Механіко-технологічний*
Кафедра *Кібербезпеки та програмного забезпечення*
Освітній ступінь *бакалавр*
Галузь знань _12 _____*"Інформаційні технології"*_____
Спеціальність 123 *"Комп'ютерна інженерія"*
Освітньо-професійна (освітньо-наукова) програма _*"Комп'ютерна інженерія"*_

ЗАТВЕРДЖУЮ
Завідувач кафедри
д.т.н., проф.
_____ Олексій СМІРНОВ
« 27 » лютого 2026 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ПЕРШИМ (БАКАЛАВРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Артеменюку Владиславу Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи *Програмне забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax*

2. Керівник роботи *Усік Павло Сергійович, доктор філософії (PhD)*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 133-02 від 27.02.2026 року

3. Строк подання студентом роботи до захисту *23.05.2026 р.*

4. Мета та завдання випускної кваліфікаційної роботи*: Метою роботи є розробка програмного забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax*

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Призначення та область використання.
2.Перегляд аналогічних існуючих систем.
3. Опис і обґрунтування проектних рішень.
4. Етапи програмування системи.
5. Впровадження системи в промислову експлуатацію.
6. Висновки

6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Структурна схема системи	*1 аркуш*
Функціональна схема системи	*1 аркуш*
Діаграма процесів	*1 аркуш*
Блок-схема алгоритму роботи додатку	*2 аркуша*

7. Дата видачі завдання « 27 » лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.03.2026 р.	
2.	Постановка задачі, оформлення ТЗ	15.03.2026 р.	
3.	Розробка моделі компонента	20.03.2026 р.	
4.	Розробка структур даних	25.03.2026 р.	
5.	Розробка алгоритмів зв’язку та відображення	30.03.2026 р.	
6.	Програмування алгоритмів	10.04.2026 р.	
7.	Оформлення ПЗ	17.04.2026 р.	
8.	Попередній захист роботи	23.05.2026 р.	

Дата видачі завдання
« 27 » лютого 2026 р.

Підпис керівника
_____ Усік П.С.
(прізвище та ініціали)

Завдання прийнято до виконання
« 27 » лютого 2026 р.

Підпис здобувача
_____ Артеменюк В.С.
(прізвище та ініціали)

АНОТАЦІЯ

Артеменюк В.С. Програмне забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2026.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

Метою розробки є програмне забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

Результат роботи – програмна реалізація системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, IoT, 802.11ah, 802.11ax

ABSTRACT

Artemeniuk V.S. Software for the implementation of the IoT system using the 802.11ah and 802.11ax standards. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2026.

In this final qualification work for the first (bachelor's) level of higher education, software has been developed that is intended for the implementation of the IoT system using the 802.11ah and 802.11ax standards.

The purpose of the development is the software for the implementation of the IoT system using the 802.11ah and 802.11ax standards.

The result of the work is the software implementation of the IoT system using the 802.11ah and 802.11ax standards.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software are provided.

The program can be used on PCs with Windows 10/11.

The program is developed in the Python environment.

Keywords: computer engineering, IoT, 802.11ah, 802.11ax

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ЕОМ	–	електрона обчислювальна машина
КВ	–	коефіцієнт варіації
КЗ	–	канал зв'язку
НСД	–	несанкціонований доступ
ПС	–	програмна середа
СВВ	–	система виявлення вторгнень
СеМО	–	експонентна мережа масового обслуговування
СМО	–	система масового обслуговування
СПД	–	система передачі даних

ВСТУП

Актуальність теми. Від промислових підприємств до розумних міст, ринок Інтернету речей (IoT) розширюється, і прогнозується, що сукупний річний темп зростання сягне від 10% до понад 20%, причому очікується, що послуги IoT демонструватимуть найбільше зростання. Інтернет речей, здається, є шляхом майбутнього, і зараз може бути гарний час для впровадження нового проекту IoT.

Першим викликом, з яким стикається компанія, яка хоче слідувати останнім тенденціям Інтернету речей, є побудова мережі пристроїв Інтернету речей, які збирають дані та надсилають їх до центральної системи для аналізу. Це особливо складно на об'єктах, що знаходяться на старих ринках, де побудова традиційної комунікаційної мережі з нуля може бути надзвичайно складною.

Отже, як передавати дані з усіх цих пристроїв Інтернету речей, інтелектуальних датчиків та інших компонентів Інтернету речей? Протоколи зв'язку з низькою затримкою, розроблені для роботи з пристроями розумного дому та інтелектуальними пристроями в наших кишенях, не обов'язково ідеально підходять для пристроїв Інтернету речей, що використовуються в периферійних обчисленнях або інших системах Інтернету речей. Wi-Fi має короткий радіус дії і може бути легко заблокований бетонними стінами, резервуарами для води або металевими механізмами. Прокладання кабелів для підключення та живлення всіх пристроїв Інтернету речей може бути дорогим, якщо не зовсім неможливим, у випадках розгортання на занедбаних територіях. Стільникові з'єднання Інтернету речей здаються гарною ідеєю, але вони дорогі, а також можуть бути нестабільними під землею або в складних промислових приміщеннях.

Інтернет Речей – це одна з найбільш популярних сфер для дослідників і розроблювачів футуристів. Але, за станом на 2026 рік, ця сфера вже є не просто уявлюваною концепцією, а вже цілком використовуваною у вузьких колах технологією.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

– Дослідження системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

– Програмна реалізація системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Отже, пояснювати що таке IoT, ми будемо через призму Інтернету Людей. Інтернет Людей – це всесвітня павутина, що дає можливість людям з будь-якої точки землі контактувати один з одним через пристрої.

Ця павутина, так чи інакше змушує нас проводити в ній багато часу, у якій ми часто робимо непотрібні витрати грошей. Все це трапляється завдяки легкості й доступності, адже купити бажані кросовки, колонки або плаття, ми можемо всього в парі кліків.

У роботу Інтернету Речей закладена та ж концепція, тому IoT – це всесвітня павутина, у якій електронні пристрої спілкуються між собою, без втручання людини.

Виходячи з того, що пристроям підключеним до Інтернету Речей не властиві емоції, його по праву можна вважати більше ощадливим і автоматизованим.

1.2 Область застосування

Настільки більша популярність і перспективність даної технології викликана можливістю зміни існуючих правил економіки й ведення бізнесу. Трохи пліч-о-пліч працюючих послуг здатні виключити з бізнес процесів цілий ряд посередників, що зробіть процеси більше дешевими, а значить більше вигідними.

На перший погляд, розмови про взаємодію “машин” можуть здатися занадто футуристичними, але майбутнє не за горами.

Уже зараз, існує більше 20 мільярдів пристроїв підключених до Інтернету Речей, а до 2027 року їхня кількість збільшиться в 2,5 рази. Так, зараз проблема полягає в тому, що ці пристрої не сильно взаємодіють один з одним, але цей “аспект” перебуває в активній фазі розробки.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти

У квітні 2026 року компанія Ericsson оприлюднила результати дослідження, які показали, що телекомунікаційні оператори використовують кілька шляхів для освоєння Інтернету речей (Internet of Things) (IoT) і одержання доходів від нього. До моменту публікації звіту лише деякі учасники ринку мають повноцінні IoT-стратегії.

Відповідно до опитування Ericsson, більше двох третин постачальників телекомунікаційних послуг не мають чіткі плани впровадження технологій Інтернету речей у свою роботу. Вони лише перебувають у пошуках джерел одержання доходів від IoT-рішень, тому тестують різні підходи.

Дослідження Ericsson показало, що телекомунікаційні оператори використовують кілька шляхів для освоєння Інтернету речей (IoT) і одержання доходів від нього.

Ericsson провела дослідження, щоб довідатися, як міжнародні телекомунікаційні компанії підтримують IoT-ініціативи. У доповіді розглянуті 20 провідних операторів. Всі вони згодні з більшим потенціалом Інтернету речей, особливо, коли мова йде про новий IoT-технологій для стільникового зв'язка, начебто Narrow Band IoT (NB-IoT).

Цей стандарт, як відзначають оператори, оптимально підходить для збору, аналізу й керування даними, дистанційного контролю за приладами. Запуск таких мереж дозволить тестувати і впроваджувати в пілотній зоні високотехнологічні продукти й послуги на базі Інтернету речей для розвитку "розумного" міста, у тому числі рішень у сфері безпеки, моніторингу транспорту й екологічної

обстановки, цифровізації різних галузей виробництва, роботи державних установ і ін. За повідомленням Ericsson, у телекомі хочуть заробляти "поза комунікаціями".

Звіт підтверджує важливість IoT для нинішнього й майбутнього бізнесу провідних постачальників послуг, незалежно від того, де вони працюють у світі. Що стосується IoT як нового типу бізнесу, то оператори інвестують у нові технології й створюють нові бізнес-моделі для розподілу доходів і більше широкого використання непрямих каналів. Вони також розробляють нові моделі для надання продуктів як сервіси й онлайн-послуг, а також розвивають інновації разом з партнерами й клієнтами.

В Ericsson розподіляють структуру Інтернету речей між чотирма основними ланками:

– оператори зв'язку, що надають базову інфраструктуру для підключення IoT-сервісів;

– постачальники комунікацій, які запускають IoT-сервіси й управляють ними;

– сервіси-провайдери, що надають сервісні платформи для розроблювачів IoT-систем;

– розроблювачі сервісів, які самі по собі стають постачальниками послуг в області Інтернету речей.

Представники більшості опитаних телекомунікаційних компаній заявили, що вважають ролі операторів зв'язку й постачальників комунікацій основними для росту IoT-виторгу. Додаткові доходи можуть забезпечити або диференціація сервісів через проміжні ролі, або поступальне надання комплексних IoT-рішень замовникам.

Дослідження показало, що телекомунікаційні компанії можуть грати різні ролі в будь-який момент, залежно від варіантів використання й інших факторів.

Експерти Ericsson також прийшли до виводу, що оператори запускають IoT-бізнес у якості окремого стартапа, після чого вони оцінюють успіх проекту на

основі ключових показників діяльності (KPI), таких як кількість завантажень додатків або число задіяних у мережі пристроїв.

Найбільші вигідними сферами застосування Інтернету речей оператори називають керування автопарками, логістику, підключені до інтернету автомобілі, "розумні" міста й промислова автоматизація. Споживчі проекти цікаві в меншому ступені через "неочевидні переваги" і "конкуренції з боку великих гравців".

Інтернет речей у медицині

Сценарії використання

Поява технологій інтернету речей (IoT) привело до захоплюючих досягнень в XXI столітті. IoT тепер використовується не тільки в промисловості, але й в інших галузях. Наприклад, сучасна охорона здоров'я вже важко представити без IoT. У травні 2026 року експерти виділяли наступні варіанти використання IoT.

Обсяг ринку

2026: Розвитку ринку IoT у медицині заважає відсутність стандартів керування

За даними на серпень 2025 року, основними рушійними факторами розвитку ринку інтернету речей в сфері охорони здоров'я є: розробка рішень IoT для зниження вартості медичної допомоги, розвиток технологій штучного інтелекту, збільшення обсягу інвестицій у медичні IoT-рішення, підвищення ефективності мережних технологій, що росте проникнення підключених пристроїв у сфері охорони здоров'я, а також потенціал країн, що розвиваються. Такі дані в середині серпня 2025 року опублікувала компанія Market Research Engine у звіті про результати свого дослідження.

До стримуючий ріст ринку факторам аналітики віднесли недолік навичок розгортання рішень на основі технологій інтернету речей, проблеми сумісності й безпеки, а також відсутність стандартів керування.

У цілому, за даними на серпень, спектр застосування IoT у медицині широкий: від дистанційного спостереження за пацієнтами до інтелектуальних датчиків і медичних гаджетів, таких як фітнес-браслети й мініатюрні пристрої, призначені для діагностики й лікування. На думку аналітиків, кількість підключених пристроїв і величезна кількість інформації, що вони збирають, можуть стати сьогоденням викликом для IT-відділів лікарень. Складність представляють типові для IoT проблеми: риск витікання даних, а також пошук ефективного методу обробки даних.

Ключовими гравцями сегмента в Market Research Engine назвали Medtronic PLC (США), Royal Philips(Нідерланди), Cisco Systems (США), IBM Corporation (США), GE Healthcare (США), Microsoft (США), SAP SE(Німеччина), Qualcomm Life (США), Honeywell Life Care Solutions (США) і Stanley Healthcare (США).

Згідно із прогнозом аналітиків, обсяг ринку інтернету речей у медицині до 2027 року перевищить порядку $158 млрд. Середній показник росту ринку (CAGR) у період з 2016 по 2027 роки експерти Market Research Engine оцінили в 30,8%.[2]

2026: Витрати в $41,22 млрд – ResearchAndMarkets

В 2025 році обсяг витрат на системи, програмне забезпечення, сервіси й медичне устаткування для інтернету речей у сфері охорони здоров'я склали $41,22 млрд, повідомили в дослідницькій компанії ResearchAndMarkets.

За прогнозами експертів, розглянутий ринок буде рости приблизно на 28,9% у рік і досягне $405,65 млрд до 2027 року.

Найбільше швидкоростучим сегментом IoT-ринку в сфері охорони здоров'я в ResearchAndMarkets називають системи й софт, завдяки якому забезпечується високий рівень безпеки даних і візуальне дослідження.

До числа факторів, що сприяють підвищенню медичних IoT-витрат у глобальному масштабі, експерти відносять зростаюче число хронічних

захворювань, введення сприятливі ініціативи урядами в різних країнах і еволюція технологій штучного інтелекту.

Головними бар'єрами на шляху розвитку ринку є слабка ефективність впровадження IoT-рішень, проблеми конфіденційності й безпеки даних, а також недостатньо високі технічні знання на ринку в цілому.

По темпах росту з географічної точки зору лідирує Азіатсько-тихоокеанський регіон за рахунок зростаючого числа лікарень і хірургічних центрів на цьому ринку.[3]

У дослідженні говориться, що Інтернет речей в охороні здоров'я застосовується в таких завданнях, як клінічна робота, оптимізація робочих процесів, мережна візуалізація, телемедицина, керування лікарськими засобами й контроль за стаціонарними пацієнтами

Що стосується найбільших виробників продуктів Інтернету речей для потреб охорони здоров'я, те тут дослідники перераховують наступні компанії:

- Philips;
- Medtronic;
- IBM;
- Accenture;
- Cisco;
- GE Healthcare;
- Microsoft;
- Qualcomm;
- NEC;
- SAP;
- Honeywell Life Care Solutions;
- NXP Semiconductors;
- Stanley Healthcare;
- Hitachi;
- Bosch Software Innovations;

– Amazon.com.

Проект GE Healthcare: датчики на пересувній апаратурі

Відомі виробники медичного устаткування теж застосовують технології Інтернету Речей (IoT) для бізнес-консультування лікарень і клінік. Приміром, компанія GE Healthcare Japan з недавнього часу стала встановлювати датчики на пересувній апаратурі, що переміщається з кабінету в кабінет, такий, як апарати УЗД. Аналіз даних про місце розташування дозволяє знайти оптимальне місце для розміщення устаткування, підвищити ефективність його використання й скоротити кількість УЗД-апаратів, необхідних клініці.

