

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ
ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**« ІОТ-СИСТЕМА ДЛЯ ЗБОРУ, АНАЛІЗУ ТА ПРОГНОЗУВАННЯ
СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ З ВИКОРИСТАННЯМ EDGE
COMPUTING»**

на здобуття освітнього ступеня магістр
за спеціальності 126 Інформаційні системи та технології
(код, найменування спеціальності)
освітньо-професійної програми Інформаційні системи та технології
(назва)

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на
відповідне джерело*

(підпис)

Микола БОВСУНІВСЬКИЙ
(ім'я, ПРІЗВИЩЕ здобувача)

Виконав:
здобувач вищої освіти
група ІСДМ-61

Микола БОВСУНІВСЬКИЙ
(ім'я, ПРІЗВИЩЕ)

Керівник д.т.н., проф. Каміла СТОРЧАК
(ім'я, ПРІЗВИЩЕ)

Рецензент:

(ім'я, ПРІЗВИЩЕ)

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально-науковий інститут Інформаційних технологій

Кафедра Інформаційних систем та технологій

Ступінь вищої освіти магістр

Спеціальність 126 Інформаційні системи та технології

Освітньо-професійна програма Інформаційні системи та технології

ЗАТВЕРДЖУЮ

Завідувач кафедру ІСТ
Каміла СТОРЧАК

“ ____ ” _____ 2025 року

**З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Бовсунівському Миколі Сергійовичу

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Іот-система для збору, аналізу та прогнозування споживання електроенергії з використанням edge computing

керівник кваліфікаційної роботи: Каміла СТОРЧАК д.т.н., проф.

(ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від “30” жовтня 2025 р. № 467

2. Строк подання кваліфікаційної роботи «26» грудня 2025 р.

3. Вихідні дані кваліфікаційної роботи:

1. Технології Інтернет речей.
2. Архітектура ІоТ.
3. Методи збору, аналізу та прогнозування.
4. Науково-технічна література.

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

1. Аналіз сучасних ІоТ-систем моніторингу електроспоживання та концепції edge computing.
2. Огляд методів збору, обробки та прогнозування даних електроспоживання в ІоТ-системах.
3. Розробка та аналіз результатів впровадження ІоТ-системи прогнозування електроспоживання.

5. Перелік ілюстраційного матеріалу: *презентація*

6. Дата видачі завдання «30» жовтня 2025р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Підбір технічної літератури	15.10 – 23.10.25	
2.	Аналіз предметної області сучасних Iot систем	24.10 – 01.11.25	
3.	Дослідження моделей прогнозування для енергетичної сфери	02.11 – 07.11.25	
4.	Поєднання методів прогнозування з одноплатними компютерами	08.11 – 22.11.25	
5.	Тестування прототипу	23.11 – 03.12.25	
6.	Результати розробки системи збору, аналізу та прогнозування електроенергії	04.12 – 14.12.25	
7.	Оформлення роботи: вступ, висновки, реферат	15.12 – 20.12.25	
8.	Розробка демонстраційних матеріалів, доповідь, презентація	21.12 – 25.12.25	

Здобувач вищої освіти Микола БОВСУНІВСЬКИЙ
(підпис) (ім'я, ПРИЗВИЩЕ)

Керівник кваліфікаційної роботи Каміла СТОРЧАК
(підпис) (ім'я, ПРИЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 73 стор., 14 рис., 4 табл., 42 джерела.

Мета роботи – розробка архітектури IoT-системи для збору, аналізу та прогнозування споживання електроенергії.

Об'єкт дослідження – процеси збору, обробки та прогнозування даних про споживання електроенергії.

Предмет дослідження – архітектурні і алгоритмічні підходи до локальної обробки та прогнозування.

Короткий зміст роботи. У першому розділі магістерської роботи виконано аналіз предметної області інтернету речей, поняття та взаємодію його з локальним рівнем, а також розглянуто існуючі види моделей прогнозування.

У другому розділі виконано огляд досліджень у сфері енергетичного сектору, загальні тенденції та динаміку змін ринку, а також проаналізовано існуючі локальні рішення для прогнозування споживання електроенергії.

У третьому розділі описується процес розробки системи для збору, аналізу та прогнозування споживання електроенергії що працює на локальному рівні. Представлені компоненти, архітектурна схема та вихідний код системи.

КЛЮЧОВІ СЛОВА: ІНТЕРНЕТ РЕЧЕЙ, МОДЕЛЬ, ПРОГНОЗУВАННЯ, СИСТЕМА, ЛОКАЛЬНИЙ РІВЕНЬ, EDGE COMPUTING, ОДНОПЛАТНИЙ КОМП'ЮТЕР, ЕЛЕКТРОЕНЕРГІЯ.

ABSTRACT

The text part of the qualifying work for obtaining a bachelor's degree: 73 pp., 14 fig., 4 tables, 42 sources.

The purpose of the work - to develop the architecture of an IoT system for collecting, analyzing, and forecasting electricity consumption.

The object of research - the processes of collecting, processing, and forecasting data on electricity consumption.

The subject of research - architectural and algorithmic approaches to local processing and forecasting.

Brief summary of the work. The first chapter of the master's thesis analyzes the subject area of the Internet of Things, its concepts and interaction with the local level, and considers existing types of forecasting models.

The second chapter provides an overview of research in the energy sector, general trends and market dynamics, and analyzes existing local solutions for forecasting electricity consumption.

The third chapter describes the process of developing a system for collecting, analyzing, and forecasting electricity consumption that operates at the local level. The components, architectural diagram, and source code of the system are presented.

KEYWORDS: INTERNET OF THINGS, MODEL, FORECASTING, SYSTEM, LOCAL LEVEL, EDGE COMPUTING, SINGLE-BOARD COMPUTER, ELECTRICITY.

ЗМІСТ

ВСТУП	10
1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ІНТЕРНЕТУ РЕЧЕЙ ТА МЕТОДОЛОГІЯ ПРОГНОЗУВАННЯ	13
1.1 Інтернет речей (IoT): концепції, архітектура та сучасні тенденції	13
1.1.1 Поняття та еволюція IoT	13
1.1.2 Основні рівні архітектури IoT-систем: сенсорний рівень, мережевий рівень, рівень обробки даних	17
1.1.3 Стандарти та протоколи зв'язку	19
1.1.4 Світові тренди: децентралізація, edge computing, енергоефективність	20
1.2 Edge computing як метод підвищення ефективності обробки IoT-даних	22
1.2.1 Сутність edge computing та відмінність від хмарних обчислень	22
1.2.2 Переваги та обмеження edge-підходу	22
1.2.3 Використання одноплатних комп'ютерів на edge рівні	25
1.2.4 Приклади застосування edge-обчислень у сфері енергоменеджменту	26
1.3 Методи аналізу та прогнозування часових рядів	28
1.3.1 Огляд моделей прогнозування	28
1.3.2 Переваги та недоліки моделей	30
2. ДОСЛІДЖЕННЯ СТАНУ ЕНЕРГЕТИЧНОГО СЕКТОРУ ТА МОДЕЛЕЙ ПРОГНОЗУВАННЯ СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ	33
2.1 Аналіз сучасного стану енергоспоживання	33
2.1.1 Загальні тенденції в Україні та світі	34
2.1.2 Динаміка споживання електроенергії за останні роки	35
2.1.3 Вплив різних чинників на споживання електроенергії	36
2.2 Аналіз існуючих технічних рішень моніторингу енергоспоживання ..	40
2.2.1 Огляд популярних комерційних та open-source систем	40
2.2.2 Виявлення переваг, недоліків та обмежень існуючих систем	42
2.2.3 Огляд необхідності існування локальних рішень	44
2.3 Аналітична оцінка моделей для прогнозування споживання електроенергії	46
2.3.1 Побудова прогнозів на короткостроковий період	47
2.3.2 Порівняння точності моделей	48
2.3.3 Обґрунтування доцільності використання прогнозів на edge-	

пристроях	49
3. РОЗРОБКА ІОТ-СИСТЕМИ ЗБОРУ, АНАЛІЗУ ТА ПРОГНОЗУВАННЯ СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ	51
3.1 Визначення ключових проблем і формування вимог до майбутньої ІоТ-системи	51
3.1.1 Виявлені недоліки існуючих підходів	51
3.1.2 Потреба у локальному опрацюванні даних	52
3.1.3 Формування технічних і функціональних вимог до системи	52
3.2 Архітектура ІоТ-системи	53
3.2.1 Вибір апаратних компонентів	53
3.2.2 Вибір програмних засобів	56
3.2.3 Методика збору, групування та обробки даних	58
3.2.4 Метод прогнозування	59
3.2.5 Архітектурна схема ІоТ-системи	59
3.3 Розробка системи прогнозування	61
3.3.1 Алгоритми збору, запису та візуалізації даних	62
3.3.2 Принцип роботи локальної моделі прогнозування	62
3.3.3 Локальний веб-інтерфейс	63
3.3.4 Автоматизація роботи системи	64
3.4 Готова система прогнозування та шляхи її реалізації	66
3.4.1 Можливості, переваги та недоліки готового рішення	67
3.4.2 Отримання вигоди завдяки використанню системи	68
3.4.3 Перспективи для використання та масштабування рішення	69
ВИСНОВКИ	71
ПЕРЕЛІК ПОСИЛАНЬ	74
ДОДАТКИ	78

ВСТУП

Актуальність теми. У сучасному світі в останній час зростає попит на електроенергію. Інтеграція відновлюваних джерел енергії та необхідність підвищення енергоефективності ставлять перед людьми завдання точного моніторингу, аналізу та прогнозування енергоспоживання. Україна що переживає періоди нестабільності енергопостачання та активну модернізацію інфраструктури, проблеми прогнозування і оперативного реагування набувають особливого значення. Вони пов'язані з підвищенням надійності постачання, оптимізацією тарифів та зниженням витрат на енергоресурси.

Сучасні IoT-технології дозволяють організувати збір даних з приладів обліку та сенсорів, проте традиційні хмарні підходи іноді не забезпечують достатньої швидкодії, приватності та економії трафіку. Поєднання інтернету речей із підходом edge computing, коли частина обробки і аналітики виконується локально на периферійних пристроях - відкриває можливості для оперативного локального прогнозування і прийняття рішень, зменшуючи затримки та навантаження на мережу. Цей підхід описаний у численних сучасних дослідженнях як перспективний напрямок для енергетичного сектору.

У науковій літературі та прикладних розробках виділяють кілька напрямків: централізовані системи прогнозування, розподілені IoT-мережі з відправкою даних у хмару та edge-орієнтовані рішення, що виконують попередню обробку або повний аналіз локально.

Огляди існуючих систем також вказують на такі проблеми як: залежність від доступності хмарних сервісів, питання безпеки IoT-пристроїв, а також недостатня увага до локальної оптимізації енергоспоживання у побутових і малих комерційних сценаріях. Звітні приклади практичних проектів демонструють набір типових методів (часові ряди, ARIMA, ковзні середні, прості регресійні моделі), які можна адаптувати для edge-пристроїв з обмеженими ресурсами.

Проблемні питання, що залишаються невирішеними:

- як забезпечити задовільну точність прогнозу на периферійних пристроях з

обмеженими обчислювальними ресурсами;

- які архітектурні рішення оптимальні для різних сценаріїв;
- як поєднати оперативність локальної аналітики з можливістю масштабного централізованого аналізу при необхідності.

Мета роботи - Розробити і обґрунтувати архітектуру IoT-системи для збору, аналізу та прогнозування споживання електроенергії із застосуванням edge computing, яка забезпечить локальну обробку даних, адекватну точність короткострокових, середньострокових прогнозів та можливість масштабування для застосування в побутових або малих комерційних середовищах.

Для досягнення мети, у магістерській роботі успішно виконано наступні задачі:

1. Аналіз проблеми збору та прогнозування енергоспоживання, доступні апаратні й програмні платформи для edge-обчислень.
2. Огляд і порівняльний аналіз методів прогнозування часових рядів, придатних для виконання на edge-пристроях (ковзні середні, ARIMA, прості регресійні моделі, легкі ML-моделі).
3. Розробка архітектури IoT-системи на базі одноплатного комп'ютера для збору й локальної обробки даних з приладів обліку.
4. Реалізація прототипу збору даних, локального збереження й базового модуля прогнозування.
5. Проведення тестування продуктивності й точності прогнозування на edge-пристрої та порівняти з варіантом обробки у хмарі.
6. Розробка рекомендації щодо практичного впровадження та масштабування рішення.

Об'єкт дослідження - процеси збору, обробки та прогнозування даних про споживання електроенергії з використанням IoT-пристроїв.

Предмет дослідження - архітектурні і алгоритмічні підходи до локальної обробки та прогнозування енергоспоживання, а також методи апробації їх ефективності на одноплатних комп'ютерах.

Методи дослідження. У роботі застосовано комплекс методів: аналіз літературних джерел і існуючих рішень, методи обробки часових рядів (ковзне

середнє, сезонні декомпозиції, ARIMA - для порівняння), методи оцінки якості прогнозу (MAE, RMSE, MAPE), експериментальна апробація на прототипі з подальшим аналізом продуктивності й затримок та інженерні методи проектування IoT-архітектури.

Наукова новизна одержаних результатів. У ході роботи описано новий підхід до локального прогнозування енергоспоживання на edge-пристроях: поєднання моделей прогнозування, оптимізованих під ресурси одноплатних комп'ютерів з архітектурними рішеннями для апаратної обробки з можливістю перетворення в гібридну. Таким чином показано можливість досягнення прийнятної точності прогнозу при зниженні затримок і трафіку порівняно з рішеннями орієнтованими на хмарні обчислення. Окремим внеском є розробка практичних рекомендацій щодо розгортання та масштабування такої системи для умов Українського енергоринку.

Практична значущість одержаних результатів. Розроблена система є основою для створення доступних систем моніторингу та прогнозування енергоспоживання в житловому фонді, навчальних закладах, малих офісах і підприємствах малого та середнього бізнесу. Локальні обчислення дозволять зменшити залежність від постійного доступу до хмарних сервісів, підвищити оперативність прийняття рішень та зекономити мережеві ресурси, що важливо в умовах обмеженого або нестабільного інтернет-з'єднання.

Апробація результатів магістерської роботи.

1. VII ВСЕУКРАЇНСЬКА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ «TELECOMMUNICATION: PROBLEMS AND INNOVATION», 30 грудня 2024 року, ДУІКТ - «Розробка енергоефективних алгоритмів для оптимізації навантаження у мультихмарних середовищах»

1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ІНТЕРНЕТУ РЕЧЕЙ ТА МЕТОДОЛОГІЯ ПРОГНОЗУВАННЯ

На даний момент напрямок іот є стрімко розвиваючимся. Кожного дня придумують нові і нові гаджети, що спрямовані на інтеграцію в екосистему інтернету речей. Користувачам подобається коли техніка може взаємодіяти між собою тим самим полегшуючи і покращуючи їх досвід використання, а компаніям подобається що користувачі мотивовані цим будуть купувати більше їх продукції. Для того щоб зрозуміти чому так сталося потрібно досконало проаналізувати поняття та структуру іот, та зрозуміти що передувало такому успіху.

1.1 Інтернет речей (IoT): концепції, архітектура та сучасні тенденції

1.1.1 Поняття та еволюція IoT

Інтернет речей, або IoT, - це мережа взаємопов'язаних пристроїв, які можуть підключатися та обмінюватися даними з іншими пристроями інтернету речей та хмарою. Пристроєм IoT притаманно оснащуватись технологічними пристроями, такими як датчики та програмним забезпечення. Вони також можуть включати механічні, цифрові елементи та побутові речі (Рис. 1.1).

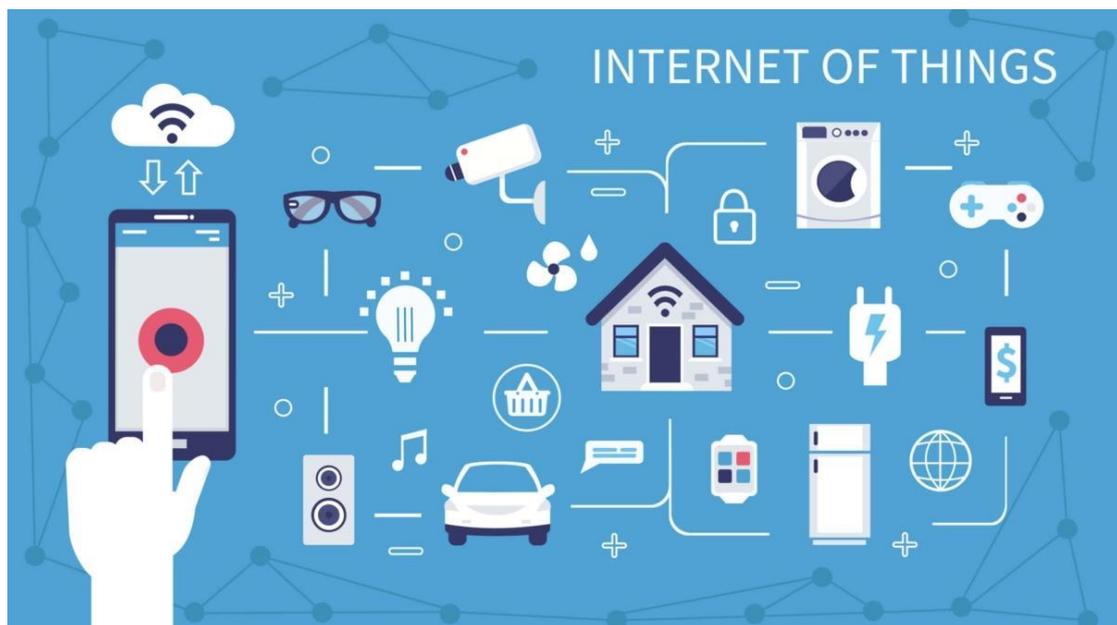


Рис. 1.1 Взаємозв'язок між складовими інтернету речей

Такі пристрої можуть охоплювати все, від повсякденних предметів до складних промислових інструментів. Останнім часом збільшилось використання IoT для підвищення ефективності роботи, поліпшення обслуговування клієнтів, вдосконалення процесу прийняття рішень та збільшення вартості бізнесу. Завдяки IoT дані передаються через мережу та не потребують взаємодії між людьми або між людиною і комп'ютером.

Об'єктом в Інтернеті речей може виступати людина з імплантованим кардіостимулятором, автомобіль із вбудованими датчиками що зчитують тиск у шинах або будь-який інший природний чи штучний об'єкт, який може мати IP-адресу і який за допомогою цього здатний передавати дані через мережу.

IoT охоплює електроніку, комунікації та комп'ютерну інженерію. Термін «Інтернет речей» вважається неправильним, оскільки пристрої не обов'язково мають бути підключені до загальнодоступного Інтернету. Вони повинні бути підключені лише до мережі та мати індивідуальну IP-адресу.

Хоча на споживчому ринку технологія IoT є синонімом продуктів для «розумного будинку», включаючи пристрої такі як термостати та розумне освітлення, найбільше застосування ця технологія знаходить у бізнесі та промисловому секторах. Відстеження комерційних активів та управління флотом є найбільшим окремим застосуванням IoT, що на момент 2024 року становить 22% від загального ринку (Рис. 1.2), зумовленим необхідністю моніторингу мобільних активів, таких як транспортні засоби та контейнери для транспортування товарів.

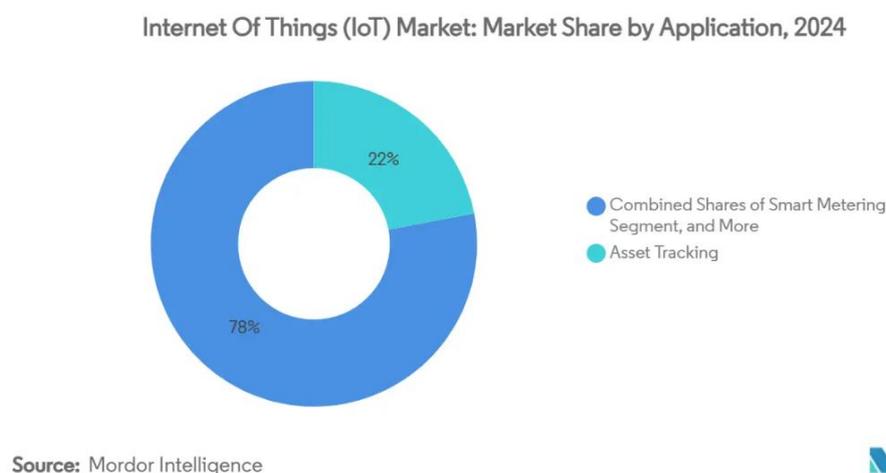


Рис. 1.2 Діаграма розподілення ринку Iot

Інші основні сфери застосування включають промисловий моніторинг, інтелектуальні лічильники в комунальних службах та підключену систему охорони здоров'я.

Однак існує низка занепокоєнь щодо ризиків, пов'язаних із зростанням і поширенням технологій та продуктів IoT, особливо в сферах конфіденційності та безпеки. Як наслідок, низка галузей, технологічних компаній та урядів багатьох країн вжили численних заходів та впровадили різноманітні запобіжні заходи для адекватного вирішення цих проблем і мінімізації ризиків безпеки, включаючи розробку та впровадження міжнародних і місцевих стандартів, керівних принципів та нормативних рамок. Через свою взаємопов'язаність пристрої Інтернету речей вразливі до атак на безпеку та порушень конфіденційності. Водночас спосіб бездротового зв'язку цих пристроїв створює регуляторні неоднозначності, ускладнюючи визначення меж юрисдикції щодо передачі даних.

Поштовхом до створення інтернету речей таким яким ми його знаємо сьогодні можна вважати події 1972 року, що відбулись у Стенфордській лабораторії штучного інтелекту. Там розробили комп'ютерний торговий автомат, адаптований з автомата орендованого у компанії Canteen Vending, який продавав товари за готівку або через комп'ютерний термінал. Серед його продуктів були пиво, йогурт і молоко. Оновлена версія апарату і досі знаходиться на території, але вже з оновленим програмною та апаратною частинами.

Однак саме за початок створення IoT галузі прийнято брати 1982 рік, коли була розроблена рання концепція мережевого підключеного інтелектуального пристрою як інтернет-інтерфейсу для датчиків на автоматі з продажу Coca-Cola. Апарат був встановлений у відділенні комп'ютерних наук Університету Карнегі-Меллона, де добровольці студенти та аспіранти могли відстежувати модель температури та стан запасів. Надихались вони вже раніше згаданим апаратом що стояв в Стенфордській лабораторії штучного інтелекту. Популярності він набрав саме через те що був першим пристроєм, підключеним до ARPANET.

Сучасне бачення Інтернету речей сформували стаття Марка Вейзера 1991

року про повсякденну обчислювальну техніку «Комп'ютер 21 століття», а також академічні заходи, такі як UbiComp і PerCom. В 1994 році Резою Раджі була описана концепція в IEEE Spectrum як «переміщення невеликих пакетів даних до великого набору вузлів з метою інтеграції та автоматизації всього, від побутової техніки до цілих заводів».

У період з 1993 по 1997 рік кілька компаній запропонували свої рішення, такі як Microsoft at Work або Novell NEST. Ця галузь набула імпульсу, коли Білл Джой уявив собі комунікацію між пристроями як частину своєї концепції «Шість веб-мереж», яка була представлена на Всесвітньому економічному форумі в Давосі в 1999 році.

Вперше ж поняття «Інтернет речей» і сам термін фігурували у виступі Пітера Т. Льюїса на 15-му щорічному законодавчому вікенді Фонду Конгресу чорношкірих у Вашингтоні, що була опублікована у вересні 1985 року. За словами Льюїса, «Інтернет речей, або IoT, - це інтеграція людей, процесів і технологій з підключеними пристроями та датчиками, що дозволяє здійснювати дистанційний моніторинг, контроль стану та управління такими пристроями».

Термін «Інтернет речей» був незалежно введений Кевіном Ештоном з компанії Procter & Gamble, а пізніше з Центру автоматичної ідентифікації Массачусетського технологічного інституту, в 1999 році незважаючи на те що він віддавав перевагу виразу «Інтернет для речей». На той момент він вважав радіочастотну ідентифікацію RFID - важливою складовою Інтернету речей, оскільки вона ефективно дозволяла б комп'ютерам керувати всіма окремими речами. Основною визначальною характеристикою Інтернету речей вважалася його здатність вбудовувати мобільні приймачі-передавачі короткого діапазону в різні гаджети та предмети щоденного вжитку, що дозволяло б створювати нові форми комунікації між людьми та речами, а також між речами безпосередньо.

У 2004 році Корнеліус «Піт» Пітерсон, генеральний директор NetSilicon, передбачив, що «наступна ера інформаційних технологій буде домінувати пристроями IoT, а мережеві пристрої в кінцевому підсумку набудуть такої популярності та значущості, що значно перевищать кількість мережевих комп'ютерів та робочих станцій». Пітерсон вважав, що медичні пристрої та

промислові системи управління стануть домінуючими сферами застосування цієї технології.

Визначаючи Інтернет речей як «просто момент часу, коли до Інтернету було підключено більше речей або об'єктів, ніж людей, компанія Cisco Systems оцінила, що IoT «народився» між 2008 і 2009 роками, причому співвідношення речей до людей зросло з 0,08 у 2003 році до 1,84 у 2010 році.

1.1.2 Основні рівні архітектури IoT-систем: сенсорний рівень, мережевий рівень, рівень обробки даних

Для того щоб зрозуміти, як розумні пристрої спілкуються та функціонують разом, необхідно ознайомитися з архітектурою Інтернету речей. Вона визначає, як датчики, мережі та хмарні системи взаємодіють між собою для ефективного збору, обробки та обміну даними.

Архітектура IoT є основою для всіх підключених пристроїв і додатків. Кожен рівень має свою роль, від зчитування даних реального світу до виконання інтелектуальних дій. Чітка структура забезпечує масштабованість, безпеку та безперебійну комунікацію між пристроями. Ця архітектура є основою систем IoT, забезпечуючи роботу всього, від розумних будинків до промислової автоматизації.

Архітектура Інтернету речей складається з чотирьох різних рівнів, а саме: сенсорного рівня, мережевого рівня, рівня обробки даних та рівня додатків (Рис 1.3).

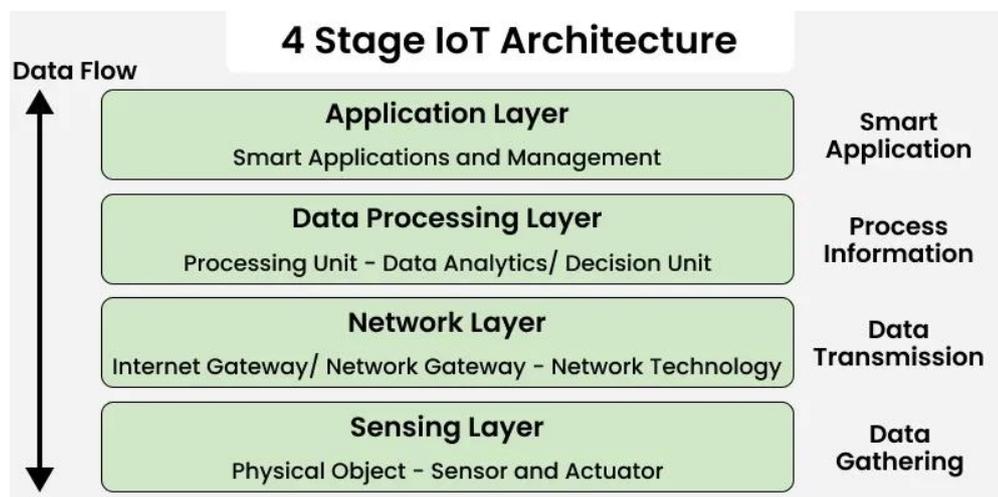


Рис 1.3 4 Ступені архітектури інтернету речей

Сенсорний рівень - Це найнижчий рівень, відповідальний за виявлення фізичних умов навколишнього середовища. Відповідає за такі функції як: збір необроблених даних, такі як температура, вологість, рух, звук або тиск. Виявляє зміни в навколишньому середовищі за допомогою вбудованих компонентів. При необхідності ініціює дії за допомогою виконавчих механізмів. Компоненти які найчастіше використовуються на цьому рівні це датчики, такі як датчики вологості, газу, інфрачервоного випромінювання, ультразвукові. Приводи, такі як двигуни, перемикачі, клапани. Мікроконтролери та RFID-мітки. Також цей рівень відповідальний за передачу даних, що зчитуються, до мережевого рівня через дротові або бездротові канали зв'язку.

Мережевий рівень - забезпечує зв'язок і комунікацію між пристроями IoT і хмарними системами. Мережевий рівень виконує такі функції як: безпечна передача зібраних даних з датчиків на платформи обробки, підтримка комунікації між пристроями і серверами та між пристроями безпосередньо, обробка адресація, маршрутизація і пересилання даних. Також рівень використовує технології зв'язку, такі як: Wi-Fi, Bluetooth, Zigbee, LoRaWAN Мережі Ethernet і супутникові мережі та маршрутизатори і комутатори.

Рівень обробки даних - аналізує, фільтрує та інтерпретує дані, отримані від мережевих пристроїв. Рівень відповідає за очищення та форматування необроблених даних з датчиків для отримання значущої інформації. Застосовує аналітику для виявлення закономірностей або відхилень. Зберігає дані для історичного аналізу або звітності. Для обробки даних рівень використовує Хмарні платформи IoT, масиви та сховища даних та механізми потокової обробки і машинного навчання. На виході отриманий результат буде виражатись у прогностичних сповіщеннях та звітах, сигналах виявлення відхилень, та необхідних правилах прийняття рішень для автоматизації.