Розробка Hitachi: контроль переміщення персоналу

Як стало відомо в червні 2025 року, японський промисловий конгломерат Hitachi тестує у себе на батьківщині систему Інтернету речей (Internet of Things, IoT), за допомогою якої медичні установи зможуть поліпшити ефективність своєї роботи.

Після випробувань, що проходять у декількох японських лікарнях, на основі IoT-технології Hitachi буде створений консалтинговий сервіс, що планується пропонувати лікарням при місцевих університетах і іншим великим медичним організаціям. У рамках тестування Hitachi оснастила територію лікарень антенами, що стежать за пересуваннями медичного персоналу за допомогою спеціальних сенсорів, закріплених на уніформі лікарів і медсестер. Надалі, проаналізувавши зібрані дані, Hitachi запропонує установі індивідуальні рекомендації з оптимізації роботи.

Керуючись ними, лікарня зможе переглянути графіка робочих змін і скоротити часові витрати за рахунок виявлення непотрібних завдань, виконуваних персоналом. Такі рекомендації дозволять підвищити продуктивність праці в установі, переконані в Hitachi. Сенсори, що носяться також співробітниками, визначають відстань між ними, що дає можливість оцінити, чи досить персонал спілкується між собою.

В Hitachi вважають перспективним консалтинговий бізнес в області медицини й прогнозують, що в 2026 році виторг компанії на цьому напрямку досягне 2 млрд єн (18,1 млн доларів).[4] **Industrial Internet of Things – IIoT**

Промисловий інтернет речей поступово набирає оберти разом з розвитком інтернету речей. Обидві ці концепції припускають передачу даних через інтернет, використовують загальні апаратні платформи й управляються за допомогою спеціалізованого програмного забезпечення, що в остаточному підсумку приводить до росту кількості загальних уразливостей і можливих атак на промислові об'єкти. Зі звіту Frost & Sullivan треба, що промислова й IT-інфраструктури стають прозоріше. Це, насамперед, пов'язане з розвитком стандарту Industrial 4.0 і виключенням повної ізоляції промислових об'єктів, що, звичайно, спричиняє загальні уразливості, використання сервісів безпеки по моделі SaaS для промислових об'єктів, а також використання апаратних пристроїв, доступ до яких потенційний зловмисник може одержати досить легко.

Аналітики відзначають, що ринок послуг промислової кібербезпеки перебуває на піку свого життєвого циклу. Це характеризується ростом поінформованості про правила поведінки серед кінцевих користувачів у зв'язку зі зростаючою потребою в навичках забезпечення кібербезпеки. Що, втім, не рятує компанії від високого ризику атак на промислові системи керування.

Промисловий інтернет речей приносить не тільки гарні прибутки, але й ризики

Багато кінцевих користувачів використовують трудомісткі методи забезпечення безпеки й не мають строгих політик. Постачальники послуг повинні допомогти в автоматизації процесу забезпечення кібербезпеки й запропонувати більше цілісний підхід. Він може полягати, наприклад, у консолідованому погляді на IT- і OT-інфраструктуру.

У звіті Frost & Sullivan вказуються кілька рекомендацій для компаній, які хочуть рости на ринку послуг забезпечення кібербезпеки. Серед них – розробка інтегрованих платформ, що забезпечують високий рівень безпеки кінцевих

користувачів, паралельне впровадження кращих практик забезпечення ІБ, використання автоматизованих сервісів керування й розширеної аналітики для розробки комплексного портфеля послуг, що може бути адаптований для всіх типів кінцевих користувачів. Крім того, аналітики вважають перспективними гнучкі моделі ціноутворення й підхід CSaaS (as-a-Service – «кібербезпека як послуга»)[2].

Ріст Індустріального Інтернету до 2027 року

ABI Research

Аналітики ABI Research прогнозують, що загальний дохід від використання промислових і виробничих IoT-додатків, для роботи яких застосовується мобільний і супутниковий зв'язок, може скласти більш ніж $138 млн. В 2025 році кількість нових провідних і бездротових IIoT-з'єднань збільшиться більш ніж на 13 млн, склавши під кінець року 66 млн з'єднань (разом з 53 млн з'єднань, зафіксованих в 2015 році).

Азіатсько-тихоокеанський регіон займає лідируючі позиції по новим IIoT-підключеннях. В 2026 році тут очікується більше п'яти мільйонів додаткових підключень. Глобальні можливості ринку будуть рости протягом наступних чотирьох років: прогнозується 18 млн нових IIoT-з'єднань по досягненні 2027 року. Незважаючи на це, очікується зниження доходів від послуг зв'язку, що знизиться до $122 млн в 2027 році.

Основна база встановлених з'єднань доводиться на фіксований зв'язок (DSL, кабель, Ethernet і PSTN). На бездротові з'єднання в 2026 році буде доводитися близько 25% нових IIoT-з'єднань. Оператори мобільних мереж продовжать перенос своїх мереж і устаткування для надання IoT-сервісів від технології 2G в 4G LTE діапазон. Підключення в промисловому виробництві також збільшують використання мереж далекого радіуса дії з низьким енергоспоживанням (LPWA). На LPWA у плині наступних чотирьох років прийде найбільше підключень.

Accenture

На думку аналітиків компанії Accenture, до 2030 року обсяг ринку промислового Інтернету речей перевищить оцінку в $14 трлн. Цифри гарні, якщо їсти необхідність залучити інвесторів або переконати начальників зайнятися впровадженням IoT. Не коштує до цих цифр ставитися занадто серйозно. З погляду прогнозу зовсім очевидно, що індустрія росте дуже швидкими темпами.

Ріст очікує кілька стрибків, пов'язаних із прийняттям загальних стандартів і параметрів протоколів, які повинні істотно знизити ризики великих інвестицій у системи IoT.

Прогноз Ovum

Ключовим драйвером росту стане триваюче зниження вартості сенсорів і устаткування, послуг зв'язку, обробки даних і системної інтеграції, з одного боку, і зниження витрат і підвищення виторгу підприємств, які впроваджують інноваційні рішення, з іншої сторони.

Honeywell IIoT: Більшість виробників планують збільшити інвестиції в технології аналізу даних

В опитуванні за назвою «Вплив великих даних на виробництво: вивчення думки керівників» взяли участь більше 200 керівників північноамериканських виробничих компаній. Опитування проводилося підрозділом «Промислова автоматизація» компанії Honeywell разом з корпорацією **KRC Research** з 23 травня по 8 червня 2025 року.

Проведене опитування дозволило зробити наступні основні виводи:

– Деякі компанії змушені працювати під погрозою незапланованих простоїв і поломок устаткування, які розглядаються як найбільш пагубні фактори, що впливають на зростання доходів.

– Більшість компаній визнають, що вже вкладають кошти в технології аналізу даних.

– Більше чверті керівників повідомили, що не планують вкладати кошти в аналіз даних у наступному році; як причини в цій групі найбільше часто згадуються нерозуміння переваг аналізу даних і недостача ресурсів.

Незаплановані простої вважаються основною погрозою для зростання доходів, але 42 % респондентів зізналися, що експлуатують устаткування з більшим навантаженням, чим треба. На питання про те, як часто їхні компанії зіштовхувалися з виробничими проблемами за останні роки, 71 % респондентів відповіли, що відмови устаткування в них виникають час від часу, 64 % сказали те ж саме про незаплановані простої.

40 % респондентів указали незаплановані простої як найбільшу погрозу для зростання доходів. Серед інших причин були зазначені наступні:

– Проблеми керування ланцюжками поставок (39 %).

– Неадекватна кваліфікація персоналу (37 %).

– Брак (36 %).

– Поломки устаткування (32 %).

Аналіз даних як ефективне рішення

Аналіз даних – це ключовий компонент успішного впровадження технологій IIoT для виробників. Більшість респондентів визнають ефективність технологій для аналізу даних. Наприклад, керівники заявили, що згодно з тим, що аналіз великих даних може привести до скорочення:

– поломок устаткування (70 %);

– незапланованих простоїв (68 %);

– позапланового техобслуговування (64 %);

– збоїв у керуванні ланцюжками поставок (60 %).

Респонденти заявили, що, на їхню думку, аналіз даних може допомогти швидше приймати обґрунтовані рішення (63 %), зменшити обсяг шлюбу (57 %) і прогнозувати ризики простою (56 %).

Крім того, більш ніж дві третини респондентів (68 %) заявили, що вони в цей час вкладають кошти в аналіз даних, а 50 % повідомили, що, на їхню думку,

очолювані ними компанії перебувають на правильному шляху відносно впровадження технологій аналізу даних; а 15 % заявили вони перебувають випереджають інші в плані застосування технологій аналізу даних.

Переваги очевидні не для всіх

У той час як більшість респондентів заявили, що вже інвестують і/або планують збільшити інвестиції в аналіз даних у наступному році, 32 % сказали, що в цей час не вкладають кошти в аналіз даних. Тим часом 33 % опитаних заявили, що їхньої компанії не планують інвестувати в аналіз даних протягом наступних 12 місяців або що їм невідомо про такі плани компанії.

Із числа респондентів, які в цей час не планують інвестиції в аналіз даних:

– 61 % вважають, що їхньої компанії вже мають у своєму розпорядженні системи, здатними забезпечити безпека, прибутковість і успіх бізнесу;

– 45 % заявили, що в їхній компанії спостерігається певний ріст і без аналізу даних;

– 42 % сказали, що не розуміють повною мірою переваги великих даних;

– 35 % вважають, що люди переоцінюють переваги великих даних;

– 63 % респондентів, які не планують інвестувати в аналіз даних, указали, що їм просто не вистачає ресурсів для інвестицій, у той час як 39 % повідомили, що не мають у своєму розпорядженні потрібні кадри для необхідного аналізу даних.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – це потужна мова програмування, яка проста у вивченні. Він має ефективні структури даних високого рівня та простий, але ефективний підхід до об’єктно-орієнтованого програмування. Елегантний синтаксис і динамічна типізація Python разом з його інтерпретованим характером роблять його ідеальною мовою для створення сценаріїв і швидкої розробки додатків у багатьох

сферах на більшості платформ. Інтерпретатор Python і обширна стандартна бібліотека доступні у вихідному або двійковому вигляді для всіх основних платформ на веб-сайті Python https://www.python.org/ і можуть вільно поширюватися. Цей же сайт також містить дистрибутиви та вказівники на багато безкоштовних сторонніх модулів Python, програм і інструментів, а також додаткову документацію.

Інтерпретатор Python легко розширюється за допомогою нових функцій і типів даних, реалізованих у C або C++ (або інших мовах, які можна викликати з C). Python також підходить як мова розширення для налаштовуваних програм.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускну кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

В процесі розробки випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформувати висновки про виконаний обсяг робіт та одержані результати.

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Донедавна кожна чергова версія стандарту Wi-Fi була покликана замінити попередню. Однак завдання, розв'язувані з його допомогою, настільки різні, що підхід «один стандарт на всі випадки життя» не відповідає пропонованим вимогам. Це відбивається в розробці специфічних стандартів для різних застосувань. І хоча швидкість як і раніше має значення, на перше місце виходять інші фактори: ємність, вартість, керованість, підтримка різних додатків – від низькошвидкістних підключень IoT до високошвидкісного потокового відео 4K.

Двадцять років тому, в 1997 році, з'явився перший стандарт на Wi-Fi, що забезпечував скромні як на теперішній час 1-2 Мбіт/с. З тих пор підтримувана швидкість зросла на три порядки й тепер становить трохи гігабітів у секунду. А в розроблювальних стандартах пропускну здатність Wi-Fi планується збільшити до декількох десятків гігабіт. Так, стандарт 802.11ay обіцяє більше 20 Гбіт/с і, можливо, навіть понад 100 Гбіт/с у діапазоні 60 ГГц.

Протягом усього свого шляху розвитку Wi-Fi програвав провідному Ethernet у швидкості: все-таки радіохвилі – не краще середовище для передачі. Однак цей відносний недолік виявився набагато менш важливим у порівнянні з таким достоїнством, як мобільність: Wi-Fi став кращим способом доступу в Мережу. Як повідомляє Едгар Фігуєра, глава Wi-Fi Alliance, ще в 2015 році через Wi-Fi було передано більше 50% інтернет-трафіку, а по даним Cisco, користувачі Wi-Fi і абоненти стільникового зв'язка генерують понад дві третини всього мережного трафіку.

Без Wi-Fi як без рук. Цей спосіб передачі даних став культурним явищем: наявність якісного бездротового доступу тепер настільки ж нагальна потреба, як вода в крані або світло в розетці. Так, при виборі готелю більшість постояльців

звертають увагу на доступність Wi-Fi, і, відповідно до недавнього дослідження 2Europe Ltd, проведеному за замовленням компанії ZyXEL, його низька якість є другий після зовнішнього шуму причиною скарг. Втім, це не дивно, адже 38% європейських готелів не можуть забезпечити стабільний доступ в Інтернет через те, що мережа не справляється з більшою кількістю підключених пристроїв.

І хоча швидкість як і раніше має значення, при щільності, що збільшується, використання Wi-Fi, особливо в громадських місцях, і зростаючій розмаїтості застосувань на перше місце в розробці стандартів виходять інші фактори: ємність, вартість, керованість, підтримка різних додатків – від низькошвидкістних підключень IoT до високошвидкісного потокового відео 4K. Потреба задовольнити різноманітні запити знаходить висвітлення в нових стандартах. І звичайно, підтримка більше високих швидкостей ніколи не буде зайвою.

Поточний рівень досягнень в області бездротових локальних мереж характеризується стандартом 802.11ac Wave 2 – відповідне устаткування Wi-Fi Alliance почав сертифікувати торік. У порівнянні з Wave 1 швидкість передачі даних на фізичному рівні зросла з 1,3 до 2,3 Гбіт/с у діапазоні 5 ГГц (хоча це все ще нижче специфікованої межі в 6,9 Гбіт/с у стандарті IEEE на 802.11ac).

Крім підвищення швидкості, була збільшена максимальна можлива ширина каналу – з 80 до 160 ГГц. Теоретично це допоможе полегшити передачу більших файлів. Крім того, додана підтримка четвертого просторового каналу, що теж повинне сприяти підвищенню продуктивності. Розширено спектр робочих частот у діапазоні 5 ГГц, тому доступних каналів стало більше. Таким чином, пристрою будуть менше створювати перешкод один одному.