Рівень додатків - Цей найвищий рівень безпосередньо взаємодіє з кінцевими користувачами та бізнес-системами. Функції що він виконує такі: надає інтерфейси для віддаленого моніторингу та керування пристроями IoT, відображає візуальну аналітику за допомогою інформаційних панелей та діаграм, запускає автоматизовані дії на основі оброблених даних. Компоненти

що використовують на цьому рівні: мобільні додатки, веб-панелі, портали і інструменти візуалізації та оповіщення. Рівень додатків надає можливість до віддаленого керування пристроями, моніторингу стану в режимі реального часу, а також інтеграцію з корпоративними додатками.

1.1.3 Стандарти та протоколи зв'язку

Стандарти та протоколи комунікації IoT забезпечують обмін даними між пристроями з урахуванням вимог до низького енергоспоживання, надійності та роботи в умовах обмежених ресурсів (Рис. 1.4). До них належать як мережеві протоколи, що працюють на різних рівнях так і протоколи додатків, що контролюють прямий обмін даними.

До стандартів комунікації iot можна віднести такі основні як:

- IEEE 802.15.4 - фізичний рівень і MAC для малопотужних бездротових мереж.
- IEEE 802.15.1 - основа класичного Bluetooth.
- IEEE 802.11 - правила роботи Wi-Fi на фізичному та каналному рівнях.
- NB-IoT 3GPP Release 13/14 - стільниковий стандарт вузько смугового IoT.
- LTE-M (Cat-M1) - стільниковий стандарт енергоефективного IoT.
- OPC UA Standard - між промисловий стандарт взаємодії в автоматизації.

До протоколів же відносять такі:

- MQTT - спрощений мережевий протокол, що працює на TCP/IP.
- Zigbee - специфікація на основі стандарту IEEE 802.15.4 для набору високорівневих протоколів зв'язку.
- Z-Wave - протокол бездротового зв'язку, який використовується переважно для домашньої автоматизації.
- HTTP/HTTPS - протокол передачі даних, що використовується в комп'ютерних мережах.

Основна відмінність протоколів від стандартів в полягає в тому, що стандарти чітко визначають загальні технічні вимоги до IoT-систем, тоді як визначають конкретні правила передачі даних між пристроями.

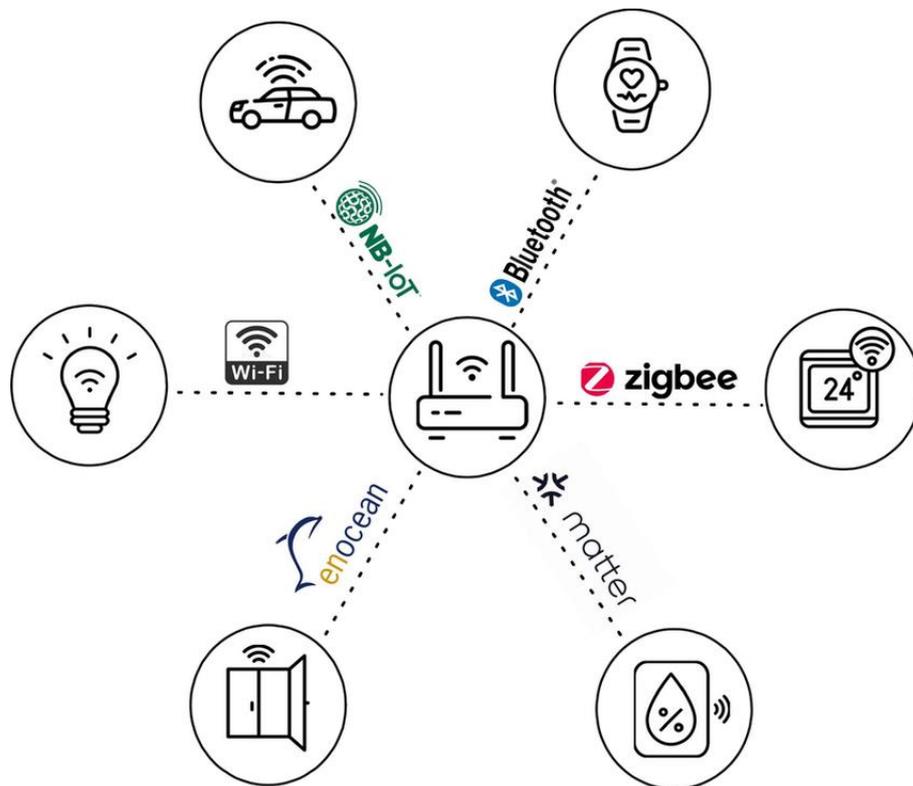


Рис 1.4 Обмін з різними пристроями за допомогою протоколів

1.1.4 Світові тренди: децентралізація, edge computing, енергоефективність

Тенденція децентралізації в IoT - це один із найважливіших трендів сьогоденного ринку. Однією з основних причин такого руху виділяють появу Блокчейну.

Блокчейн - це розподілений реєстр із зростаючим списком записів (блоків), які надійно пов'язані між собою за допомогою криптографічних хеш-функцій.

Децентралізація означає що замість централізованого сервера, який керує всіма пристроями, блокчейн дозволяє IoT-пристроєм безпосередньо взаємодіяти один з одним. Кожна взаємодія, транзакція або дані записуються в незмінний реєстр, який неможливо змінити заднім числом. Це підвищує цілісність даних. Пристрої отримують власні децентралізовані ідентичності, аутентифікуючись через криптографічні ключі. Це знижує залежність від центрального авторитету. Смарт-контракти можуть автоматизувати дії: наприклад, виконувати оплату або виділяти ресурси, коли певні умови виконуються.

Децентралізація в IoT - це не просто технологічна мода, а реальний напрям

розвитку, який може змінити архітектуру мереж речей. Поєднання блокчейну з IoT дає значні переваги у безпеці, масштабованості та автоматизації, водночас створює нові бізнес-моделі. Але ця трансформація має свої виклики: технічні, регуляторні, інтеграційні. Для можливості широкого впровадження їх потрібно вирішувати вже зараз.

Сьогодні також вагомою проблемою у сфері Iot виділяють проблему надмірного енергоспоживання і пропонують стратегії для її зменшення.

Основними методами оптимізації спільнота виділяє такі:

Розумні алгоритми маршрутизації та кластеризації - використання динамічних стратегій допомагає мінімізувати енергозатрати при передачі даних між вузлами.

Адаптивне розподілення ресурсів - через планування задач, балансування навантаження між пристроями, а також врахування їх енергетичного стану.

Легковагові методи безпеки - важливо створювати безпечні протоколи та механізми аутентифікації, які не споживають багато енергії, щоб не знижувати ефективність мережі.

Інтелектуальні рішення - застосування нечіткої логіки, біо-натхненних алгоритмів та машинного навчання для прийняття рішень щодо того, коли і як пристрої повинні передавати дані чи переходити в режим сну.

Гібридне живлення - поєднання батарей з відновлюваними джерелами таких як сонячна енергія - для продовження життя пристроїв і зменшення необхідності частого обслуговування.

Енергоефективне розміщення сервісів - використання fog і edge-комп'ютингу для обробки даних ближче до пристроїв, що знижує потребу передавати великі обсяги інформації в хмару і економить енергію.

Тенденція оптимізації енергоспоживання в IoT зараз є критичною. З огляду на масове розгортання сенсорних мереж, важливо забезпечити їхню стійкість і автономність. Застосування адаптивних алгоритмів, інтелектуальних систем керування а також більш розподілених обчислень таких як edge compute, дає змогу значно зменшити енергетичні витрати без втрати продуктивності.

1.2 Edge computing як метод підвищення ефективності обробки IoT-даних

Edge computing - це розподілена обчислювальна платформа, яка наближає корпоративні додатки до джерел даних, таких як пристрої IoT або локальні периферійні сервери. Така близьке розташування до даних у їхньому джерелі може забезпечити значні переваги для бізнесу, зокрема швидше отримання аналітичної інформації, покращення часу відгуку та кращу доступність пропускну здатності.

1.2.1 Сутність edge computing та відмінність від хмарних обчислень

Edge Computing та Cloud Computing - це дві парадигми сучасного цифрового світу. Обидві парадигми набувають все більшого поширення у сфері зберігання даних у хмарі.

Edge Computing - це розподілена обчислювальна архітектура, яка наближає обчислення та зберігання даних до джерела даних. Обробка інформації відбувається на периферії мережі, поруч із пристроєм, який генерував дані, а не в центральному сервері, такому як центр обробки даних. Зменшення затримки та потреб у пропускну здатності є бажаними результатами edge computing під час передачі великих обсягів даних до центру обробки. Edge computing полегшує прийняття рішень у режимі реального часу, обробляючи дані локально та прискорюючи передачу даних до хмари та з неї.

Пристрої IoT, автоматизована техніка та системи доповненої та віртуальної реальності потребують переваг низької затримки, особливо від edge computing. Додатки що використовуються в промисловості, аналізі відео та зображень, а також інтелектуальні також генерують великі обсяги даних, а тому цей метод обробки робить цей підхід неможливістю - а необхідністю.

1.2.2 Переваги та обмеження edge-підходу

Edge Computing як сучасний метод обробки інформації локально, має свої переваги та недоліки перед відмінними технологіями обробки.

До переваг Edge Computing можна віднести такі:

1. Зменшена затримка - Edge Computing дозволяє швидше обробляти та аналізувати дані в місці їх виникнення, що скорочує час, необхідний для передачі даних до хмари та назад. Завдяки значному зменшенню затримки, це ідеально підходить для додатків що вимагають прийняття рішень у реальному часі таких як робототехніка, промислова автоматизація та автомобілі що керуються автопілотом.
2. Підвищення безпеки - завдяки обробці та аналізу даних поблизу місця їх появи та зменшенню обсягу даних, які необхідно передавати в хмару, периферійні обчислення можуть підвищити рівень безпеки. В результаті хакерам стає складніше зламати систему, оскільки зменшується можливість для атаки та потенційні вразливі місця.
3. Більша ефективність пропускної здатності - завдяки можливості локальної обробки та аналізу даних, периферійні обчислення можуть зменшити обсяг даних, які необхідно передавати до хмари. В результаті краща ефективність пропускної здатності може знизити витрати на передачу даних та забезпечити швидшу обробку.

До недоліків Edge Computing можна віднести такі:

1. Низька обчислювальна потужність - порівняно з хмарними обчисленнями, периферійні пристрої часто мають меншу обчислювальну потужність і обсяг пам'яті. Як наслідок, типи додатків, які можна використовувати локально на таких пристроях, можуть бути обмежені.
2. Підвищена складність - впровадження периферійних обчислень може бути складнішим, ніж стандартні стратегії хмарних обчислень. Це пов'язано з вимогами Edge Computing, які можуть бути складними в управлінні та обслуговуванні, щодо встановлення ресурсів обробки та зберігання даних ближче до джерела.
3. Збільшення витрат - Edge Computing може бути дорожчим, ніж хмарні обчислення. Це пов'язано з тим, що Edge computing вимагає розгортання ресурсів обробки та зберігання даних на декількох серверах, що може бути дорожчим у налаштуванні та обслуговуванні.

Для того щоб розуміти різницю між підходами для обробки даних, а саме Edge та Cloud Computing, потрібно порівняти їх параметри (Таблиця 1.1).

Таблиця 1.1 Головні відмінності по параметрам Edge та Cloud Computing

Параметр	Edge Computing	Cloud Computing
Визначення	Edge Computing - це розподілена обчислювальна архітектура, яка наближає обчислення та зберігання даних до джерела даних.	Cloud Computing - це модель надання інформаційних технологій через Інтернет.
Місце обробки	Аналіз та обробка здійснюється на периферії мережі, поблизу пристрою, що генерує дані.	Аналіз та обробка даних здійснюються в центральному місці, наприклад, у центрі обробки даних.
Вимоги до пропускну здатності	Необхідна низька пропускну здатність, оскільки дані обробляються поблизу джерела.	Необхідна більша пропускну здатність оскільки дані повинні передаватися через мережу до центрального місця для.
Витрати	Edge Computing є більш дорогим, оскільки на периферії може бути необхідне спеціалізоване обладнання та програмне забезпечення.	Cloud Computing є менш витратним, оскільки користувачі платять лише за ті ресурси, які вони фактично використовують.
Масштабованість	Для Edge Computing може бути більш складною, оскільки на периферії може знадобитися додаткові обчислювальні ресурси.	Для Cloud Computing Простіше, оскільки користувачі можуть швидко та легко збільшувати або зменшувати обсяг обчислювальних ресурсів відповідно до своїх потреб.
Приклади використання	Додатки, що вимагають низької затримки та прийняття рішень у режимі реального часу, такі як пристрої IoT, транспортні засоби з автопілотом та системи AR/VR.	Додатки, які не мають суворих вимог до затримки, такі як веб-додатки, електронна пошта та зберігання файлів.
Безпека даних	Безпека даних може бути покращена, оскільки дані обробляються поблизу джерела і не передаються по мережі.	Безпека даних є більш складною задачею, оскільки дані передаються через мережу до центрального місця для обробки.

1.2.3 Використання одноплатних комп'ютерів на edge рівні

Одноплатні комп'ютери дають широкий простір для реалізації Edge Computing завдяки поєднанню низького енергоспоживання, достатньої обчислювальної потужності та компактності. У таких пристроях усі основні компоненти розміщені на одній платі, що робить їх дешевими, простими у розгортанні та придатними для розміщення безпосередньо біля джерел даних у системах IoT. Використання SBC дає змогу переносити значну частину обробки даних із хмарних серверів на локальний рівень, що зменшує затримки, знижує навантаження на мережу й підвищує рівень безпеки, оскільки дані не потребують постійної передачі до зовнішніх сервісів.

Edge-вузли на базі SBC здатні виконувати попередню обробку, фільтрацію та аналіз інформації в реальному часі. Це забезпечує стабільну роботу навіть за нестійкого зв'язку та дає можливість передавати лише необхідні результати, а не повні сирі потоки даних. Такий підхід зменшує мережевий трафік і підвищує ефективність системи загалом. Завдяки малому енергоспоживанню SBC є економічним рішенням у розгортанні масштабних систем, а можливість швидкого додавання пристроїв робить інфраструктуру легко масштабованою.

Застосування SBC у межах edge computing охоплює промислові IoT-системи, домашню та офісну автоматизацію, локальні сервери, вузли збору та аналізу даних, а також моделі машинного навчання. У цих сценаріях SBC забезпечують достатню продуктивність для обробки інформації від сенсорів, камер або роботизованих систем без потреби постійного доступу до хмари. Водночас їхні апаратні обмеження визначають сферу ефективного використання: для задач, що потребують високої продуктивності або великих моделей, може знадобитися кластеризація або поєднання локальних вузлів із хмарними ресурсами.

Таким чином, одноплатні комп'ютери формують надійну основу для edge-computing, забезпечуючи локальну обробку даних, зменшення затримок і підвищення енергоефективності, що є ключовими вимогами сучасних IoT-інфраструктур.

1.2.4 Приклади застосування edge-обчислень у сфері енергоменеджменту

Окрім Edge Computing, енергетичний сектор переживає глибокі перетворення внаслідок декількох взаємопов'язаних явищ: децентралізації виробництва електроенергії, появи просumerів, які виробляють і споживають енергію, а також нових гравців на стороні попиту та все більшого використання відновлюваних джерел енергії.

Ці перетворення породжують нові моделі відносин між постачальниками послуг та кінцевими споживачами, такі як поява енергетичних спільнот або ринків гнучкості попиту і змушують компанії в цьому секторі здійснювати серйозні зміни.

Серед цих значних змін - цифровізація процесів збору інформації та навіть прогнозування майбутніх сценаріїв, що передбачає необхідність збирати великі обсяги даних, аналізувати їх та розуміти, які рішення потрібно прийняти та які дії потрібно виконати.

Щоб зробити цю обробку більш ефективною, в галузі активно розвивається нова модель обчислень - Edge Computing. Вона передбачає доповнення обробки в централізованих хмарних інфраструктурах алгоритмами машинного навчання або штучного інтелекту, що виконуються на периферії мережі. Тобто на вузлах, які знаходяться ближче до місця збору даних.

Ця модель обчислень поширюється дуже швидко. Якщо в 2019 році Grand View Research оцінила вартість ринку Edge Computing в 3,5 мільярда доларів, то сьогодні та ж сама компанія стверджує, що ця цифра зросте до 43,4 мільярда доларів у 2027 році, з щорічним зростанням на 37,4%. І головна причина цієї тенденції полягає в тому, що вона дає три основні переваги для промислового сектора в порівнянні з централізацією обчислень в хмарі.

Edge Computing вже використовується в дуже конкретних випадках, які починають впроваджувати компанії в електро-енергетичному секторі

1. Edge Computing для віртуалізації трансформаторних підстанцій - Це, можливо, найважливіше застосування в цьому секторі. Центри перетворення електроенергії середньої та низької напруги це

інфраструктура, відповідальна за адаптацію електричної енергії для споживання громадянами в своїх домівках. Вони є частиною розподільної мережі, і в країні розміром з Іспанію їх налічується сотні тисяч. Ці центри перетворення мають низку промислового обладнання, цифровізація якого може надати надзвичайно цінну інформацію як операторам центрів, так і самим виробникам, або навіть кінцевим споживачам.

2. Edge Computing для виявлення шахрайства - являє собою розробку алгоритмів штучного інтелекту, здатних виявляти можливі випадки шахрайства у споживанні електроенергії в низьковольтній мережі. Споживання електроенергії через незаконні підключення до розподільчої мережі призводить до не запланованого збільшення попиту, що може спричинити великі збитки для розподільчих компаній через несправності та пожежі на їхніх трансформаторних станціях. Крім того, ці несправності часто завдають серйозної шкоди іншим громадянам, чиє енергопостачання в результаті цього порушується. Саме тому головним завданням розподільчих компаній є раннє виявлення такого роду порушень, що дозволяє їм захистити свої команди та розподільчі центри і не нехтувати своїми законними клієнтами.
3. Edge Computing для обслуговування розподіленого енергопостачання - кінцеві споживачі перетворюються на «просумерів» завдяки поширенню домашніх та промислових сонячних установок. У мережі з'являються локальні джерела енергії, акумулятори і зарядні станції, що робить низьковольтну інфраструктуру двонапрявленою замість традиційної односпрямованої. Паралельно формуються енергетичні спільноти та агрегатори попиту, у межах яких групи користувачів виробляють, розподіляють або продають енергію. Зростання кількості розподілених елементів вимагає ефективного керування їх роботою, тому компанії впроваджують рішення Edge Computing для локальної обробки даних і оптимізації розподілу енергії.
4. Edge Computing як шлях до сервітизації - останній випадок, який варто виділити, - це сервітизація. Концепція сервітизації передбачає зміну бізнес

моделі та майже культурну трансформацію компаній. Вона полягає у переході від моделі великих інвестицій в активи до нової, набагато гнучкішої моделі оплати послуг та використання продуктів. Ця нова парадигма відносин дозволяє, наприклад, виробникам електричних розподільних пристроїв та силової електроніки отримати нове джерело доходу з пропозицією високої доданої вартості для своїх клієнтів. Завдяки додаткам та алгоритмам Edge Computing вони надають інтелектуальні можливості своїм промисловим активам, що дозволяє їм надавати надзвичайно цінну інформацію для експлуатації заводів або об'єктів, якими мають керувати їхні клієнти, промислові та ті компанії, що експлуатують електромережі.

1.3 Методи аналізу та прогнозування часових рядів

У математиці часовий ряд - це послідовність точок даних, індексованих (або перерахованих, або зображених на графіку) у хронологічному порядку. Найчастіше часовий ряд - це послідовність, взята в послідовних рівновіддалених точках часу. Таким чином, це послідовність дискретних даних у часі. Прикладами часових рядів є висота океанських припливів, кількість сонячних плям і щоденне значення індексу Доу-Джонса на момент закриття біржі.

1.3.1 Огляд моделей прогнозування

Методи прогнозування часових рядів поділяються на дві великі, історично домінуючі групи: моделі, які представляють серійну кореляцію з авторегресійними та ковзними середніми термінами, на кшталт ARMA або ARIMA, та їх сезонні розширення, і методи згладжування/просторового стану, які фіксують рівень, тенденцію та сезонні компоненти за допомогою експоненціально зважених середніх таких як ETS. Моделі ARIMA пояснюють значення як лінійну функцію минулих значень та минулих помилок прогнозування - частина «I» (інтеграція) робить нестационарні ряди стаціонарними шляхом диференціювання, а сезонна ARIMA додає періодичне

диференціювання для сезонних циклів.

Методи експоненціального згладжування ETS оцінюють рівень, тренд та сезонні компоненти безпосередньо та поширюють їх вперед. У своїй формі простору станів вони дають точні ймовірності та інтервали прогнозування і є особливо ефективними, коли структура ряду добре описується комбінаціями адитивних та мультиплікативних рівнів/трендів/сезонності. ETS і ARIMA не суперечать одна одній з філософської точки зору - обидві можуть бути записані мовою простору станів і на практиці працюють однаково з багатьма наборами даних - але вони підкреслюють різні механізми (згладжувальні компоненти проти авторегресійної/середньозваженої залежності).

Емпіричні порівняння не виявляють універсального переможця: ARIMA часто переважає, коли домінують серійна кореляція та структура короткої пам'яті, а залишки наближаються до гаусівських, тоді як ETS виграє, коли сигнал є переважно плавною динамікою рівня/тенденції/сезону та важливою є швидка адаптація до останніх змін. Порівняльні дослідження та останні застосування включаючи прогнозування епідемій та екологічних явищ показують, що автоматичний вибір моделі (auto.arima, ets) плюс ретельна перехресна валідація зазвичай дають міцні базові показники, а гібридні або машинні підходи можуть поліпшити точність у складних, нелінійних або високочастотних задачах, але за рахунок інтерпретованості та іноді надійності.

Практичний вибір моделі залежить від питання короткостроковий чи довгостроковий горизонт даних таких як довжина, наявність і стабільність сезонності, розподіл шуму та операційних потреб наприклад: ймовірні інтервали, інтерпретованість, автоматизація. Якщо можливо, слід використовувати як добре налагоджену модель ARIMA/SARIMA, так і прогноз ETS/state-space, перевіряти їх за допомогою перехресної валідації часових рядів і віддати перевагу простішій моделі, яка відповідає вимогам точності та невизначеності. Комбінувати моделі треба тільки в тому випадку, якщо це послідовно зменшує похибку прогнозу на утриманих даних.

1.3.2 Переваги та недоліки моделей

Переваги та недоліки поширених моделей прогнозування часових рядів, зокрема моделей з родини авторегресивних інтегрованих ковзних середніх таких як ARIMA або ARMA та методів експоненціального згладжування ETS можна описати таким чином.

Переваги та недоліки моделей типу ARIMA та ARMA

Переваги: 1. ARIMA (або ARMA, коли дані є стаціонарними) може явно моделювати серійну кореляцію: прогноз базується на минулих значеннях (авторегресія) та на минулих помилках прогнозу (рухоме середнє).

2. Модель є гнучкою: змінюючи параметри (p , d , q), часто можна охопити широкий спектр динаміки - тенденція, повернення до середнього значення, циклічна поведінка та короткострокові залежності.
3. Статистична структура часто забезпечує формальну діагностику - після підгонки можна перевірити залишки на білизну (відсутність автокореляції), умови стаціонарності, адекватність припущень моделі - це допомагає підтвердити відповідність моделі.
4. Коли процес генерації даних є приблизно лінійним і добре апроксимується структурою ARMA (після диференціювання, якщо потрібно), ARIMA часто забезпечує високу ефективність прогнозування.

Недоліки: 1. Моделювання ARIMA вимагає вибору параметрів (порядки p , d , q) та перевірки припущень - це вимагає статистичних знань та ручної роботи.

2. Модель передбачає лінійні взаємозв'язки і може дати збій, якщо базовий процес є нелінійним, гетероскедастичним або демонструє структурні зміни чи зовнішні шоки - у такому випадку ARIMA може давати недостовірні прогнози.
3. Обчислювальне навантаження та людські зусилля: побудова добре специфікованої моделі ARIMA може бути трудомісткою, особливо для нестаціонарних або сезонних даних або при застосуванні сезонної ARIMA (SARIMA).
4. Для багатовимірних або систем рядів проста одновимірна ARIMA є

недостатньою. Потрібні більш складні моделі наприклад, VAR, динамічна регресія.

Переваги та недоліки моделей типу експоненціального згладжування та ETS-моделей

Переваги: 1. Методи ETS є концептуально простими та інтуїтивно зрозумілими.

Вони згладжують минулі спостереження, надаючи більшу вагу останнім даним, та поступово оновлюють оцінки рівня, тренду та сезонності.

2. Вони швидко адаптуються до мінливих моделей. Оскільки згладжування надає більшу вагу останнім даним, ETS може досить швидко реагувати на тренди або сезонні зміни (за умови, що структура залишається приблизно стабільною).
3. ETS може враховувати різні комбінації компонентів (тільки рівень, рівень та тенденція, сезонність, мультиплікативна проти адитивної сезонності) залежно від характеристик даних, що забезпечує гнучкість для моделювання різноманітних реальних часових рядів.
4. Ефективний з точки зору обчислень і простіший у застосуванні, ніж ARIMA, коли структура даних проста або коли потрібне швидке прогнозування, без необхідності важких циклів вибору моделі.

Недоліки: 1. ETS припускає, що майбутні моделі будуть схожими на минулі спостереження (тобто тенденція та сезонність залишаються стабільними). Модель може погано працювати в разі раптових структурних зривів, винятків або змін режиму.

2. ETS зазвичай не має явного моделювання автокореляції. Вона не відображає залежності минулих помилок, як це робить ARIMA. Коли базовий процес генерації даних має значні автокорельовані залишки, ETS може працювати неефективно.
3. Вибір моделі по параметрам, таким як: визначення адитивного чи мультиплікативного підходу, які компоненти включити - може вимагати знань у цій галузі або методу проб і помилок. Неправильна специфікація може погіршити точність прогнозу.
4. Для дуже складних часових рядів, таких як нелінійна динаміка, раптові

зміни, зовнішні шоки, множинні взаємодіючі ряди - ETS може бути недостатньо. Можуть знадобитися більш гнучкі і складні моделі наприклад гібридні або машинного навчання.

Вибір між ARIMA та ETS залежить від характеристик даних та цілей прогнозування. Якщо серія демонструє сильну автокореляційну структуру зі стаціонарною динамікою (або може бути зроблена стаціонарною), є приблизно лінійною, а також якщо важливими є статистична діагностика та інтерпретованість, ARIMA може бути кращим вибором. Якщо серія демонструє плавні тенденції та сезонність, а вам потрібен простий, швидкий і надійний метод особливо для короткострокового прогнозування або коли структура домену є чіткою - ETS буде більш підходящою.

Емпіричні дослідження та література з опитуваннями підкреслюють, що жоден з методів не є універсально кращим. Ефективність залежить від даних. Гібридні моделі, що поєднують переваги обох підходів, в деяких випадках можуть давати кращі прогнози, хоча й за рахунок складності.

Підсумовуючи - ARIMA є кращою, коли динаміка даних визначається серійними залежностями та стаціонарною поведінкою. ETS працює добре, коли домінують рівень, тенденція та сезонність і останні дані повинні мати більшу вагу. Вибір моделі повинен базуватися на діагностиці даних, горизонті прогнозування та стабільності сезонності/тенденції, а не на стандартних перевагах.

2. ДОСЛІДЖЕННЯ СТАНУ ЕНЕРГЕТИЧНОГО СЕКТОРУ ТА МОДЕЛЕЙ ПРОГНОЗУВАННЯ СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ

У 2025 році енергетичний сектор побачив значне зростання попиту на видобутку відновлюваної енергії, а саме сонячної та вітрової, яка в першій половині року вперше перевищила виробництво електроенергії з вугілля в усьому світі (Рис. 2.1). Також завдяки цій події значно стабілізувались викиди в навколишнє середовище. Основними тенденціями на даний момент є рекордне очікуване зростання обсягів та потужності відновлюваних джерел енергії, збільшення інвестицій у сонячну енергетику та акумуляторні батареї, а також відновлення інвестицій у ядерну енергетику та природний газ.