Мабуть, головною відмінністю другої хвилі від першої є підтримка багатокористувальницького множинного уведення-виводу (Multi-User Multi-In Multi-Out, MU-MIMO): на кожний пристрій може направлятися окремий просторовий потік. Інакше кажучи, точка доступу тепер здатна здійснювати передачу даних на кілька пристроїв одночасно (до чотирьох клієнтських

пристроїв 1?1:1 MIMO у випадку точки доступу 4?4:4 MIMO), що підвищує ефективність використання спектра. Однак MU-MIMO в 802.11ac підтримує трафік тільки від точки доступу до клієнтів, можливість передачі у зворотному напрямку буде реалізована в стандарті 802.11ax.

Наприкінці минулого року з'явилися перші пристрої стандарту 802.11ad, більше відомого як WiGig. У цьому випадку теоретичний максимум пропускної здатності становить 8 Гбіт/с (перші випущені пристрої підтримують тільки близько 4,6 Гбіт/с). Однак з погляду функціональності новий стандарт багато в чому схожий на своїх попередників. Його головною відмітною рисою є робота в діапазоні 60 ГГц – точніше, в інтервалі від 57 до 66 ГГц залежно від країни (в України ГКРЧ дозволила використовувати весь діапазон для пристроїв стандарту 802.11ad).

Як відомо, чим вище частота, тим сильніше загасання. Відповідно, зв'язок можливий тільки в умовах прямої видимості на відстані до 10 м, оскільки навіть одна стіна стає непереборною перешкодою для сигналу WiGig. У випадку з'єднань точка-точка при використанні спеціальних антен дальність може бути збільшена, що дозволить підключати станції стільникового зв'язка.

Завдяки доступності широкого діапазону частот 9 ГГц WiGig дозволяє передавати більші обсяги даних. Споконвічно цей стандарт розроблялася як заміна кабельного підключення, принаймні в межах кімнати. Як перспективні додатки розглядаються обмін файлами (з'єднання комп'ютерів із принтерами й іншою периферією), потокова передача відео з високим дозволом, доповнений реальність і т.п.

Як і у випадку з діапазоном 5 ГГц, навряд чи перехід на 60 ГГц буде швидким. Для прискорення цього процесу передбачена підтримка динамічного перемикання сеансів, тобто пристрою Wi-Fi зможуть динамічно перемикатися між стандартними діапазонами 2,4 і 5 ГГц і новим 60 ГГц. ABI Research прогнозує, що до 2027 року буде продано 4,7 млрд пристроїв WiGig.

802.11AX

Багато які з висунутих вимог і запитів покликані задовольнити розроблювальний стандарт 802.11ax. Він повинен буде забезпечити швидкість до 10 Гбіт/с у діапазоні не тільки 5 ГГц, але й 2,4 ГГц, що не передбачено в 802.11ac. Однак головним достоїнством цього стандарту стануть не підвищення швидкості й розширення діапазону, а збільшення щільності підключень. Він розробляється розраховуючи на підтримку таких ресурсномістких додатків, як потокове відео, віртуальні приватні мережі й відеоконференції в гетерогенних мережах з безліччю активних користувачів.

У випадку провідних підключень з'єднань 100 Мбіт/із цілком достатньо для більшості наявних додатків. Тим часом Wi-Fi упевнено переборов планку 1 Гбіт/с. Однак провідні мережі є що комутируються, тоді як бездротові – поділюваними. Інакше кажучи, доступна смуга пропущення ділиться між декількома абонентами, і кожний одержує свою частку. Відповідно, проблема не стільки в пропускній здатності бездротової системи, скільки в усуненні можливих перевантажень і конфліктів.

Попередні стандарти Wi-Fi були розраховані на більш-менш нерегулярне використання, при цьому передбачалося, що трафік асиметричний: обсяг завантаження більше обсягу вивантаження. 802.11ax вирішує проблему високої щільності шляхом перегляду механізмів роботи Wi-Fi, для чого були запозичені деякі кращі практики LTE. Одна з них – використання множинного доступу з ортогональним частотним поділом (Orthogonal Frequency Division Multiple Access, OFDMA).

У випадку OFDMA кожний канал ділиться на безліч більше дрібних підканалів зі своєю частотою й до 30 клієнтів можуть спільно використовувати канал, замість того щоб чекати зручного моменту для передачі. У результаті підвищується ефективність його використання, тому що абонентам не доводиться конкурувати за канал, розсилаючи широкомовні повідомлення, коли канал звільняється.

OFDMA ділить спектр на так звані тимчасово-частотні ресурсні блоки (Resource Unit, RU). Точка доступу 802.11ax, що виконує функції центрального координатора, розподіляє підканали (RU) між станціями для прийому й передачі, тому з погляду абонента мережа виявляється неперевантаженою.

Завдяки можливості одночасного використання діапазонів 2,4 і 5 ГГц число доступних для передачі даних каналів може бути додатково збільшено. А використання квадратурної амплітудної модуляції (Quadrature Amplitude Modulation, QAM) дозволяє передавати більше даних в одному пакеті. Якщо в 802.11ac багатокористувальницька передача здійснювалася тільки при завантаженні, то в 802.11ax підтримка MU-MIMO MU стала двонаправленою.

Серед інших поліпшень 802.11ax – значне збільшення часу роботи від батареї. За допомогою функції «Пробудження за розкладом» (Wake Time Scheduling) точка доступу зможе повідомляти клієнта, коли йому можна заснути й коли прокинутися. Періоди сну нетривалі, але їх досить багато, що дозволяє значно заощадити заряд батареї.

Поява перших продуктів 802.ax очікується вже в наступному році, хоча стандарт навряд чи буде прийнятий раніше 2027 року.

Wi-Fi для різних завдань

Донедавна кожна чергова версія стандарту Wi-Fi була покликана замінити свого попередника. Так, наприклад, 802.11ac швидше, менш чутливий до перешкод і підтримує більше клієнтів, чим його попередник 802.11n. Однак через розмаїтість завдань, для яких використовується Wi-Fi, підхід «один стандарт на всі випадки життя» перестає відповідати пропонованим вимогам. Це проявляється в розробці специфічних стандартів для різних застосувань.

Першим таким стандартом став 802.11ad (WiGig). Він забезпечує високу швидкість передачі, але на невеликій відстані. Одним із цільових додатків є трансляція потокового відео з мобільного пристрою на телевізор. Так, по оцінках Strategy Analytics, передача відео ультрависокої чіткості (UHD) обсягом 60 Гбайт займе менш 2,5 хв.

Можливість використовувати широку, якщо не повсюдну, інфраструктуру здешевляє підключення пристроїв Інтернету речей і робіть Wi-Fi природним кандидатом для цих цілей. Багато хто з них, імовірно, будуть підключатися за допомогою розроблювального стандарту 802.11ah. Якщо WiGig призначений для передачі більших обсягів даних на короткі відстані, то 802.11ah, що одержав назву HaLow, навпаки, орієнтований на передачу менших обсягів на досить значні відстані, причому, що важливо для додатків IoT, стандарт розрахований на дуже мале споживання енергії.

Специфікація HaLow схвалена, чипмережі доступні на ринку, але деякі аналітики вважають, що через занадто пізню появу йому буде важко скласти конкуренцію стандартам стільникового зв'язка, що пропонує схожі можливості, наприклад LTE-M. До того ж поширенню HaLow буде перешкоджати і його робота в нетиповому для Wi-Fi субгігагерцевому діапазоні, причому в різних регіонах світу використовуються різні частоти: 700 Мгц у Китаї, 850 Мгц у Європі й 900 Мгц у США. По оцінках ABI Research, до 2026 року щорічно буде продаватися 11 млн пристроїв із чипсетами 802.11ah. З урахуванням прогнозів росту Інтернету речей це зовсім небагато.

Втім, для підключення пристроїв IoT можуть використовуватися й інші стандарти Wi-Fi. Так, наприклад, 802.11ax розглядається в якості одного з кандидатів для забезпечення роботи IoT; зокрема, завдяки підтримці MU-MIMO і меншим інтервалам між каналами (78,125 кГц), 18 клієнтів можуть одночасно відправляти дані в каналі 40 Мгц. А стандарт 802.11p передбачає підтримку виділених коротких з'єднань (Dedicated Short-Range Communications, DSRC), які можуть застосовуватися, наприклад, у безпілотних автомобілях.

Ще один новий стандарт, 802.11af (так званий White-Fi або SuperWi-Fi), розрахований на передачу більших обсягів даних на більші відстані. Він використовує вільний частотний спектр між телевізійними каналами (смуги 6–8 МГц). Конкретні частоти в діапазоні від 54 до 790 МГц кожна країна визначає самостійно. Завдяки дальності передачі до декількох кілометрів, основне призначення цього стандарту – забезпечення зв'язку в сільській місцевості.

За неповні 20 років своєї історії технологія Wi-Fi проникнула майже в усі електронні пристрої. Точки доступу тепер працюють на гігабітних швидкостях, навчилися управляти діаграмою спрямованості й можуть обслуговувати декількох абонентів одночасно. У комітету 802.11 давно закінчилися букви для позначення нових технологій (і відповідних робітників груп), і він перейшов на двобуквені найменування.

Орієнтуватися в цьому різноманітті стандартів, підстандартів і доповнень стає усе складніше: крім наявних технологій, необхідно брати до уваги й ті, які от-от з'являться, адже вони обіцяють швидкість 5-10-20 Гбіт/с. Але ми живемо тут і зараз, і найчастіше нас цікавить більше практичне завдання: як вибрати рішення на найближчі п'ять років, щоб згодом не пошкодувати про це. Тому дозвольте сфокусуватися на цьому питанні, не претендуючи на всеосяжність.

Вибір стандарту: n, ac Wave 1, ac Wave 2

На даний момент при побудові мережі Wi-Fi не можна повністю відмовитися від стандарту n, тому що стандарти ac використовуються в діапазоні 5 ГГц, а значна частина кінцевих пристроїв працює в 2,4 ГГц. Тому, швидше за все, нова точка доступу буде дводіапазоною: n (2,4 ГГц) + ac Wave 1/2 (5 ГГц). В офісі їсти зміст використовувати ТД 802.11ac Wave 2: можливо, уже зовсім незабаром багато ваших колег будуть мати пристрою з підтримкою каналів 160 Мгц і режиму MU-MIMO. Що стосується вуличних мереж, в України навряд чи вдасться витягти перевагу від реалізації ac Wave 2: якщо повезе, на одну точку доступу вам виділять канал 20 Мгц для 5 ГГц і ще 20 Мгц ви займете в 2,4 ГГц, пославшись на положення про пристрої малого радіуса дії. Крім цього, через моторошну перешкодну обстановку в місті ви, швидше за все, зволієте використовувати спрямовані антени, тому про MU-MIMO прийде забути.

Wi-Fi потрібна інфраструктура

До кожної точки доступу повинен бути прокладений кабель від комутуючої підсистеми (існуючої або планованої). В офісі це звичайно кручена пари, на вулиці – кручена пара або оптика. На сьогоднішній день і на найближчу

перспективу цілком достатньо гігабітних каналів: незважаючи на описувані в стандартах високі швидкості, всі вони є граничним випадком і ставляться до передачі даних у радіоканалі. Реальний трафік через порт Ethernet набагато менше.

Окремі точки доступу можна об'єднати по радіо за допомогою технології Mesh/WDS, тому переконаєтеся в тому, що ця функціональність підтримується. Крім того, уже на етапі планування інфраструктури варто задуматися про живлення. Якщо використовувані комутатори не підтримують Po, поруч із комутатором або точкою доступу прийде розмістити інжектор живлення, якому буде потрібно розетка 220 В.

Контролери Wi-Fi

Контролер Wi-Fi не описується стандартами 802.11, тому на ринку представлено багато різних рішень. Сучасний контролер виконує два основні завдання: керування мережею Wi-Fi і її моніторинг. Контролер може бути реалізований програмно або апаратно. Програмний контролер – це або ПЗ, що запускається на одній із точок доступу в мережі (як правило, це застосовно тільки в невеликих мережах, скажемо до 30 точок доступу), або ПЗ, установлюване на сервері. Окремо слід зазначити хмарні рішення: деякі вендори пропонують такий сервіс, так що користувачам не потрібно встановлювати в мережі власний сервер, стежити за його відновленням і т.д.

Апаратний контролер являє собою окремий пристрій, як правило, того ж виробника, що й точки доступу. По своїй суті, це той же сервер керування, але в маленькому корпусі й з логотипом вендора. Деякі виробники сполучають контролер з комутатором. Один час була навіть мода на «тупі» точки доступу, які передавали всю інформацію в сирому виді на контролер, а останній здійснював всю обробку, але цей підхід не прижився.

Антени

Велика кількість антен у сучасних точок доступу – це змушена необхідність. Оскільки найпростіше рішення – використовувати окрему

зовнішню антену на кожний приймачепередатчик (chain), та загальна кількість антен може досягати шести й навіть восьми. У результаті така точка доступу виглядає як павук і свій зовнішній вигляд може залучити, мабуть, тільки вандала. Сучасні технології дозволяють усередині корпуса розмістити складні антенні системи, у тому числі спрямовані антени з високим коефіцієнтом підсилення й антени з підтримкою MU-MIMO. Якщо в мережі, вуличній або офісній, планується більша щільність розміщення абонентів, зверніть увагу на наявність у портфоліо у виробника антени з вузькою діаграмою спрямованості.

Підтримка старих клієнтів a/b/g

Масовий випуск таких пристроїв закінчився більше 10 років тому, тому їм можна сміло сказати «до побачення» і відключити їхню підтримку у властивостях точок доступу. Якщо ж ви один з тих «щасливчиків», хто використовує у своєму господарстві старі (як правило, спеціалізовані) пристрою a/b/g, то поцікавтеся, чи підтримує устаткування, що здобувається, режим Airtime Fairness. Це не панацея, але хоча б не дозволить старим пристроям «з'їсти» всю пропускну здатність точок доступу.

Безшовний роумінг / хендовер

Сьогодні майже всі власники смарфонів використовують додатки для передачі голосу й відео реального часу (WhatsApp, Viber, Skype, Telegram, FB Messenger і ін.). При переміщенні абонента між точками доступу, зміна точки може зайняти якийсь час і викликати затримку перемикання (handoff-time). При OPEN-автентифікації або доступі з паролем (Pre-shared Key, PSK) затримка перемикання мінімальна (за умови використання якісного устаткування від ведучих вендорів) і не впливає на роботу таких додатків.

Коли використовується сервер RADIUS, зміна точки доступу зажадає повторної автентифікації клієнта на RADIUS, що може зайняти 500 мс і більше, так що перебої у зв'язку гарантовані! Один з механізмів забезпечення бесшовності в цьому випадку – стандарт 802.11r. Його основне завдання – кешувати ключ на точках доступу, щоб абонента не потрібно було повторно

автентифікувати на сервері RADIUS при зміні точки доступу. 802.11r підтримується тільки сучасними мобільними пристроями, тому різні вендори пропонують пропрієтарні методи для «старих» клієнтів, наприклад Opportunistic Key Caching (OKC).