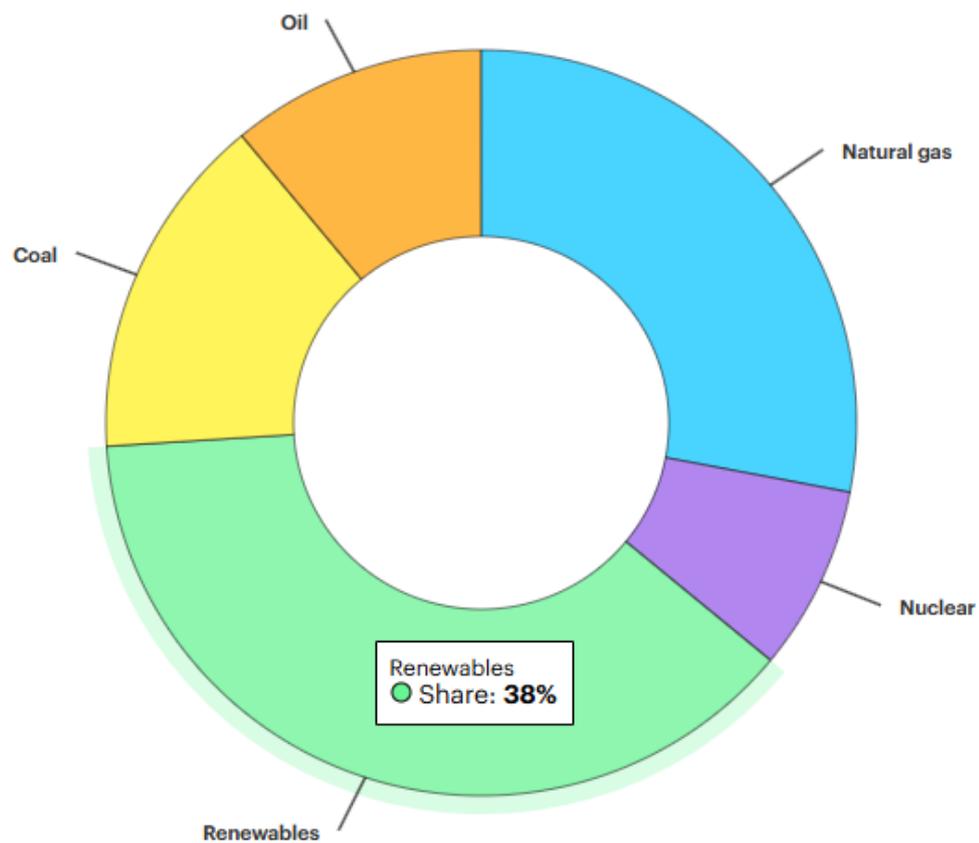


Рис 2.1 Частка зростання попиту на енергію з відновлювальних джерел

2.1 Аналіз сучасного стану енергоспоживання

Аналіз поточного енергоспоживання показує постійне зростання світового попиту, що в основному обумовлено споживанням електроенергії, а також виявляє значні регіональні розбіжності та зростання частки відновлюваних

джерел енергії, незважаючи на постійну домінанту викопних видів палива.

2.1.1 Загальні тенденції в Україні та світі

Останнім часом світове споживання електроенергії демонструє стійку тенденцію до зростання. За даними Міжнародного енергетичного агентства (МЕА), у 2024 році світовий попит на електроенергію зріс на 4,3% - це найвищий темп зростання за останні десятиліття, а абсолютне збільшення є найбільшим за всю історію, не враховуючи значне відновлення після рецесії.

Загальне світове споживання електроенергії у 2024 році перевищило 30 000 ТВт·год. Цей стрибок зумовлений структурними змінами: розширенням доступу до електроенергоємних приладів (наприклад, охолодження), електрифікацією опалення, транспорту та промисловості, розширенням центрів обробки даних, цифровізацією та зростанням популярності електромобілів і теплових насосів. Екстремальні погодні умови також відіграють свою роль - підвищення глобальної температури збільшило попит на охолодження, що значно сприяло стрибку попиту в 2024 році.

З боку пропозиції, виробництво енергії з низьким рівнем викидів вуглецю значно зросло. У 2024 році відновлювані джерела енергії в поєднанні з атомною енергією забезпечили понад 40 % світового споживання електроенергії - вперше з 1940-х років джерела енергії з низьким рівнем викидів вуглецю досягли такої частки. Зростання відновлюваних джерел енергії склало майже три чверті збільшення виробництва електроенергії, причому лише сонячна фотоелектрична енергія забезпечила приблизно 480-500 ТВт·год нової електроенергії.

Виробництво сонячної енергії подвоїлося за три роки. Вітроенергетика та гідроенергетика також зросли, хоча річний темп зростання вітроенергетики залишився більш помірним через високу базу та регіональні відмінності у швидкості вітру. Водночас зростання виробництва атомної енергії залишилося помірним порівняно з відновлюваними джерелами енергії.

Отже, світове споживання електроенергії швидко зростає під впливом структурної електрифікації та попиту, пов'язаного з кліматом, тоді як пропозиція все більше схиляється до низьковуглецевих джерел, особливо сонячної енергії.

У контексті України споживання електроенергії в останні роки зазнало значних коливань, що було зумовлено війною, що спричинила пошкодження інфраструктури та структурі зміни. У 2023 році споживання електроенергії в Україні становило близько 89,40 млрд кВт·год, а виробництво електроенергії - близько 102,6 млрд кВт·год.

Війна та пов'язані з нею пошкодження енергетичної інфраструктури спричинили різке падіння споживання електроенергії на душу населення, а також загальне споживання енергії у 2022 та 2023 роках значно знизилося порівняно з до воєнним станом.

У 2024 році Україна, імпортувала рекордний обсяг електроенергії, що свідчить про навантаження на внутрішні потужності з постачання. Водночас Україна має великий потенціал для розширення виробництва електроенергії з відновлюваних джерел. За останніми дослідженнями, лише сонячні фотоелектричні системи на дахах будівель можуть забезпечити до 290 ТВт·год на рік, якщо їх повністю задіяти - це значно перевищує поточне споживання електроенергії в країні.

Це вказує на шлях до відновлення та підвищення енергетичної стійкості України, одночасно зменшуючи залежність від зовнішніх поставок.

2.1.2 Динаміка споживання електроенергії за останні роки

Чинники, що сприяють такому швидкому зростанню, є як структурними, так і циклічними. З одного боку, центральну роль відіграють такі довгострокові тенденції, як збільшення електрифікації транспорту, опалення та промисловості. Зростання використання електро енергоємних послуг, таких як центри обробки даних та поширення таких приладів, як кондиціонери та обігрівачі. З іншого боку, 2024 рік відзначився надзвичайно високими глобальними температурами та поширеними хвилями спеки, що спричинило зростання попиту на охолодження, додавши значного поштовху пов'язаного з погодою.

Зростання є географічно сконцентрованим: переважна більшість приросту припадає на країни з економікою, що розвивається, та країни, що розвиваються в цілому. У 2024 році більше половини глобального зростання припадало на одну

країну - Китай. Іншими великими учасниками є Індія та стрімко зростаючі економіки Південно-Східної Азії. Тим часом багато розвинених економік, наприклад, у Північній Америці, Європі та деяких частинах Азії зазнали уповільнення зростання або навіть тимчасової стагнації, хоча у 2024 році в деяких з них спостерігалось відновлення попиту.

З боку пропозиції, виробництво чистої та низьковуглецевої електроенергії значно зросло. Відновлювані джерела енергії, особливо сонячна фотоелектрична енергія, забезпечили значну частку зростання світового виробництва електроенергії у 2024 році - лише сонячна фотоелектрична енергія додала приблизно 480 ТВт·год. Вітроенергетика та гідроенергетика також зросли, хоча темпи зростання вітроенергетики були більш помірними порівняно з швидким зростанням сонячної енергетики. Загалом чисті джерела покрили значну частину додаткового попиту, що свідчить про структурні зміни у способі покриття зростаючих потреб світу в електроенергії.

Таким чином, останні роки демонструють збіг обставин: сильне структурне зростання попиту на електроенергію, спричинене електрифікацією та цифровізацією, підсилене кліматичним попитом на охолодження у 2024 році, поряд із значним прискоренням розвитку потужностей з виробництва чистої енергії, особливо сонячної, що допомагає задовольнити це зростання.

Загалом, світове споживання електроенергії стрімко зростає під впливом структурної електрифікації та пов'язаних з кліматичними сплесками попиту, тоді як виробництво чистої енергії швидко розширюється. Тим часом Україна спостерігає різке скорочення споживання електроенергії через перебої, пов'язані з війною, але також досліджує можливості післявоєнної енергетичної трансформації на основі відновлюваних джерел енергії та енергоефективності.

2.1.3 Вплив різних чинників на споживання електроенергії

Серед чинників що впливають на споживання електроенергії прийнято виділяти деякі як більш впливові порівняно з іншими. До них можна віднести такі як: економічне зростання та підвищення рівня життя. У житлових умовах вищі доходи та покращення соціально-економічного статусу призводять до

ширшого володіння електроприладами та більших житлових площ. Дослідження, проведене в Тайвані, показало, що такі змінні, як ВВП, рівень зайнятості та площа житла, сильно корелюють із більшим споживанням електроенергії домогосподарствами. Аналогічно, опитування в міських районах Буркіна-Фасо показало, що доходи домогосподарств, наявність та використання побутових приладів (холодильників, кондиціонерів, телевізорів тощо), тип та розмір житла значною мірою впливають на споживання електроенергії. Урбанізація посилює цей ефект: міграція до міст зазвичай призводить до зростання доходів та переходу до енергоємного способу життя.

Поширення побутової техніки та електрифікація будівель. У міру того, як домогосподарства купують все більше побутової техніки (кондиціонери, холодильники, кухонне обладнання, освітлювальні прилади, електронні пристрої), загальне споживання електроенергії зростає. У випадку Тайваню лише кондиціонери становили ~26,8% споживання (без урахування освітлення), холодильники - ~14,9%, а рисоварки - ~6,3%. У глобальному масштабі зростання електрифікації систем опалення, охолодження, нагрівання води, приготування їжі та інших побутових потреб збільшило базовий попит на електроенергію - навіть без урахування нових технологій, таких як електромобілі.

Зміна клімату та екстремальні погодні умови, особливо спекотні хвилі. У 2024 році безпрецедентна глобальна спека спричинила широке використання кондиціонерів, що значно підвищило попит на електроенергію. МЕА повідомляє, що зростання попиту на охолодження було основним фактором рекордного збільшення споживання електроенергії в будівлях. У регіонах з теплішим кліматом і частішими спекотними хвилями потреба в охолодженні буде все більше і більше домінувати в попиті на електроенергію в житловому та комерційному секторах. Недавні дослідження також показують, що прогнозоване збільшення впливу спеки в поєднанні з соціально-економічним зростанням посилить попит на енергію для охолодження - з нерівномірним впливом на різні групи за доходами та регіони.

Технологічні та структурні зміни - електрифікація транспорту, промислове розширення та цифровізація. Швидке поширення електромобілів у всьому світі

додає нове значне навантаження на електромережу. За даними МЕА, у 2024 році попит на електроенергію для електромобілів зріс на 38% (+62 ТВт·год), що склало близько 0,2% від загального зростання попиту на електроенергію у світі. Попит на електроенергію в промисловості також зріс приблизно на 4% у 2024 році, що було зумовлено збільшенням активності в електроємному виробництві та більш широким промисловим зростанням.

Тим часом цифровізація - особливо зростання центрів обробки даних, хмарних послуг та обчислювальних потужностей штучного інтелекту - стала важливим новим фактором: центри обробки даних та пов'язана з ними інфраструктура значно збільшили споживання електроенергії у 2024 році, і тенденція до їх розширення прискорюється.

Зростання населення та урбанізація. У міру того, як все більше людей переїжджають до міст і населення в усьому світі зростає, загальний попит на електроенергію збільшується. Міські домогосподарства, як правило, споживають більше електроенергії, ніж сільські, через більш щільну електрифіковану інфраструктуру, більшу кількість побутової техніки та більш інтенсивне використання енергоємних послуг.

Структурні зміни в кінцевому споживанні та електрифікація енергетичних послуг. Історично деякі енергетичні послуги надавалися за допомогою неелектричних видів палива, таких як газ, біомаса тощо. По мірі розвитку економіки та просування електрифікації - для опалення, приготування їжі, гарячого водопостачання, транспорту - ці послуги переходять на електроенергію. Ці структурні зміни збільшують загальний попит на електроенергію, навіть якщо загальне споживання енергії залишається приблизно стабільним. Звіти вказують на цю тенденцію в усьому світі як частину більш широких змін в енергетичних системах.

Регіональна динаміка та нерівномірні моделі зростання. Значна частина нещодавнього зростання попиту на електроенергію припадає на країни з економікою, що розвивається, та країни, що розвиваються в цілому. За даними МЕА, очікується, що до 2027 року на ці країни припадатиме 85 % світового зростання попиту на електроенергію, зокрема завдяки таким країнам, як Китай,

Індія та країни Південно-Східної Азії, що зумовлено економічним зростанням, підвищенням доходів, урбанізацією та все більшим використанням електроенергоємних приладів. Однак у 2024 році в розвинених країнах також спостерігалось відновлення зростання попиту на електроенергію, що змінило попередню стагнацію, завдяки ліпшій погоді та збільшенню попиту з боку будівель і центрів обробки даних.

Поведінкова, демографічна та соціально-економічна неоднорідність. На рівні домогосподарств такі фактори, як розмір домогосподарства, склад (кількість мешканців, вік), поведінка (частота використання приладів), характеристики будівлі (площа, ефективність, ізоляція) та тип житла, сильно впливають на споживання електроенергії.

У деяких випадках, навіть якщо побутові прилади є у власності, моделі використання - як часто і яким чином вони використовуються значно впливають на попит.

Підсумовуючи, можна сказати, що глобальне споживання електроенергії зростає під впливом таких факторів, як підвищення доходів населення, зростання чисельності населення, урбанізація, електрифікація послуг і транспорту, розширення цифрової інфраструктури та збільшення попиту на охолодження через зміни клімату. Баланс і відносна вага цих факторів варіюються залежно від регіону, кліматичних зон та стадій економічного розвитку.

Оскільки попит зростає так швидко і одночасно під впливом багатьох чинників, забезпечення сталого енергопостачання шляхом розширення використання відновлюваних джерел енергії, підвищення енергоефективності, модернізації енергомереж, управління попитом та декарбонізації, а також збір, аналіз та прогнозування електроенергії стають критично важливими глобальними викликами.

2.2 Аналіз існуючих технічних рішень моніторингу енергоспоживання

Моніторинг енергоспоживання зазвичай здійснюється за допомогою систем, які часто називають системами енергоменеджменту (EMS). Ці системи складаються з апаратного забезпечення, на кшталт лічильників, датчиків, програмного забезпечення для збору даних, зберігання, інформаційні панелі для аналізу та процедур для прийняття рішень на основі даних. Згідно з опитуванням щодо проектування та впровадження EMS, основною основою таких систем є встановлення вимірювального та моніторингового обладнання, яке забезпечує точність і технічну якість зібраних даних.

2.2.1 Огляд популярних комерційних та open-source систем

При простому (інтрузивному) моніторингу на об'єкті встановлюються звичайні лічильники або датчики (для електроенергії, води, газу тощо). Такі датчики безперервно або періодично збирають дані про споживання. Потім дані агрегуються та обробляються програмними засобами енергоменеджменту, що дозволяє користувачам або менеджерам відповісти на ключові питання: коли використовується енергія, де і якими пристроями або підсистемами, та як змінюються енергетичні моделі з часом.

У більш просунутих підходах пропонується неінтрузивний моніторинг навантаження (NILM). У NILM система намагається реконструювати профіль споживання окремих приладів або навантажень, аналізуючи агреговані дані про споживання, таких як напруга, струм, потужність, гармоніки та часові характеристики, а не вимірюючи кожний пристрій окремо. Це зменшує необхідність у встановленні великої кількості датчиків і в принципі може забезпечити детальне розбиття енергії.

Сучасні реалізації EMS все частіше включають архітектури Інтернету речей: датчики та інтелектуальні лічильники спілкуються через дротові або бездротові мережі. Дані збираються централізовано - на місці або в хмарі. Програмне забезпечення виконує моніторинг у реальному часі, аналіз історичних даних, надає панелі інструментів користувача та аналітику.

Деякі фреймворки навіть поєднують моніторинг енергоспоживання з моніторингом якості внутрішнього середовища (IEQ) (температура, вологість, якість повітря), надаючи інтегровані дані про продуктивність будівлі.

Серед комерційних програмних платформ є такі рішення, як ClearVUE.Zero (Рис. 2.2), що забезпечує моніторинг електроенергії, газу та води в режимі реального часу, «віртуальний облік» на декількох об'єктах або в декількох мережах, інформаційні панелі, аналіз історичних даних, автоматизовану звітність щодо дотримання вимог до енергоспоживання та викидів, а також сповіщення, коли споживання або викиди перевищують заздалегідь визначені порогові значення.



Рис 2.2 Реалізація інтерфейсу системи моніторингу ClearVUE.Zero

Інші інструменти поєднують моніторинг енергії з управлінням об'єктами та оптимізацією портфеля будівель, наприклад ECOCITIES, який інтегрує облік енергії, управління будівлями та оптимізацію енергії для підтримки планування

заходів щодо зниження викидів вуглецю в декількох будівлях.

У промислових середовищах існують спеціалізовані протоколи для енергоефективної автоматизації, наприклад PROFIenergy (профіль протоколу зв'язку PROFINET), який дозволяє керувати енергоспоживанням обладнання автоматизації, особливо під час планових і позапланових перерв у виробництві, без необхідності використання зовнішніх дротових систем.

Академічні дослідження показують зростаючий інтерес до інтелектуальних систем управління енергоспоживанням на основі Інтернету речей для інтелектуальних будівель та міст. Ці системи використовують розподілені датчики, бездротовий зв'язок, вбудовані мікроконтролери, хмарну обробку даних та методи машинного навчання для розбиття навантаження та аналізу моделей споживання. Такі системи обіцяють більш гнучкий, детальний та зручний моніторинг та контроль енергоспоживання.

Таким чином, спектр технічних рішень для моніторингу енергоспоживання простягається від простих лічильників і ручного аналізу рахунків, через класичні системи енергоменеджменту (EMS) з датчиками та інформаційними панелями, до інтелектуального моніторингу на основі Інтернету речей і навіть неінтрузивного розподілу навантаження.

2.2.2 Виявлення переваг, недоліків та обмежень існуючих систем

Переваги моніторингу енергії - EMS та пов'язаних систем є численними і часто значними у фінансовому, операційному та екологічному вимірах. Надаючи інформацію про споживання в режимі реального часу або майже в режимі реального часу, EMS може виявити неефективність, марнотратство або можливості для перенесення споживання на час поза піковим навантаженням, що призведе до прямої економії коштів за електроенергію.

Ефективність операційної діяльності також покращується: безперервний моніторинг дозволяє своєчасно виявляти аномальні моделі споживання, які можуть свідчити про несправність або не оптимальне використання обладнання. Це дозволяє проводити профілактичне обслуговування або переналаштування,

замість повної заміни обладнання. Згодом це може продовжити термін експлуатації енергоємного обладнання та зменшити кількість не запланованих простоїв.

Системи моніторингу енергії сприяють екологічній стійкості. Зменшуючи непотрібне споживання та оптимізуючи використання енергії. Вони опосередковано зменшують викиди вуглецю, що може сприяти досягненню корпоративних або муніципальних цілей у сфері стійкості, дотриманню нормативних вимог та публічних зобов'язань щодо енергоефективності.

Централізований цифровий моніторинг та звітність спрощують адміністрування, виставлення рахунків та дотримання нормативних вимог. Замість того, щоб покладатися на ручне зчитування показань лічильників або періодичні рахунки, EMS може генерувати детальні звіти про споживання, підтримувати автоматизацію виставлення рахунків за комунальні послуги та полегшувати аудити або звітність щодо сталого розвитку.

Однак існують важливі недоліки та обмеження. Впровадження повної EMS часто вимагає значних початкових інвестицій: обладнання для вимірювання, датчики, установка, інтеграція та ліцензування програмного забезпечення. Для малих підприємств або домогосподарств ці витрати можуть бути надто високими.

Складність впровадження є ще однією перешкодою. Інтеграція датчиків, лічильників, збору даних та програмного забезпечення, а також забезпечення сумісності між пристроями та протоколами зв'язку може вимагати спеціалізованих знань. Для надійного функціонування системи можуть бути необхідні технічне обслуговування та регулярні оновлення.

Залежність від технологій та підключення до мережі збільшує ризик. Хмарні або мережеві EMS можуть бути вразливими до перебоїв в інтернеті, втрати даних або загроз кібербезпеці. Безпека та конфіденційність даних стають критично важливими, особливо коли детальні дані про споживання реєструються та передаються дистанційно.

Неінтрузивний моніторинг навантаження (NILM), хоча і привабливий своїми мінімальними вимогами до обладнання, залишається в основному

експериментальним і має обмежену практичну застосовність. Згідно з опитуванням 2016 року, NILM ще не є повністю життєздатним рішенням для точного вимірювання та контролю енергії, оскільки вимагає, щоб кожен пристрій мав впізнавану «сигнатуру», а розробка алгоритмів, які надійно ідентифікують усі навантаження в реальній установці, залишається складним і невіршеним завданням. Багато типів навантажень можуть бути нерозподілені точно або взагалі.

Інфраструктури на основі Інтернету речей та орієнтовані на інтелектуальні мережі покращують гнучкість та масштабованість, але вони також успадковують типові проблеми IoT: надійність датчиків, стабільність мережі, перевантаження даними, необхідність інфраструктури для обробки даних та потенційні вразливості кібербезпеки. Для великих систем обробка обсягів даних та перетворення необроблених вимірювань у практичні висновки залишається нетривіальною проблемою.

Крім того, закриті або пропрієтарні рішення EMS можуть створити залежність від постачальника - це може обмежити гнучкість, зробити технічне обслуговування або оновлення дорогими, перешкоджати взаємодії та обмежувати адаптацію до мінливих нормативних або технічних вимог.

2.2.3 Огляд необхідності існування локальних рішень

З огляду на обмеження та компроміси існуючих систем, існують вагомі підстави для розробки локальних, контекстно-орієнтованих рішень для моніторингу енергії, адаптованих до конкретних регіонів, будівель або груп користувачів. Локальні рішення можуть бути адаптовані до нормативно-правової бази, тарифів на енергію, інфраструктурних обмежень та моделей використання, що є актуальними для конкретного регіону. Це зменшує залежність від іноземних постачальників або глобальних хмарних платформ, які можуть не відображати місцеві ринки електроенергії або екологічні пріоритети.

Місцеві рішення, особливо ті, що базуються на відкритих стандартах, програмному забезпеченні з відкритим кодом або модульному обладнанні,

можуть бути більш економічно ефективними, гнучкими та прозорими. Вони дозволяють здійснювати індивідуальне налаштування: наприклад, у житловому будинку рішення може інтегрувати місцеві типи лічильників, враховувати типові моделі навантаження (опалення, побутова техніка, змінне енергопостачання) та підтримувати місцеві вимоги до звітності та аудиту.

Крім того, місцеві системи можуть уникнути деяких вразливостей, пов'язаних з хмарними або контрольованими постачальниками послугами: легше гарантувати конфіденційність даних, оскільки інформація залишається під місцевим контролем. Також можна зменшити залежність від мережі. Технічне обслуговування та оновлення можна здійснювати на місцевому рівні, а довгострокова стійкість є більш реальною без зовнішніх витрат на передплату або ліцензування іноземного програмного забезпечення.

У регіонах, що стикаються з нестабільністю інфраструктури, мінливістю енергопостачання або змінами в законодавстві, наявність локально керованої модульної системи управління енергоспоживанням (EMS) може підвищити стійкість і зробити довгострокове управління енергоспоживанням більш доступним. Тому розробка місцевих технічних рішень, або адаптація існуючих архітектур EMS до місцевих умов - повинна розглядатися як стратегічний пріоритет для забезпечення стійкої енергоефективності.

Існуючі технічні рішення для моніторингу енергоспоживання варіюються від класичних інтрузивних лічильників до передових підходів на основі Інтернету речей і навіть неінтрузивних методів моніторингу навантаження. Вони забезпечують очевидні переваги в плані економії витрат, операційної ефективності, впливу на навколишнє середовище та аналітики. Однак вони також мають обмеження: висока вартість, складність, технологічна залежність, проблеми конфіденційності та безпеки, а в деяких підходах (наприклад, NILM) - технічна незрілість. Для подолання цих обмежень і максимізації переваг моніторингу енергії необхідні локальні, контекстно-орієнтовані рішення, засновані на відкритих або модульних фреймворках і адаптовані до регіональної інфраструктури та моделей використання.

2.3 Аналітична оцінка моделей для прогнозування споживання електроенергії

Клас моделей, заснованих на авторегресійній інтегрованій ковзній середній (ARIMA), залишається основним інструментом для прогнозування споживання електроенергії, оскільки він є добре зрозумілим, інтерпретованим і працює надійно, коли базовий часовий ряд є достатньо стаціонарним після диференціювання (видалення тренду/сезонності). Формалізм моделі ARIMA (та її сезонний варіант Seasonal ARIMA - SARIMA) передбачає, що після відповідного диференціювання залишковий ряд можна моделювати як комбінацію минулих значень та минулих похибок за допомогою середнього значення.

Однак чиста ARIMA/SARIMA може бути недостатньою, коли часовий ряд споживання електроенергії демонструє складні сезонні закономірності, нелінійні залежності або зовнішні впливи, такі як погода, завантаженість, економічні або поведінкові фактори. У таких випадках гібридні підходи, що поєднують ARIMA з моделями машинного навчання або нейронними мережами, як правило, забезпечують кращі результати прогнозування. Наприклад, недавній огляд показав, що ARIMA досягла конкурентоспроможних середніх похибок, але гібридні та ансамблеві моделі (включаючи моделі на основі глибокого навчання) продемонстрували вищу надійність і точність, особливо за наявності екзогенних змінних.

Дослідження, що зосереджуються на мікро мережах та споживанні в менших масштабах, наприклад, домогосподарства, будівлі показують, що поєднання методів декомпозиції (тренд + сезонність + залишок) зі статистичними або ML-моделями може забезпечити вищу стабільність і точність. Декомпозиція ізолює детерміновані компоненти та дозволяє ефективніше моделювати стохастичну поведінку залишку.

Підсумовуючи, моделі на основі ARIMA забезпечують надійну базу для прогнозування споживання електроенергії, особливо коли даних багато, історія стабільна, а зовнішні впливи обмежені. Для більш складних, змінних або зашумлених даних про споживання в реальному світі (навантаження

домогосподарств, мікромережі, міський попит) гібридні моделі, що поєднують статистичну строгість із гнучкістю машинного навчання, як правило, дають більш точні та надійні прогнози.

2.3.1 Побудова прогнозів на короткостроковий період

Короткострокове прогнозування, такі як годинного до щоденного та щотижневого горизонту споживання електроенергії має на меті відобразити безпосередні коливання та сезонність, притаманні моделям попиту. Такі прогнози мають вирішальне значення для оперативного планування: збалансування попиту та пропозиції, коригування виробництва або зберігання, запобігання перевантаженням та оптимізація рішень щодо закупівель або відправки. Для короткострокового прогнозування моделі повинні враховувати добові цикли, тижневі закономірності (будні дні проти вихідних), залежні від погоди коливання - температура, вологість та, можливо, непередбачувані події такі як зайнятість, свята, поведінка людей.

Традиційні моделі ARIMA або SARIMA можуть застосовуватися для короткострокового прогнозування, якщо часові ряди споживання після диференціювання та за необхідності сезонного коригування демонструють стаціонарну поведінку або стабільну структуру автокореляції. Ці моделі можуть ефективно фіксувати регулярні закономірності, наприклад, щоденну або щотижневу сезонність за умови, що сезонність є послідовною та періодичною. Однак, коли на попит сильно впливають зовнішні, нестаціонарні фактори, такі як зміни погоди, раптові зміни навантаження, суто авторегресійна/середньозважена структура ARIMA може не встигати швидко адаптуватися.

У багатьох останніх дослідженнях для короткострокового прогнозування використовуються гібридні моделі, що поєднують ARIMA для фіксації лінійних, автокорельованих компонентів з моделями машинного навчання або глибокого навчання (для моделювання нелінійних залежностей або залишкової структури). Наприклад, гібридне поєднання ARIMA та мереж Long Short-Term Memory

(LSTM) показало меншу похибку прогнозування порівняно з ARIMA окремо. У таких гібридних моделях лінійний тренд та сезонність обробляються ARIMA, а нелінійні залишкові залежності залишаються LSTM або іншим підходам ML. Ця подвійна методологія, як правило, дає надійні та достовірні короткострокові прогнози навіть у шумних або швидкозмінних середовищах.

Порівняння точності моделей прогнозування за різними стратегіями моделювання зазвичай демонструє, що хоча ARIMA є конкурентоспроможною і часто перевершує прості регресійні або наївні методи, гібридні або ML-орієнтовані методи (особливо ті, що використовують LSTM, GRU або подібні рекурентні архітектури) мають тенденцію до досягнення нижчих показників похибки, таких як RMSE (середньоквадратична похибка), MAE (середня абсолютна похибка) або MAPE (середня абсолютна процентна похибка). У деяких випадках гібридні моделі забезпечують значно кращу продуктивність, особливо коли часові ряди демонструють нелінійну поведінку або коли використовуються зовнішні контекстні змінні.

Тому для короткострокового прогнозування споживання електроенергії гібридні підходи, що використовують як статистичне моделювання часових рядів, так і машинне навчання, як правило, є найсучаснішими, забезпечуючи баланс між інтерпретованістю, адаптивністю та точністю прогнозування.

2.3.2 Порівняння точності моделей

Порівняння точності різних моделей прогнозування сильно залежить від характеристик набору даних, горизонту прогнозування та показників оцінки. В огляді методів прогнозування споживання електроенергії модель ARIMA досягла середньої похибки в 3,8%, перевершивши регресійні моделі (близько 8,1%) і навіть штучні нейронні мережі в цьому конкретному контексті. В іншому дослідженні, в якому порівнювалися моделі ARIMA, Support Vector Machine (SVM) та LSTM для прогнозування споживання енергії, SVM показала нижчий RMSE, ніж ARIMA та LSTM у сценарії прогнозу на один рік, хоча автори зазначають, що навіть SVM мала труднощі з підтриманням достатньої точності

протягом такого тривалого горизонту.