Розширена функціональність. Опис можливостей сучасних точок доступу може зайняти кілька аркушів, тому зупинимося на основних функціях, які вам напевно будуть потрібні:

а) розумне керування радіопараметрами (сканування сусідніх каналів без скидання абонентських сеансів, можливість перемикання каналу при погіршенні перешкодну обстановки, керування вихідною потужністю для зниження внутрісистемної перешкоди або для керування покриттям території);

б) підтримка власного гостьового порталу й/або можливість http/ https-редиректа на зовнішні гостьові портали;

в) підтримка Band Steering для підключення дводіапазоних клієнтів у діапазоні 5 ГГц, що, як правило, менш завантажений перешкодами;

г) балансування навантаження, що забезпечує рівномірний розподіл клієнтів між декількома точками доступу, що виключає наявність перевантажених точок при пустуючих сусідніх;

д) технологія Passpoint (Hotspot 2.0) зацікавить, мабуть, тільки мобільних операторів: вона дозволяє автентифікувати абонента на основі SIM-Карти й направляти мобільний трафік через мережу Wi-Fi, коли абонент виявляється в зоні її дії;

е) API для розроблювачів буде затребуваний там, де мережа Wi-Fi будується для рішення нестандартних завдань (наприклад, для одержання даних про місце розташування абонентів або для інтерактивних додатків з урахуванням поводження абонентів).

Ціна

Останній пункт у цьому переліку, але не останній за своїм значенням для замовника – це ціна рішення, що складається з вартості точок доступу, ліцензій

для розблокування потрібного функціонала (наприклад, 802.11r або 802.11s), контролера, ліцензій на підключення точок доступу до контролера, послуг техпідтримки. Якщо ви орієнтуєтеся на рішення з вищої ліги, то будьте готові заплатити по всіх перерахованих статтях.

Wi-Fi освоює всі нові й нові території, допомагаючи вирішувати усе більше складні завдання: масштаб претензій прихильників цієї технологи на нові ринкові ніші росте от уже кілька років. Так, у прогнозі Wi-Fi Alliance на поточний рік вказується, що завдяки новим стандартам Wi-Fi здатний упоратися з більшістю тих сценаріїв, де потрібні висока ємність і низька затримка, для реалізації яких і створюється наступне покоління стільникового зв'язка 5G.

Крім описаних у роботі стандартів, ведуться й інші численні розробки для поліпшення характеристик і функціональності Wi-Fi, у яких поряд з IEEE активну роль грає Wi-Fi Alliance. Так, наприклад, у рамках ініціатив Multiband Operations і Optimized Connectivity Experience ведеться пошук способів автоматичного розподілу клієнтів між точками доступу й частотних діапазонів.

Всі прогнози по подальшому поширенню Wi-Fi досить оптимістичні. Кількість використовуваних пристроїв Wi-Fi уже перевищило чисельність населення Землі. По даним Wi-Fi Alliance, тільки цього року буде продане близько 3 млрд пристроїв з підтримкою Wi-Fi, а під кінець року їхнє загальне число складе майже 9 млрд. Так що краще для цієї технології, незважаючи на її солідний 20-літній вік і достаток конкурентів, звичайно, спереду.

3.2 Розробка структурної схеми

Інтернет речей і хмари нероздільні, відтворення технічної бази рішення й виявлення даних для аналізу в більшості випадків здійснюється на хмарній платформі обраного провайдеру. Хмара оснащена повним комплексом захисних інструментів від навмисної крадіжки даних, випадкової втрати у випадку відмови IT-систем або людських помилок. Дорогі безвідмовні системи зберігання хмари

здатні забезпечити роботу рішення без простоїв, у свою чергу програмне забезпечення й доступність IT-експертів 24/7 дозволяють відкласти ряд технічних питань убік, приділивши більше часу вдосконаленню самих розумних технологій і аналізу отриманих даних.

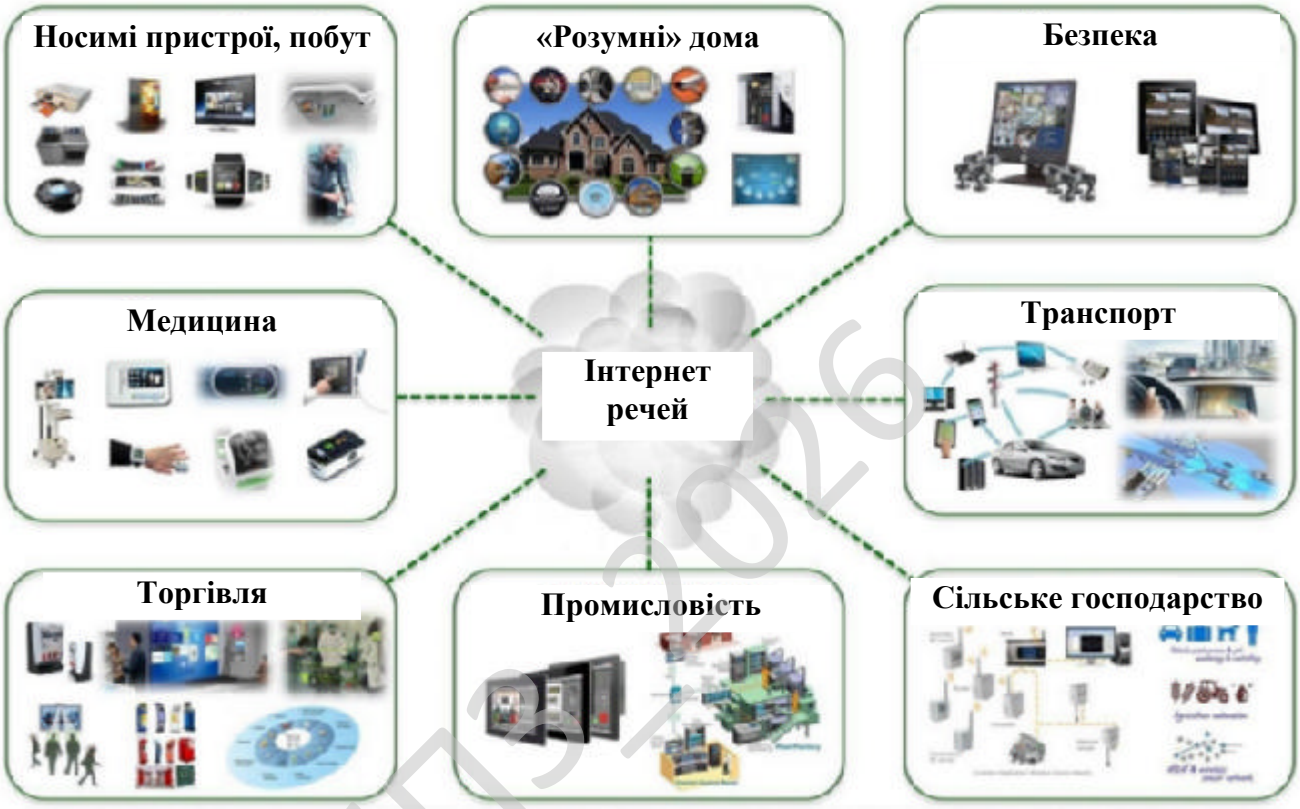

Рисунок 3.1 – Структурна схема системи

Управління споживанням енергії за допомогою підключених пристроїв Інтернету речей

Говорячи про низьке енергоспоживання, одним із найвигідніших способів використання мереж пристроїв Інтернету речей, а отже, однією з ключових тенденцій Інтернету речей, на яку варто звернути увагу, є управління споживанням енергії. Незважаючи на падіння середніх цін на електроенергію в ЄС у 2025 році, вартість електроенергії в регіоні все ще в середньому вища, ніж у США чи Китаї. Хоча нові тенденції свідчать про те, що завдяки зниженню імпорту газу, розгортанню відновлюваних джерел енергії та зростанню гнучкості

та інтеграції розумних мереж ЄС на шляху до досягнення нижчих цін на електроенергію, ніж у США, ситуація все ще достатньо складна, щоб, наприклад, уряд Німеччини запровадив регулювання цін на промислову електроенергію на найближчі роки. Враховуючи значні регіональні відмінності в ціноутворенні енергії, підвищення операційної ефективності в цій галузі, безумовно, буде цінною інвестицією у 2026 році.

Таким чином, серед тенденцій в Інтернеті речей, на які варто звернути увагу у 2026 році, можна очікувати збільшення використання технологій Інтернету речей, таких як датчики Інтернету речей, та аналітика даних для зменшення споживання енергії в промислових умовах. Цінні дані про сучасні системи Інтернету речей походять, серед інших факторів, від деталізації даних, що надаються рішеннями Інтернету речей. Споживання енергії кожного підключеного пристрою може вимірюватися індивідуально та з високою частотою, що виявляє будь-які відхилення від норми. Цю інформацію від взаємопов'язаних пристроїв можна використовувати для контролю того, які пристрої не працюють з піковою ефективністю або знаходяться під загрозою поломки, а також які можна періодично вимикати, щоб обмежити споживання енергії в періоди пікового навантаження. У сукупності ці дані можна використовувати для прогнозної аналітики, щоб коригувати попит і потужність відповідно до коливань тарифів на електроенергію та пом'якшувати такі проблеми, як втрати реактивної потужності .

Один із способів використання аналітики даних Інтернету речей у цій якості – це такі рішення, як система управління енергією. Це основна підсистема екосистеми IoT, яка відповідає за моніторинг споживання енергії, включаючи не лише електроенергію, а й інші середовища, стиснене повітря та промислові гази, а також за аналіз даних про споживання та якість енергії. Вона може попереджати користувачів про досягнення або перевищення лімітів, допомагаючи запобігти штрафам та забезпечуючи енергетичну стабільність, дозволяючи оптимізувати витрати на енергію та створювати звіти відповідно до ISO 50001 та ESG. Коротше кажучи, поєднуючи збір промислових даних з розширеною аналітикою, ця

система надає керівникам виробництва практичну інформацію, що дозволяє розумніше керувати енергією.

Штучний інтелект у виробничих процесах – прогнозна аналітика цін на енергоносії

Коли йдеться про управління енергією, однією з тенденцій у технології Інтернету речей, на яку варто звернути увагу у 2026 році, є зростаюче використання штучного інтелекту. На відміну від часто перебільшених випадків використання, пов'язаних з LLM, спеціалізовані алгоритми машинного навчання можуть запропонувати реальні переваги, прогнозуючи постачання електроенергії та коливання цін. Це може бути надзвичайно корисним, оскільки на деяких ринках вартість електроенергії може значно змінюватися день у день або навіть щогодини, залежно від попиту та доступності, особливо з відновлюваних джерел. Погода також може мати величезний вплив (наприклад, коли рівень води в польській річці Вісла падає під час літніх посух, обмежуючи потужність охолодження на електростанціях, тоді як усі кондиціонери працюють на повну потужність).

Здатність реагувати на ці зміни – це ще один виклик, який можуть вирішити такі IoT-рішення, що використовує моделі штучного інтелекту для надання точних короткострокових прогнозів цін на енергоносії. Алгоритми машинного навчання використовуються для аналізу численних параметрів, включаючи ринкові дані (ціни на енергоносії, попит на електроенергію, виробництво відновлюваної енергії) та інформацію про погоду (температура, опади, хмарність, швидкість вітру), для прогнозування очікуваного зростання та падіння цін на енергоносії на строк до тижня вперед, що дозволяє планувати виробництво на найбільш економічно ефективні часові вікна. Завдяки розумному плануванню виробництва енергоємні процеси можна узгодити з періодами нижчих витрат на енергоносії, тоді як стратегічні закупівлі підтримують рішення про закупівлі та переговори щодо контрактів. Фіксуючи загальні тенденції та щоденні коливання, система дозволяє компаніям краще планувати операції та використовувати сприятливі періоди витрат для підтримки розумніших рішень щодо виробництва та закупівель.

3.3 Розробка функціональної схеми

У даній роботі у якості програмного забезпечення системи реалізації реалізації IoT за допомогою стандартів 802.11ah та 802.11ax, реалізовано програмне забезпечення вимірювання вологості.

На рисунку 3.2 розглянута функціональна схема системи, розглянемо її детально. З об'єктів спостереження, яких може бути 32 (при необхідності це число можливо подвоїти), виходить інформація за допомогою датчиків вологості Sonoff TH. Датчик Sonoff TH призначений для моніторингу температури та вологості в приміщенні в реальному часі. Функціональність датчика передбачає збір інформації та передачу її власнику, використання отриманих даних для управління іншими елементами.

Його характеристики наступні:

- Виробник Sonoff.
- Управління Wi-fi за допомогою смартфона.
- Харчування: 220 Вольт.
- Потужність реле 10 Ампер, 16 Ампер.
- Кількість каналів 1 канал.
- Додаткові характеристики: з датчиком температури, з таймером.

Цей датчик підтримує Wi-fi стандартів 802.11ah та 802.11ax.

Вимір відносної вологості стає усе більше важливим в автоматизації промислового керування. Вимірювальний прилад реєструє відносну вологість навколишнього середовища або в каналі бункера за допомогою ємнісного сенсора.

Датчик перетворить ці виміри в стандартизований сигнал виходу, що готовий до подальшої обробки у розробленій бакалаврській програмі. Сенсорний елемент захищений фільтром. Кожний прилад цієї серії проходить 2-денне випробування.

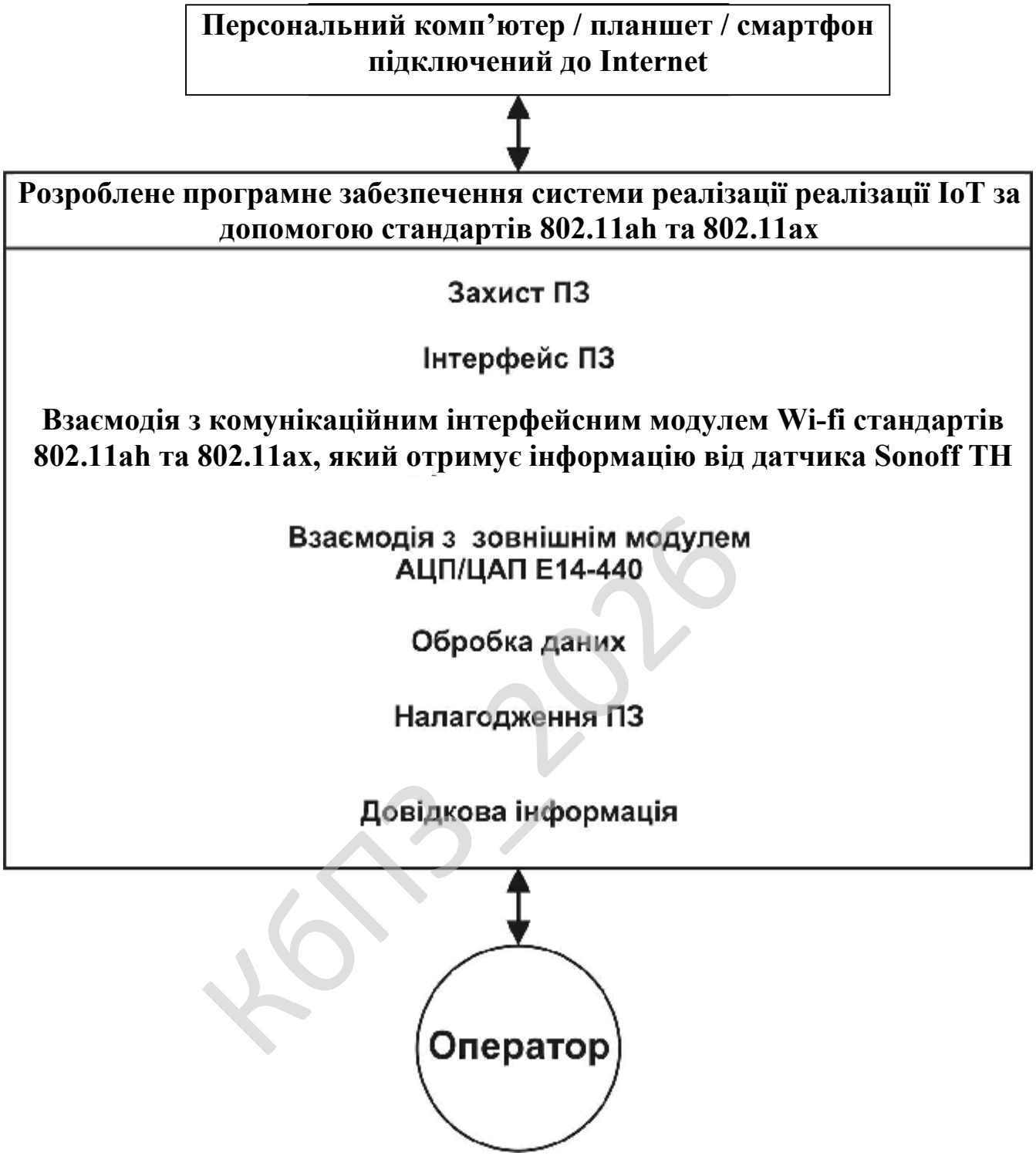

Рисунок 3.2 – Функціональна схема системи

Далі інформація передається на зовнішній модуль АЦП/ЦАП E14-440 котрий розвантажує процесор комп'ютера від керування введенням/виведенням і первинною обробкою сигналів. Програмно здійснюється вибір діапазонів виміру,

частоти дискретизації й конфігурація входів АЦП як диференціальних із загальною "землею".

Синхронізація АЦП можлива по зовнішньому синхросигналу або за рівнем вхідного сигналу. Передбачено опцію установки двоканального ЦАП. Функціонально модуль ідентичний платі L-1450. Модуль особливо зручний для створення портативних вимірювальних систем.

Для з`єднання з персональним комп'ютером необхідно використання комунікаційного інтерфейсного модуля Wi-fi стандартів 802.11ah та 802.11ax.

На рисунку 3.2 зображена розроблена функціональна схема. З неї видно як розроблена програма взаємодіє з апаратною частиною. Розроблена дипломна програма розбита на такі частини:

1. Взаємодія з комунікаційним інтерфейсним модулем Wi-fi стандартів 802.11ah та 802.11ax.
2. Взаємодія з зовнішнім модулем АЦП/ЦАП E14-440.
3. Інтерфейс ПЗ.
4. Захист ПЗ.
5. Обробка даних.
6. Налагодження ПЗ.
7. Довідкова інформація.

3.4 Розробка діаграми процесів

Розглянемо розроблену діаграму процесів яка зображена на рисунку 3.3. Основна будова діаграми процесів полягає у графічному представленні складу сукупностей даних, що характеризуються як співвідношення різних частин кожної з сукупностей.

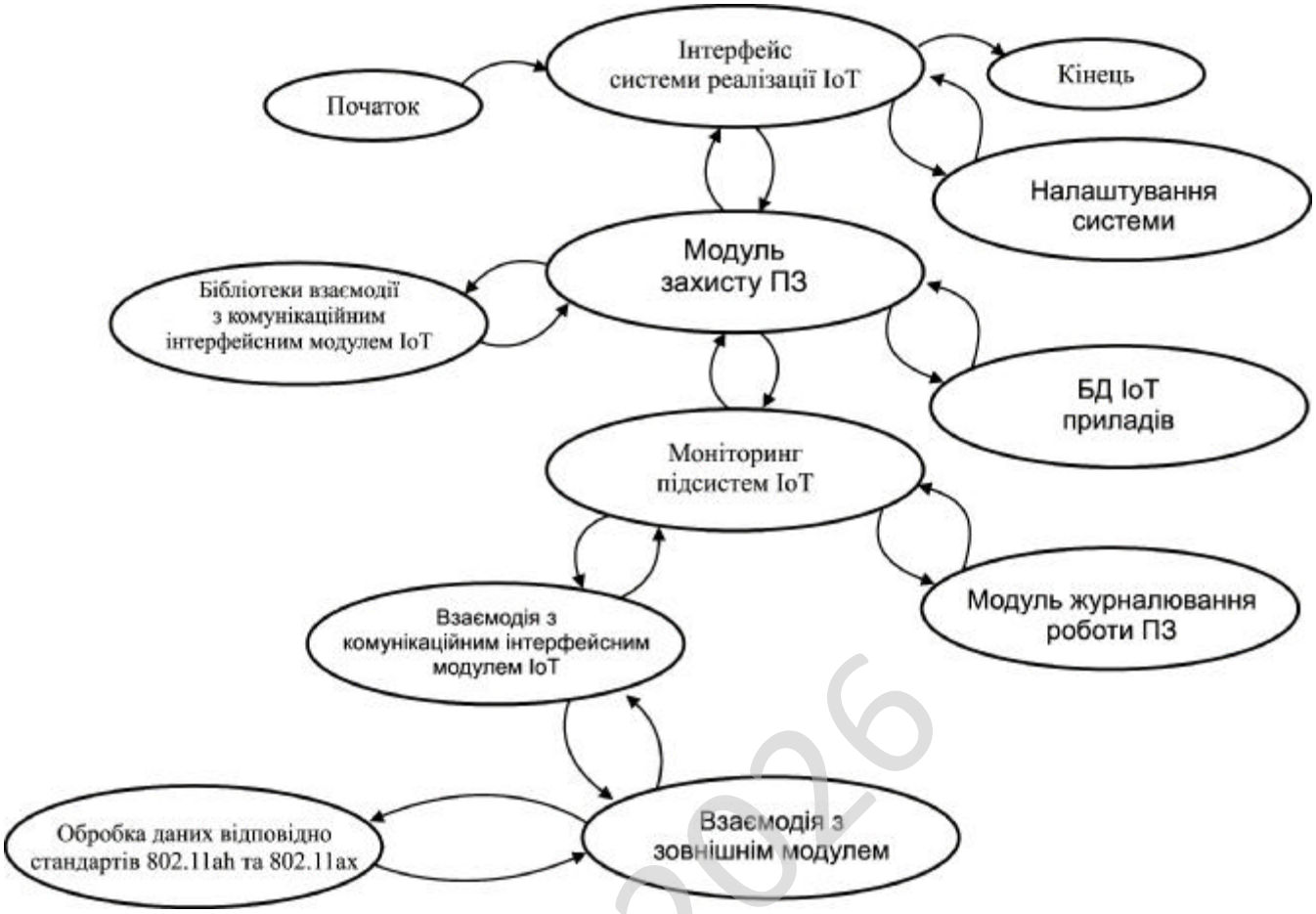

Рисунок 3.3 – Діаграма взаємодії процесів

Склад статистичної сукупності графічно може бути представлений як за допомогою абсолютних, так і відносних показників. Графічне зображення складу сукупності по абсолютними і відносними показниками сприяє проведенню більш глибокого аналізу і дозволяє проводити аналіз системи.

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування).

Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи.

Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

При детальному її розгляді можна побачити як саме проходе взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграми потоків даних містять чотири типи елементів:

– Зовнішні по відношенню до системи сутності.

– Потоки даних між елементами трьох попередніх типів.

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Програмне забезпечення системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax проєктується як комбінована система збору, передавання, зберігання та інтелектуальної обробки даних від сенсорних вузлів.

Стандарт IEEE 802.11ah використовується для енергоощадного зв'язку сенсорів на значній відстані, а стандарт IEEE 802.11ax використовується для високошвидкісного локального сегмента між шлюзом, сервером обробки, панеллю адміністратора та іншими пристроями з більшим навантаженням.

Вихідні припущення для проєктування системи:

1. Кількість сенсорних вузлів становить 120 пристроїв.

2. Один вузол передає телеметричний пакет кожні 30 секунд.

3. Корисне навантаження одного пакета становить 256 байт.

4. Службові поля пакета, ідентифікатор пристрою, контрольна сума, часова мітка та мережеві поля займають приблизно 140 байт.

5. Повний розмір одного повідомлення становить 396 байт.

6. Добова кількість повідомлень від одного вузла становить 86400 / 30 = 2880 повідомлень.

7. Добова кількість повідомлень від усієї системи становить 120 × 2880 = 345600 повідомлень.

8. Обсяг корисного трафіку за добу становить 345600 × 396 = 136857600 байт. 136857600 байт ≈ 130,5 Мбайт.

З урахуванням індексів бази даних, журналу подій, службових записів і резерву приймається коефіцієнт 2,2.

Добовий обсяг збереження становить

130,5 × 2,2 = 287,1 Мбайт

Обсяг збереження за 30 діб становить

287,1 × 30 = 8613 Мбайт

8613 Мбайт ≈ 8,4 Гбайт

Отже, локальна база SQLite або PostgreSQL на сервері шлюзу забезпечує збереження телеметрії щонайменше за 30 діб навіть на малопотужному одноплатному комп'ютері з накопичувачем від 64 Гбайт.

Це підтверджує коректність рішення щодо локального накопичення даних перед передаванням у зовнішню інформаційну систему.

Розрахунок середнього трафіку одного сенсорного вузла

396 байт × 8 = 3168 біт.

3168 / 30 = 105,6 біт за секунду.

Розрахунок середнього трафіку всієї сенсорної мережі

105,6 × 120 = 12672 біт за секунду.

12672 біт за секунду ≈ 12,7 кбіт за секунду.

Навіть з резервом у 10 разів загальне навантаження сенсорного сегмента становить приблизно 127 кбіт за секунду.

Це значення є малим для IEEE 802.11ah, тому вибір цього стандарту для сенсорного рівня є обґрунтованим. Основною перевагою є не швидкість, а енергоощадність, краща дальність і підтримка великої кількості малонавантажених IoT пристроїв.

Розрахунок енергоспоживання сенсорного вузла

Сенсорний вузол працює циклічно. Більшу частину часу він перебуває у режимі сну. Один цикл триває 30 секунд.

Струм у режимі сну становить 0,02 мА.

Струм приймання становить 15 мА протягом 0,2 секунди.

Струм передавання становить 120 мА протягом 0,05 секунди.

Струм роботи мікроконтролера становить 8 мА протягом 0,1 секунди.

Середній струм одного циклу становить

Iсер = 0,02 × 29,65 + 15 × 0,2 + 120 × 0,05 + 8 × 0,1.

Iсер = 0,593 + 3 + 6 + 0,8.

Iсер = 10,393 мА × с.

Середній струм за 30 секунд становить

10,393 / 30 = 0,346 мА.

За використання батареї 2400 мА год і коефіцієнта доступної ємності 0,8 час автономної роботи становить

2400 × 0,8 / 0,346 = 5549 годин.

5549 / 24 = 231 доба.

Отже, один сенсорний вузол працює орієнтовно 7,5 місяців без заміни джерела живлення. Якщо інтервал передавання збільшується до 60 секунд, час автономної роботи зростає приблизно до одного року. **Це підтверджує доцільність використання механізмів енергозбереження, зокрема періодичного сну та узгодженого часу пробудження.**

Розрахунок радіопокриття для IEEE 802.11ah.

Для сенсорного вузла приймається частота 900 МГц, відстань 250 м, потужність передавача 14 дБм, коефіцієнт підсилення антени передавача 2 дБ, коефіцієнт підсилення антени приймача 2 дБ.

Втрати у вільному просторі розраховуються так:

FSPL = 32,44 + 20 log 0,25 + 20 log 900.

FSPL = 32,44 − 12,04 + 59,08.

FSPL = 79,48 дБ.

Додаткові втрати на перешкоди приймаються 15 дБ. Рівень прийнятого сигналу становить:

Pr = 14 + 2 + 2 − 79,48 – 15.

Pr = −76,48 дБм.

Якщо чутливість приймача для стабільного режиму становить близько −95 дБм, запас радіолінії становить:

−76,48 − −95 = 18,52 дБ.

Запас понад 10 дБ є достатнім для стабільної роботи в умовах помірних перешкод. Тому стандарт 802.11ah раціонально використовується для датчиків, які розміщуються на території будівлі, складу, навчального корпусу, лабораторії або промислового майданчика.

Розрахунок навантаження сегмента IEEE 802.11ax

Сегмент IEEE 802.11ax використовується для зв'язку шлюзів, сервера обробки, панелі оператора, відеосенсорів або мобільних клієнтів. Для розрахунку приймається 6 вузлів підвищеного навантаження з потоком 2 Мбіт за секунду та 20 клієнтів панелі моніторингу з потоком 0,2 Мбіт за секунду.

Загальне навантаження становить

6 × 2 + 20 × 0,2 = 16 Мбіт за секунду

Для проєкту приймається робоча пропускна здатність сегмента IEEE 802.11ax 150 Мбіт за секунду з урахуванням завад, службових кадрів, повторних передавань та одночасної роботи клієнтів.

Коефіцієнт використання каналу становить

16 / 150 = 0,107.

0,107 × 100 = 10,7 відсотка.

Отже, сегмент IEEE 802.11ax має достатній запас пропускної здатності. Це підтверджує правильність архітектурного рішення, за якого сенсорна частина працює через 802.11ah, а локальна обробка, адміністративний доступ і передавання агрегованих даних виконуються через 802.11ax.

Загальна характеристика системи

Розроблюване програмне забезпечення є серверною та прикладною частиною IoT системи, яка збирає дані від сенсорних вузлів, перевіряє їхню цілісність, зберігає у базі даних, виконує первинний аналіз, формує сповіщення та надає оператору зручний інтерфейс для контролю стану об'єктів.

Система працює у змішаній мережевій архітектурі. На нижньому рівні розміщуються автономні сенсорні вузли. Вони вимірюють температуру,

вологість, освітленість, рух, рівень вібрації, напругу живлення або інші параметри залежно від призначення об’єкта. Кожен вузол має унікальний ідентифікатор, локальний годинник, модуль бездротового зв’язку IEEE 802.11ah, контролер живлення і набір сенсорів.

На проміжному рівні працює шлюз IoT. Він приймає повідомлення від сенсорних вузлів через 802.11ah, виконує буферизацію, перевіряє формат пакета, записує дані у локальну чергу та передає їх до серверної частини. Шлюз також формує службові команди для сенсорів, наприклад зміну інтервалу передавання, перезапуск, переведення у режим економії енергії, синхронізацію часу та оновлення конфігурації.