При використанні гібридної моделі, що поєднує ARIMA з LSTM (або випадковим лісом), точність, як правило, покращується порівняно з автономними методами. У недавньому експериментальному дослідженні гібридна модель ARIMA-LSTM досягла нижчого середньоквадратичного відхилення (MSE), RMSE та MAE порівняно з ARIMA окремо. Інша лінія досліджень відстоює методи декомпозиції (тренд + сезонність + залишок) з подальшим моделюванням залишку, що покращує стабільність і точність прогнозування процесів споживання.

У контекстах із сильною сезонністю та стабільними моделями споживання традиційні статистичні моделі, такі як ARIMA/SARIMA або моделі на основі експоненціального згладжування, залишаються конкурентоспроможними. У більш мінливих контекстах (мікромережі, окремі домогосподарства, попит, що залежить від клімату, коливання поведінки) машинне навчання або гібридні моделі демонструють очевидні переваги в прогностичній ефективності та надійності.

Таким чином, вибір моделі повинен відображати структуру та мінливість даних про споживання, горизонт прогнозування та прийнятні межі похибки. Для багатьох реальних завдань прогнозування споживання електроенергії гібридні або доповнені ML моделі на даний момент забезпечують найкращий компроміс між точністю, гнучкістю та інтерпретованістю.

2.3.3 Обґрунтування доцільності використання прогнозів на edge-пристроях

Впровадження моделей прогнозування споживання електроенергії на периферійних пристроях, таких як вбудовані системи, інтелектуальні лічильники, локальні контролери має кілька переваг. По-перше, периферійні прогнози дозволяють прогнозувати попит у режимі реального часу або майже в режимі реального часу без необхідності постійного зв'язку з центральними серверами. Це зменшує затримку, дозволяє швидше реагувати на коливання попиту та підтримує рішення щодо локального контролю, наприклад,

регулювання навантаження, активацію сховищ або перерозподіл споживання в мікромережах.

По-друге, коли моделі працюють локально, покращується конфіденційність і безпека даних, оскільки необроблені дані про споживання не потрібно передавати назовні. Для таких застосувань, як прогнозування на рівні домогосподарств або управління мікромережами, це особливо важливо, оскільки дані про використання можуть бути конфіденційними.

По-третє, розгортання на периферії забезпечує розподілену масштабованість: замість того, щоб обтяжувати центральні сервери величезними обсягами високочастотних даних для багатьох користувачів, кожен пристрій виконує локальне прогнозування і агрегує лише необхідну зведену інформацію. Це зменшує вимоги до пропускну здатності та обчислювальних потужностей центральної інфраструктури.

Нарешті, поєднання легких статистичних моделей (наприклад, ARIMA, прості декомпозиції) або компактних ML-моделей (наприклад, не глибокі нейронні мережі, невеликі мережі LSTM/GRU, деревоподібні моделі) дозволяє периферійним пристроям з обмеженими обчислювальними та пам'яттю ресурсами забезпечувати значущі результати прогнозування. Останні дослідження показують, що такі гібридні або ML-орієнтовані системи прогнозування можуть бути адаптовані до мобільних або вбудованих платформ, зберігаючи точність і оптимізуючи використання ресурсів.

Прогнозування на основі периферійних пристроїв підтримує управління попитом, балансування навантаження, енергозбереження та стабільність розподіленої мережі. Всі ці функції є критично важливими для сучасних інтелектуальних мереж та децентралізованих енергетичних систем.

3. РОЗРОБКА ІОТ-СИСТЕМИ ЗБОРУ, АНАЛІЗУ ТА ПРОГНОЗУВАННЯ СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ

Перед початком розробки системи інтернет речей для збору, аналізу та прогнозування споживання електроенергії потрібно провести детальний аналіз, для того щоб виявити недоліки існуючих рішень та шляхи їх подолання. Також потрібно сформулювати вимоги до майбутньої системи.

3.1 Визначення ключових проблем і формування вимог до майбутньої IoT-системи

3.1.1 Виявлені недоліки існуючих підходів

Основними недоліками готових рішень що були розглянуті в минулих розділах можна виокремити такі:

1. Ціна - може бути занадто високою, адже впровадження повноцінної системи EMS часто вимагає значних початкових інвестицій: обладнання для вимірювання, датчики, установка, інтеграція та ліцензування програмного забезпечення.
2. Інтеграція - може бути занадто складною. Впровадження датчиків, лічильників, збору даних та програмного забезпечення, а також забезпечення сумісності між пристроями та протоколами зв'язку може вимагати спеціалізованих знань.
3. Гнучкість - закриті або пропрієтарні рішення EMS можуть створювати залежність від постачальників. Це може обмежити гнучкість, зробити технічне обслуговування або оновлення дорогими, перешкоджати взаємодії та обмежити адаптацію до мінливих нормативних або технічних вимог.
4. Передача даних - залежність від технологій та підключення до мережі збільшує ризик. Хмарні або мережеві EMS можуть бути вразливими до перебоїв в інтернеті, втрати даних або загроз кібербезпеки.

3.1.2 Потреба у локальному опрацюванні даних

Саме через ці недоліки, особливо виокремлюючи мережеві вразливості - виникає необхідність в зборі та опрацюванні даних безпосередньо на локальному пристрої. Це дозволить зменшити затримки, оскільки локальне обчислення дозволить швидше аналізувати та обробляти дані в місці де вони безпосередньо виникають. За допомогою цього можна скоротити затримки а також збільшити ефективність пропускної здатності, адже даних буде відправлятися менше. Ключовим плюсом системи з локальною обробкою буде підвищення безпеки. Завдяки аналізу та обробці що відбуваються недалеко від місця збору даних можна зменшити обсяг даних що необхідно передавати в хмару. В результаті цього третім особам буде складніше зламати систему та заволодіти даними, оскільки стає менше вразливих місць та можливостей для потенційної хакерської атаки.

3.1.3 Формування технічних і функціональних вимог до системи

Розглянувши переваги та недоліки готових рішень, а також особливості опрацюванні даних можна виділити декілька основних вимог до майбутньої системи:

- Ціна - повинна бути невисокою, задля можливості інтеграції системи в невеликі структури, та конкуренції з готовими рішеннями.
- Гнучкість - система не повинна обмежувати користувача у можливості модифікації системи для покращення або адаптації роботи для специфічних сценаріїв.
- Edge Computing - локальні обчислення в системі повинні бути в пріоритеті, оскільки вони дають багато переваг, таких як менші затримки, нижча вірогідність втрати даних та хакерських атак.
- Працездатність - система повинна вміти виконувати свій основний функціонал, а саме вміти: збирати, аналізувати та прогнозувати дані що в неї поступають.

3.2 Архітектура IoT-системи

Перед етапом розробки необхідно створити та розібрати структуру майбутньої системи, вибрати програмні та апаратні компоненти, а також методи збору, обробки та модель прогнозування що будуть використовуватись. Завдяки цим етапам можна буде зрозуміти її архітектуру, та взаємодію всіх програмних та апаратних компонентів.

3.2.1 Вибір апаратних компонентів

Найголовнішим компонентом системи можна вважати центральну плату, на якій будуть виконуватись всі обчислення. У даному випадку в якості основи для системи збору, аналізу та прогнозування споживання електроенергії використовувалась плата від компанії Orange pi, а саме orange pi модель zero 3 (Рис. 3.1).

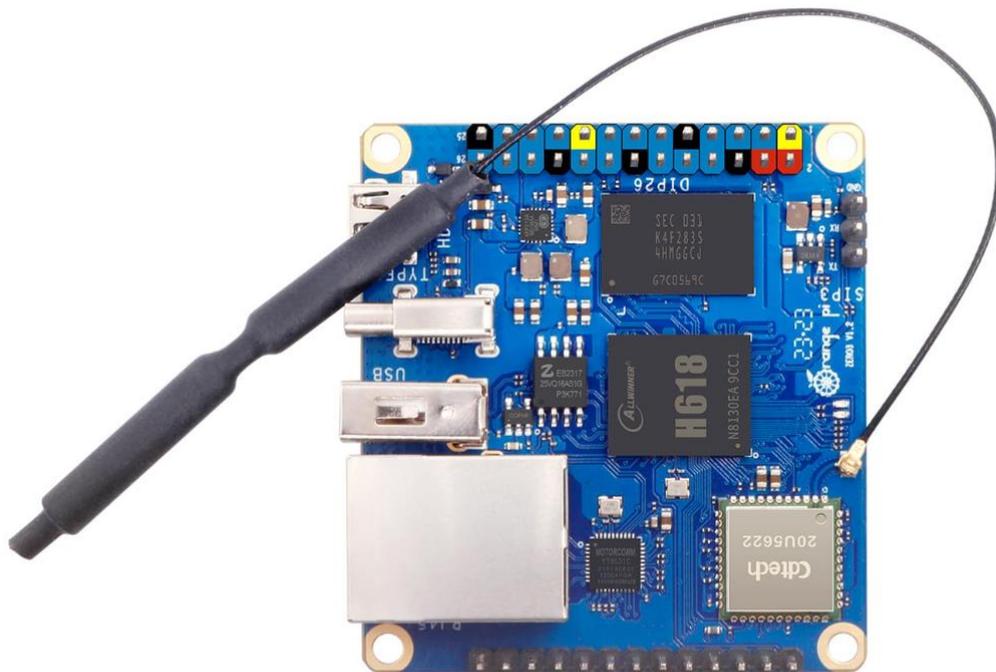


Рис 3.1 одноплатний комп'ютер Orange Pi Zero 3

Плата має чудові можливості для свого невеликого розміру, а саме 5.3 сантиметрів на 6 сантиметрів що приблизно дорівнює половині банківської

картки. На борту вона має чотириядерний процесор Allwinner H618 з ядрами Cortex-A53, що забезпечить достатню потужність для необхідних для системи обчислень. Графічним процесором є Arm Mali-G31 MP2, підтримує OpenGL ES 1.0/2.0/3.2, OpenCL 2.0, Vulkan 1.1 тощо, який може знадобитись для відмальовування інтерфейсу. В даній конфігурації плати має 1,5 ГБ оперативна пам'яті (є ще варіанти на 1, 2 та 4 гігабайти пам'яті), а також має безліч інтерфейсів, таких як usb Type-c для живлення, мікро Hdmi для вивода зображення на дисплей, слот microSD для встановлення системи на карту пам'яті та збору даних на неї, а також мережеві технології зв'язку - Wi Fi та Bluetooth для того щоб передавати дані в бездротовому режимі.

Крім того, на платі Orange Pi Zero 3 також розміщується укорочений роз'єм gpiu, а саме варіант на 26 контактів, що використовується для швидкого підключення різноманітних датчиків та доповнень.

Для вимірювання електричного струму системі необхідний відповідний датчик. В якості такого виступить модуль вимірювання параметрів споживання змінного струму PZEM-004T (Рис. 3.2), призначений для вимірювання вхідної напруги, споживаного навантаженням струму. Модуль здатний вимірювати напругу більше ніж 20 кВт, а також має хорошу точність виміру в діапазоні від 0 до 10 кВт.



Рис. 3.2 Вимірювач споживаної електроенергії PZEM-004T-100A

Для роботоспроможності системи необхідне встановлення програмного забезпечення. Оскільки плата orange pi zero 3 не має на борту вбудованої пам'яті нам необхідно буде використовувати карту пам'яті microsd. Карту можна брати будь-яку. Головним критерієм у виборі буде те - щоб вона не була занадто стара та повільна, тому що це може вплинути на швидкодію системи, а отже і на процеси що відбуваються під час її роботи, такі як збір, зберігання та прогнозування даних. В даній кваліфікаційній роботі буде використовуватись карта пам'яті microsd від компанії SanDisk моделі Ultra об'ємом 32 gb (Рис. 3.3). Вона має непогані характеристики і невисоку ціну, та стане чудовим вибором.



Рис. 3.3 карта пам'яті microsd SanDisk Ultra 32 gb

Оскільки для функціонування системи потрібна електроенергія, виникає необхідність в живленні пристрою. Для цього можна використати будь який блок живлення (Рис. 3.4) що працює від мережі. Головне щоб блок відповідав необхідним для системи характеристиками, а саме: штекер повинен бути Type-c (саме такий порт встановлений на одноплатному комп'ютері), а також видавати напругу 5 V та струм 3 A, що в сумі дають 15 W. Ця необхідність визвана тим, що саме стільки комп'ютер буде споживати при максимальному навантаженні. Відповідно якщо взяти блок з меншою потужністю і система почне споживати більше ніж те значення на яке розрахований блок - система вимкнеться, а

процеси обробки зупиняться.

Альтернативою блоку живлення може стати power bank у зв'язці з кабелем type-c. Power bank повинен мати підтримку протоколів quick charge або power delivery (в залежності штекеру що знаходиться на іншій стороні кабелю).

Ще один спосіб заживити систему - це інтегрувати в неї аккумулятори 18650, підвищуючий перетворювач MT3608 та засунути це все в корпус.

Такі способи живлення для системи дають можливості її використання під час відключень світла, або в ситуаціях де з якихось причин немає доступу до мережі електроенергії.



Рис. 3.4 Стандартний блок живлення з штекером Type-c

3.2.2 Вибір програмних засобів

Для функціонування системи на одноплатному комп'ютері потрібно встановити низку програмних компонентів, для подальшого запуску елементів прогнозування. Першим компонентом стане операційна система DietPI - дистрибутив заснований на linux полегшений та оптимізований для роботи саме на одноплатних комп'ютерів (Рис. 3.5).

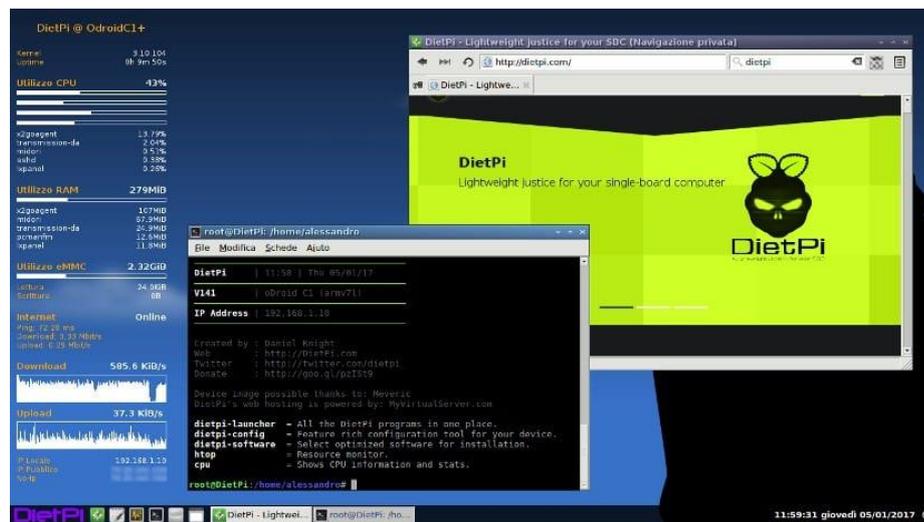


Рис. 3.5 Інтерфейс операційної системи DietPI

Для встановлення системи на плату знадобиться карта пам'яті microsd, картридер та ПК з програмою яка може записувати загрузочні флешки, в даному випадку буде використовуватись програма balena etcher (Рис. 3.6).