На верхньому рівні працює сервер Python. Він реалізує приймання даних, логіку обробки, роботу з базою даних, аналітичні функції, систему тривог і прикладний інтерфейс. Для збереження даних використовується SQLite у локальній версії або PostgreSQL у розгорнутій версії. Для обміну повідомленнями між внутрішніми модулями використовується черга подій. Для візуалізації використовується web панель або консольний інтерфейс адміністратора.

Призначення стандартів 802.11ah та 802.11ax у системі

Стандарт IEEE 802.11ah у системі використовується для сенсорних вузлів, які передають невеликі пакети даних з великими інтервалами часу. Цей стандарт добре відповідає задачам IoT, оскільки забезпечує роботу на більшій відстані, краще проходження сигналу через перешкоди та можливість економного використання батареї. У межах проєкту 802.11ah виконує роль енергоощадного каналу збору телеметрії.

Стандарт IEEE 802.11ax у системі використовується для високошвидкісного локального сегмента. Він забезпечує підключення шлюзів, сервера, робочих станцій адміністратора, мобільних клієнтів та інших пристроїв, які потребують більшої пропускної здатності. За рахунок механізмів ефективного розподілу ресурсу каналу 802.11ax краще працює у середовищі з великою кількістю клієнтів.

Таке розділення ролей забезпечує баланс між енергоефективністю та продуктивністю. Сенсорні вузли не витрачають зайву енергію на високошвидкісний зв’язок, а серверна частина не обмежується низькою пропускною здатністю сенсорного сегмента.

Архітектура програмного забезпечення

Архітектура системи має модульну структуру. Кожен модуль відповідає за окремий набір функцій і взаємодіє з іншими модулями через чітко визначені об’єкти даних. Основні програмні модулі системи:

1. Модуль приймання телеметрії. Цей модуль приймає повідомлення від IoT шлюзу. Він перевіряє наявність ідентифікатора пристрою, часової мітки, типу сенсора, виміряного значення, рівня батареї та контрольної суми. Якщо пакет має неправильний формат, система записує подію у журнал і не додає значення до основної таблиці вимірювань.

2. Модуль валідації даних. Модуль перевіряє допустимість значень. Наприклад, температура не повинна виходити за фізично можливі межі, рівень батареї не може бути більшим за 100 відсотків, а часова мітка не повинна суттєво відрізнятися від часу сервера. Така перевірка захищає систему від пошкоджених пакетів, помилок сенсора і випадкових повторів.

3. Модуль збереження даних. Модуль працює з базою даних. Він створює таблиці пристроїв, вимірювань, подій, команд і сповіщень. Для кожного вимірювання зберігається ідентифікатор пристрою, тип параметра, числове значення, одиниця вимірювання, час отримання, рівень сигналу та рівень батареї.

4. Модуль інтелектуальної обробки. Модуль аналізує часові ряди вимірювань. Він розраховує середні значення, ковзні середні, відхилення від норми, швидкість зміни параметра та ознаки аномальної поведінки. У бакалаврській роботі цей модуль доцільно реалізувати на основі простих статистичних методів, оскільки вони є зрозумілими, перевірюваними і достатніми для первинного IoT моніторингу.

5. Модуль сповіщень. Модуль створює події тривоги, якщо значення виходить за встановлені межі або якщо сенсор не передає дані довше допустимого часу. Система визначає рівень критичності події та зберігає її у базі даних.

6. Модуль конфігурації пристроїв. Модуль зберігає параметри роботи сенсорів. До них належать інтервал передавання, дозволені межі вимірювання, режим енергозбереження, поточний статус, місце встановлення та призначення вузла.

7. Модуль журналювання. Модуль фіксує службові події. До журналу потрапляють помилки приймання пакетів, відхилені повідомлення, зміни конфігурації, створення тривог, вхід адміністратора та технічні події сервера.

8. Модуль прикладного інтерфейсу. Модуль надає оператору доступ до поточного стану системи. Оператор переглядає список пристроїв, останні вимірювання, графіки, тривоги, стан батарей і якість зв'язку. У спрощеній реалізації інтерфейс може працювати як консольна програма, а в розширеній реалізації як web застосунок.

Елементи вихідного коду, модель пакета телеметрії:

```
from dataclasses import dataclass
from datetime import datetime
from typing import Dict, Any
import zlib
# структура одного IoT повідомлення
@dataclass
class TelemetryPacket:
    device_id: str
    sensor_type: str
    value: float
    unit: str
    battery_level: float
    rssi: int
    timestamp: datetime
    checksum: int
    def to_payload(self) -> Dict[str, Any]:
        return {
            "device_id": self.device_id,
            "sensor_type": self.sensor_type,
```

```
            "value": self.value,
            "unit": self.unit,
            "battery_level": self.battery_level,
            "rssi": self.rssi,
            "timestamp": self.timestamp.isoformat(),
            "checksum": self.checksum
        }
#Створення контрольної суми для перевірки цілісності
def calculate_checksum(device_id: str, sensor_type: str, value: float, timestamp:
str) -> int:
    raw_data = f"{device_id}|{sensor_type}|{value}|{timestamp}".encode("utf-8")
    return zlib.crc32(raw_data)
       Перевірка пакета даних:
from datetime import datetime, timedelta
from typing import List
#Коментар описує правила перевірки телеметрії
class TelemetryValidator:
    def __init__(self) -> None:
        self.allowed_units = {"temperature": "C", "humidity": "%", "voltage": "V",
"motion": "bool"}
    def validate(self, packet: TelemetryPacket) -> List[str]:
        errors = []
        if not packet.device_id.strip():
            errors.append("Відсутній ідентифікатор пристрою")
        if packet.sensor_type not in self.allowed_units:
            errors.append("Невідомий тип сенсора")
        if packet.sensor_type in self.allowed_units and packet.unit !=
self.allowed_units[packet.sensor_type]:
            errors.append("Неправильна одиниця вимірювання")
        if packet.battery_level < 0 or packet.battery_level > 100:
            errors.append("Некоректний рівень батареї")
        if packet.rssi < -120 or packet.rssi > 0:
            errors.append("Некоректний рівень сигналу")
        now = datetime.now()
        if packet.timestamp > now + timedelta(minutes=5):
            errors.append("Часова мітка знаходиться у майбутньому")
        return errors
```

Збереження даних у SQLite:

```
import sqlite3
from pathlib import Path
#Коментар описує сховище телеметрії
class IoTDatabase:
    def __init__(self, db_path: str) -> None:
        self.db_path = Path(db_path)
        self.connection = sqlite3.connect(self.db_path)
        self.connection.row_factory = sqlite3.Row
       self.create_tables()
    def create_tables(self) -> None:
        query_devices = """
        CREATE TABLE IF NOT EXISTS devices (
            id INTEGER PRIMARY KEY AUTOINCREMENT,
            device_id TEXT UNIQUE NOT NULL,
            name TEXT NOT NULL,
            location TEXT NOT NULL,
            network_standard TEXT NOT NULL,
            status TEXT NOT NULL,
            last_seen TEXT
        )
        """
        query_measurements = """
        CREATE TABLE IF NOT EXISTS measurements (
            id INTEGER PRIMARY KEY AUTOINCREMENT,
            device_id TEXT NOT NULL,
            sensor_type TEXT NOT NULL,
            value REAL NOT NULL,
            unit TEXT NOT NULL,
            battery_level REAL NOT NULL,
            rssi INTEGER NOT NULL,
            measured_at TEXT NOT NULL,
            received_at TEXT NOT NULL
        )
        """
        query_alerts = """
        CREATE TABLE IF NOT EXISTS alerts (
            id INTEGER PRIMARY KEY AUTOINCREMENT,
            device_id TEXT NOT NULL,
            level TEXT NOT NULL,
            message TEXT NOT NULL,
```

```
                created_at TEXT NOT NULL,
                is_closed INTEGER NOT NULL DEFAULT 0
            )
            """
            self.connection.execute(query_devices)
            self.connection.execute(query_measurements)
            self.connection.execute(query_alerts)
            self.connection.commit()
        def save_measurement(self, packet: TelemetryPacket) -> None:
            query = """
            INSERT INTO measurements (
                device_id, sensor_type, value, unit, battery_level, rssi,
measured_at, received_at
            )
            VALUES (?, ?, ?, ?, ?, ?, ?, ?)
            """
            self.connection.execute(
                query,
                (
                    packet.device_id,
                    packet.sensor_type,
                    packet.value,
                    packet.unit,
                    packet.battery_level,
                    packet.rssi,
                    packet.timestamp.isoformat(),
                    datetime.now().isoformat()
                )
            )
            self.connection.commit()
```

Були проведені розрахунки і підібрані набори тестових даних для перевірки правильності реалізації проектних рішень.

Було створено блок-схеми роботи системи. Перед їх розглядом необхідно провести роз’яснення який саме тип блок-схем використовується.

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо.

Блок-схеми показують весь процес роботи системи з підсистемами та частково доказують правильність вибраних проектних рішень. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об’єктно-орієнтована що вимагає високого рівня декомпозиції задач на класи.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підсистеми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ. При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Було використано підходи з використанням UML, це уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування. Є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, називаної UML-моделлю.

UML був створений для визначення, візуалізації, проектування й документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація. Було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу); Діаграма класів. Діаграма діяльності. Це візуальне представлення графу

діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій.

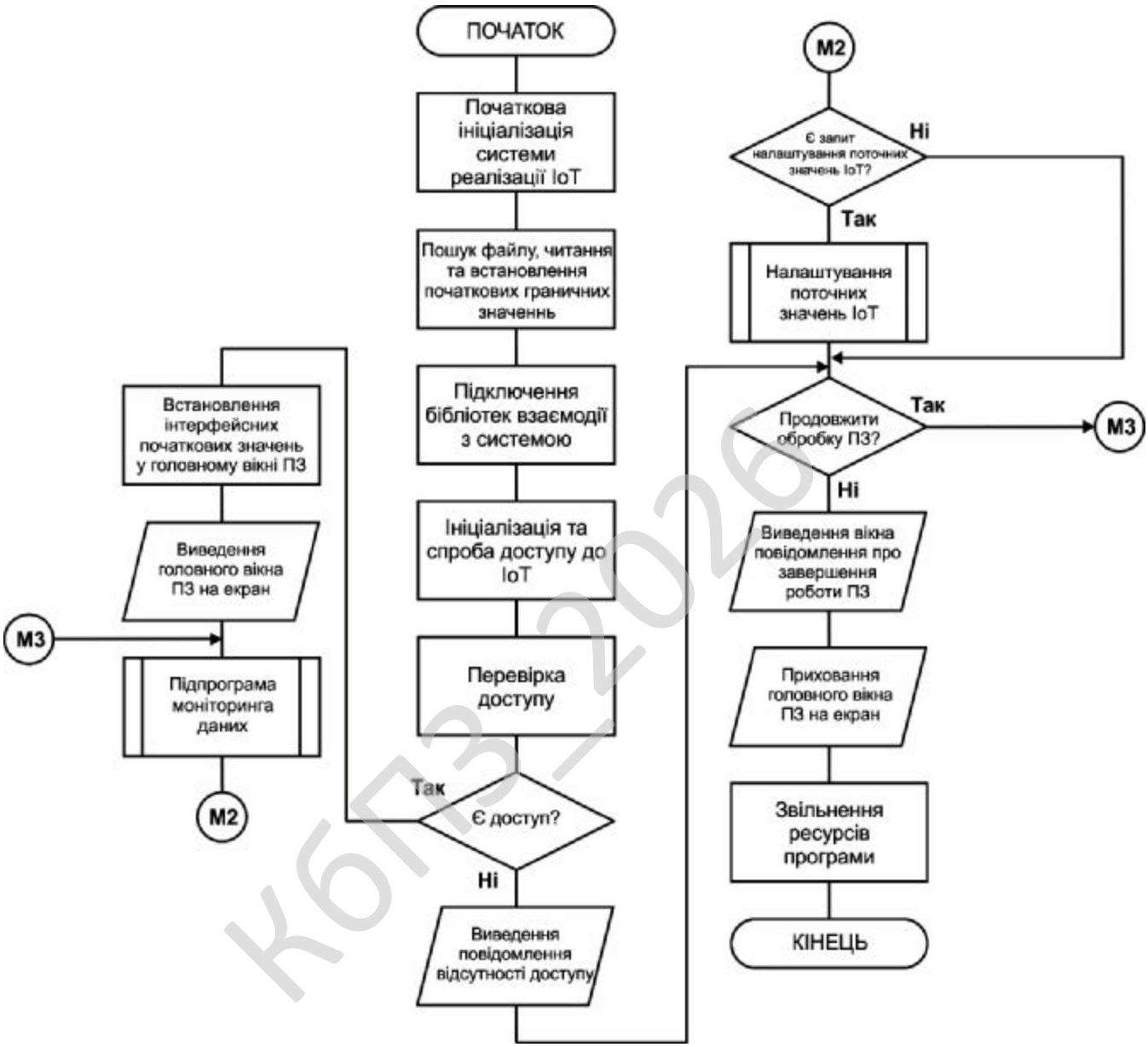

Рисунок 4.1 – Блок-схема основної програми

Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів. Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання

діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності.

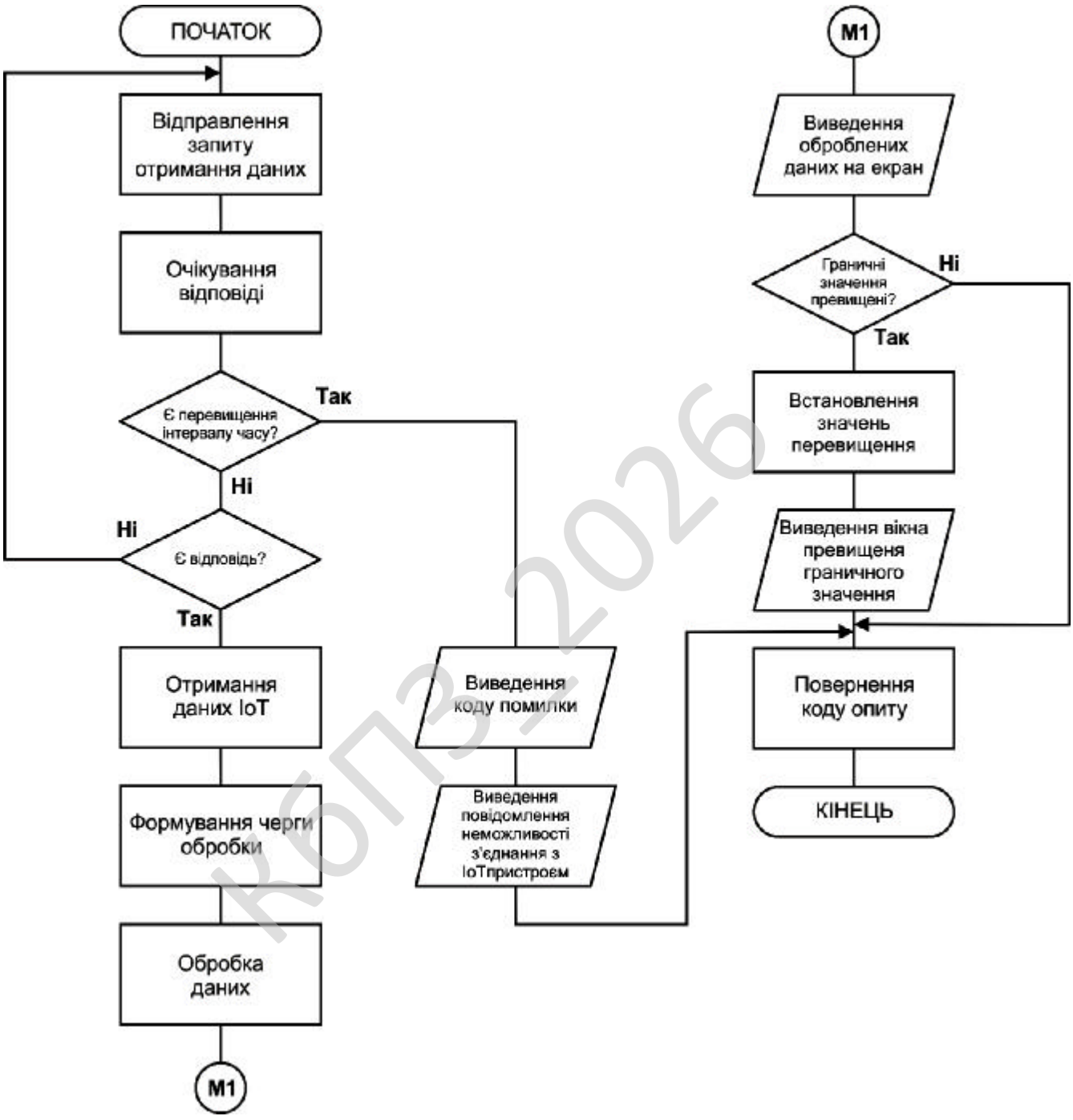

Рисунок 4.2 – Блок-схема роботи підпрограми

Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проектована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

– асоціації (association relationship);

– включення (include relationship);

– розширення (extend relationship);

– узагальнення (generalization relationship).

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме — за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (include) у мові UML - це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності (dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Діаграма класів це статичне представлення структури моделі. Відображає статичні (декларативні) елементи, такі як: класи, типи даних, їх зміст та відношення.

Діаграма класів, також, може містити позначення для пакетів та може містити позначення для вкладених пакетів. Також, діаграма класів може містити позначення деяких елементів поведінки, однак їх динаміка розкривається в інших типах діаграм.

Діаграма класів (class diagram) служить для представлення статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування. На цій діаграмі показують класи, інтерфейси, об'єкти й кооперації, а також їхні відносини.

В UML існують наступні типи зв'язків які використовуються у діаграмі класів: Асоціації; Агрегація; Композиція.

Асоціації це якщо між двома класами визначена асоціація, то можна переміщатися від об'єктів одного класу до об'єктів іншого. Цілком припустимі

випадки, коли обидва кінці асоціації відносяться до одного і того ж класу. Це означає, що з об'єктом деякого класу дозволено зв'язати інші об'єкти з того ж класу. Асоціація, що зв'язує два класи, називається бінарної. Можна, хоча це рідко буває необхідним, створювати асоціації, що зв'язують відразу кілька класів. Графічно асоціація зображується у вигляді лінії, що з'єднує клас сам з собою або з іншими класами.

Асоціації може бути присвоєно ім'я, яке описує природу відносини. Зазвичай ім'я асоціації не вказується, якщо тільки ви не хочете явно задати для неї рольові імена або у вашій моделі настільки багато асоціацій, що виникає необхідність посилатися на них і відрізняти один від одного. Ім'я буде особливо корисним, якщо між одними і тими ж класами існує кілька різних асоціацій.

Клас, що бере участь в асоціації, грає в ній деяку роль. По суті, це "обличчя", яким клас, що знаходиться на одній стороні асоціації, звернений до класу з іншого її боку. Можна явно позначити роль, яку клас грає в асоціації.

Часто при моделюванні буває важливо вказати, скільки об'єктів може бути пов'язано допомогою одного примірника асоціації. Це число називається кратністю (Multiplicity) ролі асоціації та записується або як вираз, значенням якого є діапазон значень, або в явному вигляді.

Вказуючи кратність на одному кінці асоціації, ви тим самим говорите, що на цьому кінці саме стільки об'єктів повинно відповідати кожному об'єкту на протилежному кінці. Кратність можна задати рівною одиниці (1), можна вказати діапазон: "нуль або одиниця" (0..1), "багато" (0 .. *), "одиниця або більше" (1 .. *). Дозволяється також вказувати певне число (наприклад, 3). За допомогою списку можна задати і більш складні кратності, наприклад 0. . 1, 3..4, 6 .. *, що означає "будь-яке число об'єктів, крім 2 і 5".

Агрегація це проста асоціація між двома класами відображає структурний відношення між рівноправними сутностями, коли обидва класу знаходяться на одному концептуальному рівні і ні один не є більш важливим, ніж інший. Але іноді доводиться моделювати відношення типу «частина/ціле», в якому один з

класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

Ставлення такого типу називають агрегацією; воно зараховане до відносин типу «має» (з урахуванням того, що об'єкт-ціле має кілька об'єктів-частин). Агрегація є окремим випадком асоціації і зображується у вигляді простої асоціації з незафарбованим ромбом з боку «цілого». Графічно агрегація представляється порожнім ромбом на блоці класу, і лінією, яка від цього ромба до міститься класу.

Композиція це більш суворий варіант агрегації. Відома також як агрегація за значенням.

Композиція має жорстку залежність часу існування екземплярів класу контейнера та примірників містяться класів. Якщо контейнер буде знищений, то весь його вміст буде також знищено. Графічно представляється як і агрегація, але з зафарбовані ромбиком.

Розглянемо обрані додаткові інструменти розробки системи.

Redmine – вільне серверне ПЗ для управління проектами та відстежування помилок. До системи входить календар-планувальник та діаграми Ганта для візуального представлення ходу робіт за проектом та строків виконання. Redmine написано на мові Ruby і є ПЗ розробленим з використанням відомого веб-фреймворку Ruby on Rails, що означає легкість в розгортанні системи та її адаптації під конкретні вимоги. Для кожного проекту можна вести свої вікі та форуми.

Функціональні можливості:

– Ведення декількох проектів.

– Гнучка система доступу з використанням ролей.

– Система відстеження помилок.

– Діаграми Ганта та календар.

– Ведення новин проекту, документів та управління файлами.

– Сповіщення про зміни за допомогою RSS-потоків та електронної пошти.

– Власна Wiki для кожного проекту.

– Форуми для кожного проекту.

– Облік часових витрат.

– Налаштування власних (custom) полів для задач, затрат часу, проектів та користувачів.

– Легка інтеграція із системами керування версіями (SVN, CVS, Git, Mercurial, Bazaar и Darcs).

– Створення записів про помилки на основі отриманих листів

– Підтримка LDAP автентифікації.

– Можливість самореєстрації нових користувачів.

– Багатомовний інтерфейс (у тому числі українська мова).

– Підтримка СКБД: MySQL, PostgreSQL, SQLite.

Діаграма Ганта (*Gantt chart*, також стрічкова діаграма, графік Ганта) – це популярний тип діаграм, який використовується для ілюстрації плану, графіка робіт за будь-яким проектом. Є одним з методів планування та управління проектами.

Діаграма Ганта являє собою відрізки (графічні плашки), розміщені на горизонтальній шкалі часу. Кожен відрізок відповідає окремому завданню або підзадачі. Завдання і підзадачі, складові плану, розміщуються по вертикалі. Початок, кінець і довжина відрізка на шкалі часу відповідають початку, кінцю і тривалості завдання. На деяких діаграмах Ганта також показується залежність між завданнями.

Діаграма може використовуватися для представлення поточного стану виконання робіт: частина прямокутника, що відповідає завданню, заштриховується, відзначаючи відсоток виконання завдання; показується вертикальна лінія, що відповідає моменту «сьогодні».

Часто діаграма Ганта використовується спільно з таблицею зі списком робіт, рядки якої відповідають окремо взятій задачі, зображеній на діаграмі, а стовпці містять додаткову інформацію про задачу.

Система відстеження помилок Багтрекер – прикладна програма для допомоги розробникам програмного забезпечення (програмістам, тестувальникам тощо) враховувати і контролювати помилки, знайдені у програмах, питання щодо функціональності, рішення та оновлення, побажання користувачів, а також стежити за процесом їх виконання.

Кожному, хто розробляв програмні продукти, добре знайоме співвідношення «20/80» – останні 20 % роботи тривають 80 % часу.

Як це не парадоксально, але нічого дивного в цій пропорції немає, адже саме на завершальній стадії починається тестування проекту, коли виявляються помилки, і що більший проект, то більше буде знайдено помилок.

Водночас досить часто виявляється, що більшість цих помилок були відомі та могли бути виправлені з меншими витратами на попередніх стадіях роботи, але не були вчасно описані, а потім загубилися серед інших важливих завдань.

Отже, система відстеження помилок у найпростішому варіанті – це процес, що включає в себе виявлення помилки, її опис, виправлення і перевірку цього виправлення, тобто процес «стеження» за багом протягом всього як його життєвого циклу, так і життєвого циклу розробки в цілому.

Сукупність інформації про дефект. Головний компонент такої системи – база даних, що містить відомості про виявлені дефекти. Ці відомості можуть включати в себе:

– номер (ідентифікатор) дефекту;

– хто повідомив про дефект;

– дата і час виявлення дефекту;

– версія продукту, в якій виявлено дефект;

– серйозність (критичність) дефекту та пріоритет рішення;

– опис кроків для відтворення дефекту (неправильної поведінки програми);

– відповідальний за усунення дефекту;

– обговорення можливих рішень та їх наслідків;

– поточний стан виправлення дефекту;

– версії продукту, в якій дефект виправлений.

Крім того, розвинені системи надають можливість прикріплювати файли, які допомагають описати проблему, наприклад, дамп пам'яті або скріншот.

Використання. Основна перевага систем відстеження помилок полягає в забезпеченні чітких централізованих оглядів, запитів на розробку (включаючи помилки і виправлення) та їх стан. У корпоративному середовищі, системи відстеження помилок можуть бути використані для генерації звітів по продуктивності програмістів виправлення помилок. Однак, це може іноді приводити до неточних результатів, тому що різні помилки можуть мати різні ступені пріоритету та серйозності, що пов'язано з складністю їх фіксації.

Життєвий цикл дефекту. Як правило, система відстеження помилок використовує той чи інший варіант «життєвого циклу» помилки, стадія якого визначається поточним станом помилки.

Типовий життєвий цикл дефекту:

1. Новий – дефект зареєстрований тестувальником.

2. Призначений – призначений відповідальний за виправлення дефекту.

3. Дозволений – дефект переходить назад у сферу відповідальності тестувальника. Як правило, супроводжується резолюцією, наприклад:

– Виправлено (виправлення включені у версію таку-то).

– Дубль (повторює дефект, що вже знаходиться в роботі).

– Не виправлено (працює відповідно до специфікації, має занадто низький пріоритет, виправлення відкладено до наступної версії тощо).

– «В мене все працює» (запит додаткової інформації про умови, в яких дефект проявляється).

4. Далі тестувальник проводить перевірку виправлення, залежно від чого дефект або знову переходить у стан «Призначений» (якщо він описаний як виправлений, але не виправлений), або у стан «Закрито».

5. Відкрито повторно – дефект знайдено знову в іншій версії.

Система може надавати адміністраторові можливість налаштування користувачі, які можуть переглядати і редагувати помилки залежно від їх стану, переводити їх в інший стан або видаляти.

У корпоративному середовищі, система відстеження помилок може використовуватися для отримання звітів, що показують продуктивність програмістів при виправленні помилок. Однак, часто такий підхід не дає достатньо точних результатів через те, що різні помилки мають різну ступінь серйозності та складності. При цьому серйозність проблеми прямо не стосується складності її усунення.

4.2 Захист розробленого програмного забезпечення

Розроблене програмне забезпечення захистимо за допомогою алгоритму захисту інформації RSA. Спочатку необхідно обчислити пару ключів (секретний ключ і відкритий ключ). Для цього відправник електронних документів обчислює два більших простих числа P и Q, потім знаходить їхній добуток $N = P*Q$ і значення функції $\varphi(N) = (P-1)(Q-1)$. Далі відправник обчислює число E з умов $E < \varphi(N)$, НЗД $(E, \varphi(N)) = 1$ і число D з умов $D < N$, $E*D \equiv 1 (\text{mod } \varphi(N))$.

Пари чисел (E, N) є відкритим ключем. Цю пару чисел автор передає партнерам по переписці для перевірки його цифрових підписів. Число D зберігається автором як секретний ключ для підписування.

Допустимо, що відправник хоче підписати повідомлення M перед його відправленням. Спочатку повідомлення M (блок інформації, файл, таблиця) стискають за допомогою геш-функції h(-) у ціле число m: $m = h(M)$.

Потім обчислюють цифровий підпис S під електронним документом M, використовуючи геш-значення m і секретний ключ D: $S = m (\text{mod } N)$.

Пари (M, S) передається партнерові-одержувачеві як електронний документ M, підписаний цифровим підписом S, причому підпис S сформований власником секретного ключа D.

Після прийому пари (*M*, *S*) одержувач обчислює геш-значення повідомлення M двома різними способами. Насамперед, він відновлює геш-значення m', застосовуючи криптографічне перетворення підпису *S* з використанням відкритого ключа *E*: $m' = S^E \pmod N$.

Крім того, він знаходить результат гешування прийнятого повідомлення *M* с допомогою такої ж геш-функції h(-): $m = h(M)$.

Якщо дотримується рівність обчислених значень, тобто $S^E(\text{mod}) = h(M)$, то одержувач визнає пару (*M*, *S*) справжньою. Доведено, що тільки власник секретного ключа *D* може сформувати цифровий підпис *S* по документі *M*, а визначити секретне число *D* по відкритому числу *E* не легше, ніж розкласти модуль *N* на множники. Крім того, можна строго математично довести, що результат перевірки цифрового підпису *S* буде позитивним тільки в тому випадку, якщо при обчисленні *S* був використаний секретний ключ *D*, що відповідає відкритому ключу *E*. Тому відкритий ключ E іноді називають "ідентифікатором" того, хто підписав.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено розроблене у бакалаврської дипломної роботі програмне забезпечення системи реалізації реалізації IoT за допомогою стандартів 802.11ah та 802.11ax. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

– Функціональних кнопок ПЗ: Оновлення; Встановлення часу; Налаштування граничних даних; Параметри IoT.

– Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.

– Графічного розділу виведення результату роботи системи.

– Розділу оновлення даних.

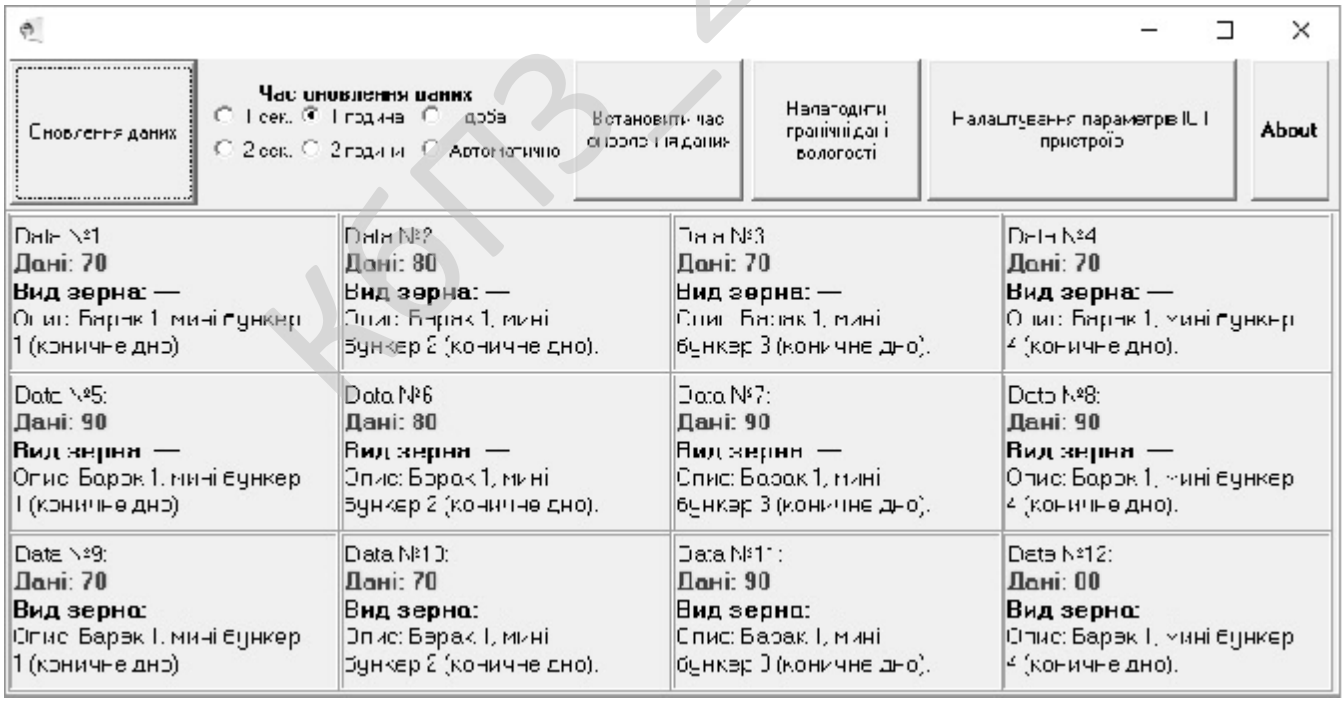

Рисунок – Головне вікно ПЗ

Розроблена програма має дуже простий і зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий. Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

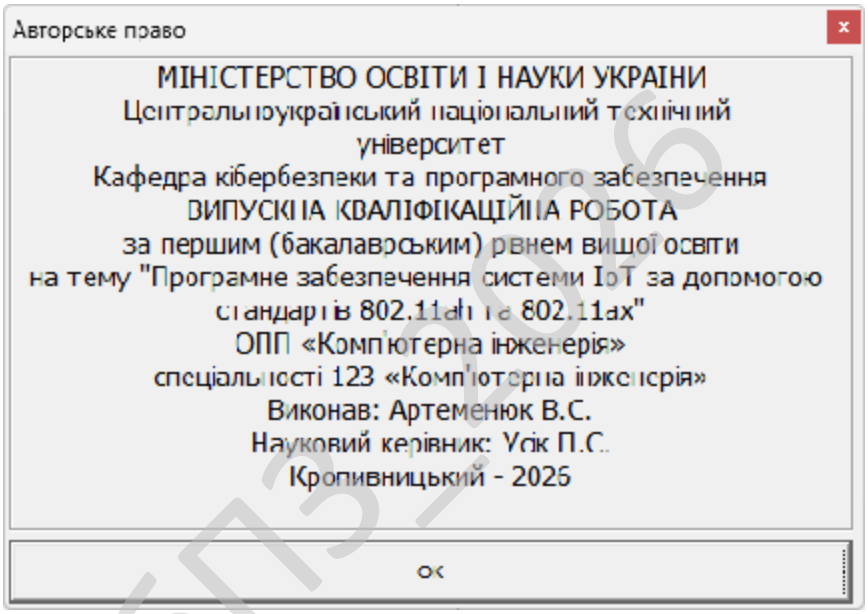

Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку

виробника так і з боку споживача. Оскільки кожна програмна система є унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

– Випуск.

– Встановлення та активація.

– Деактивація.

– Адаптація.

– Обновлення.

– Вмонтування.

– Відстежування версій.

– Видалення.

– Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати ІТ рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене ІТ рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження;

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються в IT рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Обрано умови розповсюдження – commercial software.

Програмне забезпечення, створене комерційною організацією з метою отримання прибутку від його використання іншими, наприклад, шляхом продажу копій.

Найважливішою особливістю комерційний програмних продуктів є підтримка великих компаній, прямо зацікавлених у поширенні програм. Багато організацій надають виключно платну підтримку своїх продуктів, такий підхід, як правило, використовують організації надають відкриті вихідні коди. Для продуктів, що розповсюджуються на комерційній основі діють зазвичай безкоштовні служби підтримки, покликані збільшити рівень довіри у клієнтів і потенційних покупців.

Далеко не завжди, але як правило терміни критично важливих змін в комерційних продуктах значно менше, ніж у некомерційних проектів. Це пов'язано з тим, що над комерційним продуктом працюють цілі групи розробників і ця робота є їх основним заняттям. Розробникам-початківцям як правило доводиться шукати додаткові способи заробітку, і це збільшує час, що витрачається на доповнення і зміни програм. Так як основним рушійним фактором створення комерційного ПЗ є одержання прибутку, то комерційні програмні продукти першими заповнюють вільні ніші та пропонують варіанти вирішення завдань відразу по мірі виявлення вакууму в будь-якому секторі ринку.

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

Рішення завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

– Досліджена система реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

– На основі отриманих результатів досліджень створена програмна реалізація системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання реалізації IoT за допомогою стандартів 802.11ah та 802.11ax.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створені програмного забезпечення було використано об’єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для системи реалізації IoT за допомогою стандартів 802.11ah та 802.11ax. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід,

зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення. Для підвищення рівня безпеки запропоновано застосовувати алгоритм RSA. В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Jack Ganssle and Michael Barr. 2003. Embedded Systems Dictionary. CMP Books.

2. Технології інтернету речей. Навчальний посібник [Електронний ресурс]: / Б. Ю. Жураковський, І.О. Зенів; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 12,5 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2021. – 271 с.

3. Greg Dunko, Joydeep Misra, Josh Robertson, Tom Snyder “A reference guide to the Internet of Things” / 2017 Bridgera LLC, RIoT.

4. Donald Norris “Programming with STM32. Getting started with the Nucleo Board and C/C++” 416 p. 2018.

5. Neil Kolban “Kolban’s book on ESP32”. Texas, USA. 951 p.

6. Вінтенко Б.Ю., Миронець І.В., Смірнов О.А. «Модель комп’ютерно-орієнтованої процедури системи підтримки оперативного персоналу АЕС». *IV Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи розвитку інформаційних технологій (ІПШРІТ-2025)»* м.Черкаси 25 листопада 2025 року – Черкаси: ЧДТУ.– 2025. – С.101-103.

7. Kuznetsov O., Frontoni E., Kuznetsova Y., Chevardin V., Smirnov O. «Architectural foundations for adaptive security in edge computing systems». *Cybersecurity Defensive Walls in Edge Computing,* 2025. pp. 21-61.

8. Вінтенко, Б.Ю., Миронець, І.В., Смірнов, О.А., Коваленко, О.В., Усік, П.С., Буравченко, К.О., Лисенко, І.А. «Логіко-структурна модель комп'ютерно-орієнтованої процедури системи підтримки оперативного персоналу АЕС». *Кібербезпека: освіта, наука, техніка.* 2025. Том 2 № 30. С. 413-427, 2025.

9. Смірнова, Т.В. «Дослідження методів, моделей та сучасних ІТ-рішень для підтримки технологічних процесів у критичній інфраструктурі держави». *Кібербезпека: освіта, наука, техніка.* 2025. Том 2 № 30. С.195-208, 2025.

10. Kuznetsov, O., Frontoni, E., Kuznetsova, K., Arnesano, M., Smirnov, O. «A secure biometric authentication architecture for blockchain-driven cyber-physical systems». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 193–224.

11. Kuznetsov, O., Atzeni, G., Arnesano, M., Randieri, C., Smirnov, O. «Secure IoT-based smart wheelchair system: From implementation to security enhancement strategy». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 225–257.

12. Kuznetsov, O., Smirnov, O., Kuznetsova, T., Shaikhanova, A., Svatowsky, I. «Privacy-utility trade-offs in IoT networks: A comparative analysis of differential privacy mechanisms for sensor data aggregation». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 589–622.

13. Вінтенко Б., Смірнов О., Миронець І., Смірнова Т., Смірнов С. «Імітаційна модель шляхів вхідних даних комп'ютерної інтелектуальної системи підтримки оператора енергоблоку АЕС». *Комбінаторні конфігурації та їхні застосування: Матеріали XXVII Міжнародного науково-практичного семінару, присвяченого 125-річчю Національного університету «Запорізька політехніка»* (Запоріжжя-Кропивницький-Київ, 4-6 червня 2025 р.). Запоріжжя: НУ «Запорізька політехніка», 2025. С.82-91.

14. Al-Azzeh, J., Ayyoub, B., Mesleh, A., Smirnova, T., Gnatyuk, S., Drieiev, O., Smirnov, O., Dorenskyi, O. «Cloud-Based Information System for Evaluating Caverns in the Process of Blasting Metal Surfaces of Details». *International Review on Modelling and Simulations* 18 (1), 2025. pp. 32-42.

15. Вінтенко Б.Ю., Смірнов О.А., Миронець І.В., Смірнова Т.В., Коваленко О.В., Мацуй А.М. «Модель шляхів отримання вхідних даних комп'ютерної інтелектуальної системи підтримки оперативного персоналу АЕС». *Центральноукраїнський науковий вісник. Технічні науки.* 2025. Вип. 11(42), ч. II. С.52-62.

16. Вінтенко Б.Ю., Смірнов О.А., Миронець І.В., Смірнова Т.В. «Методи забезпечення відмовостійкості інтелектуальних систем підтримки оператора». *VIII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології",* м. Кропивницький. 24-25 квітня 2025 р. – Кропивницький: ЦНТУ. – 2025. – С. 44-46.

17. Вінтенко, Б., Миронець, І., Смірнов, О., Коваленко, А., Конопліцька-Слободенюк, О., Смірнова, Т., Константинова, Л. «Дослідження застосування систем підтримки оперативного персоналу об'єкту критичної інфраструктури при керуванні енергоблоком АЕС з реактором типу ВВЕР-1000». *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»,* 2024. № 2(26), С. 6-26.

18. Смірнова Т.В., Конопліцька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка.* 2024. №4(24), С. 6-27.

19. Вінтенко, Б., Миронець, І., Смірнов, О., Кравчук, О., Козірова, Н., Савеленко, Г., Коваленко, А. «Дослідження вимог та аналіз кібербезпеки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Кібербезпека: освіта, наука, техніка.* 2024. №3(23), С. 111-131.

20. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.

21. Kuznetsov, O., Kuznetsova, Y., Smirnov, O., Kostenko, O., Zvieriev, V. «Evaluating Hashing Algorithms in the Age of ASIC Resistance». *CEUR Workshop Proceedings*, 2023, 3628, pp. 93-105.

22. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings,* Volume 3530, 2023, pp. 256-265.

23. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.

24. Вінтенко Б.Ю., Смірнов О.А., Коваленко А.С., Смірнов С.А., Буравченко К.О. «Дослідження вимог міжнародних стандартів IEC60880 та IEC62138 з розробки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 3(73), С. 155-166.

25. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи,* 2023, том 7, № 2, С. 49-56.

26. Smirnova, T., Gnatyuk, S., Yudin, O., Sydorenko, V., Polozhentsev, A., «The Model for Calculating the Quantitative Criteria for Assessing the Security Level of Information and Telecommunication Systems». CEUR Workshop Proceedings Volume 3156, 2022, Pages 390-399.

27. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління,* № 2(70). 2022. С. 28-37.

28. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку,* 2022, № 3(69). С. 93-98.

29. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного

захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»,* № 2 (307). С. 46-52. 2022.

30. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.

31. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». *Communications in Computer and Information Science*, 2021, vol 1486. Springer, Cham. pp 169-184.

32. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.

33. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.

34. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT),* Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

35. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

36. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 125-136.

37. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 366-379.

38. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», *CEUR Workshop Proceedings* Volume 2608, 2020, Pages 633-645.

39. Smirnov O. Kuznetsov A., Zaicenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». *International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019*; Odessa; Ukraine; 9-13 September 2019. P.22-28.

40. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

41. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T),* Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

42. Smirnov, O., Odarchenko, R., Abakumova, A., Usik, P., Kundyz, M., «QoE optimization technique for media delivery in 5G networks». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T),* Kyiv, Ukraine, 8 – 11 October 2019. P.597-601.

43. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». *CEUR Workshop Proceedings*, Vol 2588, P. 90-106, 2019.

44. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», *2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019*/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.

45. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS)*, Kyiv, Ukraine April 17-19, 2019 P. 353-358.

46. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS)*, Kyiv, Ukraine April 17-19, 2019 P. 347-352.

47. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», *CEUR Workshop Proceedings* Volume 2353, *CEUR Workshop Proceedings* 2019, Pages 618-629.

48. Smirnov A.A., Kuzne tsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», *Telecommunications and Radio Engineering*. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

49. Смірнов О.А., Смірнова Т.В., Буравченко К.О., Кравченко С.С., Горбов В.О., «Хмарна система підтримки прийняття рішень технологічного процесу відновлення поверхонь конструкцій і деталей машин». *Сучасні інформаційні системи*. 2021. Т. 5, № 4. С. 79-95

50. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у комп'ютерних системах операторів стільникового зв'язку» *Вісник Черкаського державного технологічного університету. Технічні науки.* №4. С. 103-110. 2020.