Для того щоб встановити операційну систему необхідно завантажити образ DietPI що сумісний з нашою платою, далі відформатувати карту пам'яті в формат fat32, після цього завантажити образ системи в програму balena etcher та обрати в інтерфейсі програми карту пам'яті для встановлення і натиснути кнопку Flash. Процедура займе кілька хвилин, по закінченю яких програма сповістить користувача про успіх чи провал операції. Якщо операція пройшла успішно можна діставати карту пам'яті з ПК та вставляти її в плату.

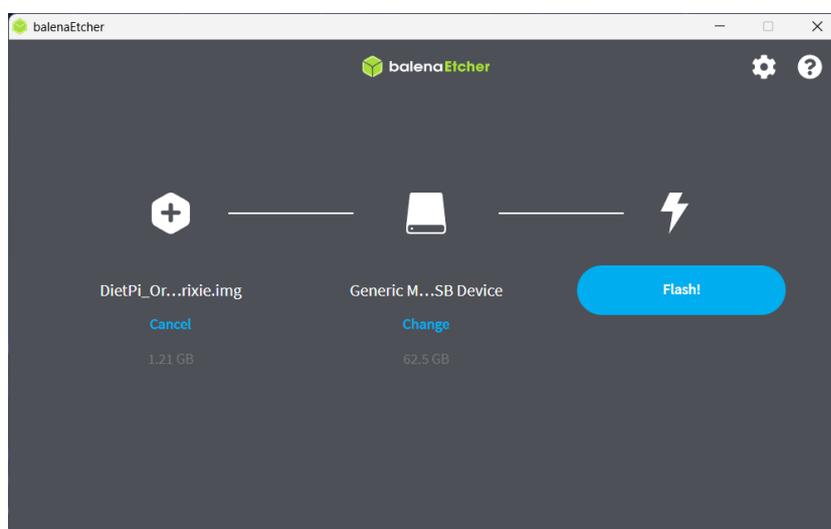


Рис. 3.6 Інтерфейс програми balena etcher

3.2.3 Методика збору, групування та обробки даних

Для того щоб система могла збирати, групувати та обробляти дані їй окрім операційної системи також необхідні інструменти. В якості такого виступить Python. Легка і ефективна мова програмування для реалізації системи. Для її функціонування також буде необхідне віртуальне середовище, таке як `venv`. `Venv` (`virtual environment`) це стандартний інструмент Python для створення легких, ізольованих середовищ для продуктів, що містять власний інтерпретатор Python та бібліотеки, запобігаючи конфліктам версій між різними проектами та глобальною інсталяцією. Всі залежності будуть фіксуватись у файлі `requirements.txt` - це дає відтворюваність середовища.

Також для функціонування системи необхідно додати до python декілька бібліотек, які розраховані на ті чи інші функції.

А саме:

1. `sqlite3` (вбудована бібліотека) - локальна база даних для збереження показників. `Sqlite` це непогане рішення, адже вона легковага тобто не потребує окремого сервера, достатня для невеликих і середніх обсягів телеметрії й історичних записів, а також має просту процедуру бекапів і сумісність з Python. Крім того використання режиму `WAL` (`Write-Ahead Logging`) підвищує паралельність: дозволяє читачам працювати під час записів. Це важливо для веб-інтерфейсу та фонових записів.
2. `numpy` - займається ефективними числовими операціями, швидкими арифметичними обчисленнями.
3. `plotly` - побудова графіків. Дозволяє будувати інтерактивні графіки в браузері.
4. `logging` - стандартна бібліотека для логування роботи сервісів.
5. `pytest` / `unittest` - для модульного тестування.

Крім бібліотек також необхідно встановити `Flask` - він допоможе в побудові локального веб-інтерфейсу, та сервіс `systemd` для автозапуску всієї системи. Це допоможе наблизити систему до готового рішення яким можна користуватись “з коробки” замість того щоб вводити кожен раз команди.

3.2.4 Метод прогнозування

Для прогнозування в системі буде використовуватись модель SMA (Simple Moving Average) - це інструмент, що використовується для обчислення та аналізу точок даних шляхом створення серії середніх значень різних вибірок з повного набору даних. В даному випадку буде використовуватись простий підвид моделі. Його буде цілком достатньо для системи і своїми обчисленнями вона не буде перевантажувати систему.

Формула моделі продемонстрована на рис 3.7.

$$SMA_k = \frac{1}{k} \sum_{i=n-k+1}^n p_i$$

The diagram illustrates the formula for Simple Moving Average (SMA) with the following annotations:

- Window size:** An arrow points from the text "window size" to the variable k in the denominator.
- Total # of observed values:** An arrow points from the text "total # of observed values" to the variable n in the upper limit of the summation.
- Single observed value:** An arrow points from the text "single observed value" to the variable p_i inside the summation.
- Summation Index:** A bracket under the lower limit $i=n-k+1$ is labeled "i.e. total values minus window size plus one".
- Alpha:** A small Greek letter α is located at the bottom right of the diagram.

Рис. 3.7 Формула Simple Moving Average

Перевагами цього методу прогнозування буде легкість, а також простота реалізації на пристроях з обмеженими ресурсами. З недоліків можна відмітити те, що такий метод прогнозування не враховує сезонність, а тобто погано реагує на раптові зміни. Це можна виправити за допомогою комбінації з детекцією сезонності або більш складними моделями для різних масштабів часу.

3.2.5 Архітектурна схема IoT-системи

Апаратна частина розробляємої системи ґрунтується на використанні одноплатного комп'ютера Orange Pi, який виконує роль центрального вузла edge-обробки даних. До нього підключається датчик електричних параметрів

PZEM-004T яких виконує функцію вимірювання напруги, струму та активної потужності у реальному часі. Комунікація між датчиком та одноплатним комп'ютером забезпечується через стандартний апаратний інтерфейс - UART, що дозволяє здійснювати передавання даних із мінімальною затримкою та високою надійністю.

Orange Pi виконує функції локального центру обробки інформації, приймаючи вимірювані дані та здійснюючи відповідну попередню фільтрацію, усереднення та підготовку до подальшого аналізу. Наявність повноцінної операційної системи на базі Linux дає змогу запускати програмне забезпечення, яке обробляє ці дані без потреби у зовнішніх серверних ресурсах. Додатково пристрій підтримує локальну мережеву взаємодію через Ethernet або Wi-Fi, що забезпечує доступ користувача через веб-інтерфейс до поточних та історичних показників електроспоживання. Таким чином, апаратна частина системи поєднує компактність та низьку вартість з достатньою обчислювальною потужністю, необхідною для виконання edge-функцій у режимі реального часу. Архітектуру апаратної частини можна зобразити за допомогою діаграми, яка зображена на рис 3.8.

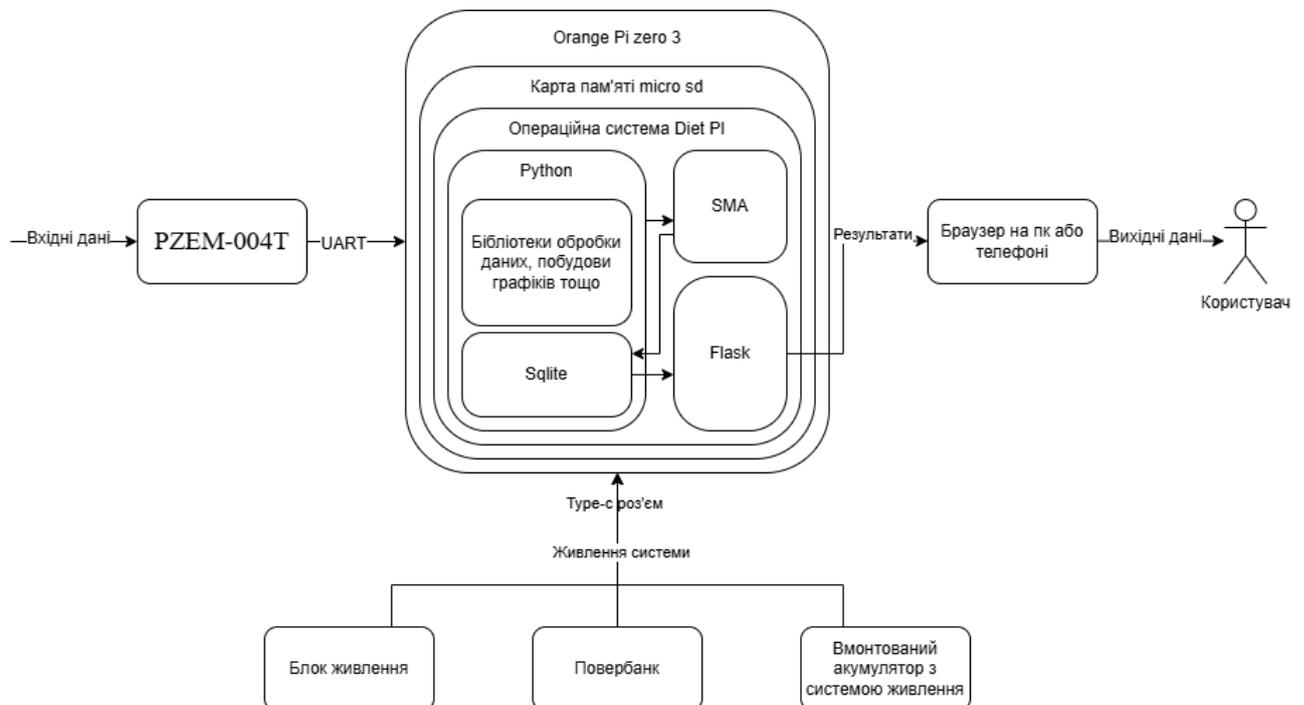


Рис. 3.8 Архітектурна схема апаратної частини системи прогнозування

Програмна частина системи реалізує повний цикл обробки даних: від отримання показників із сенсорів до їх збереження, аналізу та побудови прогнозів. Основою програмного забезпечення є скрипти на мові Python, які забезпечують періодичне зчитування вимірних параметрів, виконання валідації значень та їх подальшу агрегацію.

Після обробки дані зберігаються у локальній базі SQLite, що виступає компактним і надійним рішенням для збереження часових рядів без потреби встановлення повноцінної серверної СУБД.

На основі накопичених даних система виконує прогнозування споживання електроенергії. Для цього застосовуються метод аналізу часових рядів, а саме ковзне середнє. Алгоритм прогнозування запускається за розкладом і дозволяють отримати короткострокові прогнози на основі останніх агрегованих вимірювань, що підвищує оперативність прийняття рішень та забезпечує корисність системи у побутових або локальних умовах.

Взаємодія з користувачем реалізована через веб-інтерфейс, який розгортається на Orange Pi за допомогою веб-фреймворку Flask. Користувач має можливість переглядати актуальні дані, історичні графіки та прогнозні значення через будь-який веб-браузер на пристроях, таких як ПК або смартфон, що знаходяться у локальній мережі. Така організація дозволяє повністю відмовитися від зовнішніх хмарних сервісів, зберігаючи автономність і конфіденційність даних, а також забезпечуючи роботу системи навіть за відсутності доступу до Інтернету. Програмна архітектура вирізняється простотою, модульністю та гнучкістю, що спрощує розширення функціональності та адаптацію системи під різні умови експлуатації.

3.3 Розробка системи прогнозування

Функціонування розробленої IoT-системи базується на послідовному виконанні операцій зі збору, оброблення та візуалізації даних електроспоживання.

3.3.1 Алгоритми збору, запису та візуалізації даних

Збирання первинної інформації здійснюється за допомогою датчика PZEM-004T, який передає вимірювальні дані через інтерфейс UART на одноплатний комп'ютер Orange Pi Zero 3. Програмна реалізація алгоритму збору інтегрована у модуль *data_acq.py*, що використовує бібліотеку *minimalmodbus* для доступу до реєстрів датчика та отримання значень напруги, струму, активної потужності та спожитої енергії. Для забезпечення робастності роботи передбачено симуляційний режим, який дозволяє генерувати тестові дані під час налагодження або за відсутності фізичного сенсора.

Після отримання вимірювань виконується їх стандартизована перевірка, що передбачає відсіювання аномальних або некоректних значень та підготовку даних до збереження. Агреговані показники разом із часовими мітками фіксуються в локальній базі даних SQLite за допомогою модуля *db.py*, який інкапсулює механізми створення таблиць, вставки записів та забезпечує цілісність даних. Використання вбудованої файлової бази є оптимальним з огляду на обмежені апаратні ресурси системи та потребу у стабільній автономній роботі.

Візуалізація даних реалізована у веб-інтерфейсі, що функціонує на основі додатку Flask - *app.py*. Інтерфейс забезпечує доступ до актуальної інформації користувача через REST-ендпоінти, які повертають останні вимірювання, історичні дані та допоміжну аналітичну інформацію. Для графічного подання показників використовується бібліотека Plotly, що генерує інтерактивні веб-графіки у браузері користувача. Таким чином, система формує замкнений цикл оброблення даних - від одержання первинних значень до формування візуалізованих результатів, що повністю здійснюється на локальному edge-вузлі.

3.3.2 Принцип роботи локальної моделі прогнозування

Модель прогнозування, реалізована у модулі *forecast.py*, що побудована на основі методу простого ковзного середнього (Simple Moving Average). Обґрунтування вибору цього методу визначається його низькою

обчислювальною складністю, високою стабільністю та адекватністю для короткострокового прогнозування часових рядів електроспоживання. Зважаючи на специфіку апаратної платформи, застосування SMA дозволяє забезпечити виконання аналітичних операцій без суттєвого навантаження на процесорні ресурси Orange Pi Zero 3.

Процес прогнозування ґрунтується на вибірці останніх N агрегованих вимірювань потужності з бази даних, після чого обчислюється їх середнє значення. Отриманий результат слугує основою для оцінювання майбутніх значень електроспоживання на короткий часовий інтервал. Обчислення виконується функцією *compute_and_store()*, яка не лише формує прогноз, а також за необхідності зберігає його у відповідній таблиці бази даних для подальшої аналітичної обробки або відображення у веб-інтерфейсі.

Перевагою застосованої моделі є її прозорість і можливість інтерпретації результатів, що суттєво спрощує аналіз та оцінювання якості прогнозів. На відміну від складніших стохастичних або нейронних моделей, SMA не потребує стадій навчання, параметричного налаштування або тривалої обчислювальної підготовки, що робить її придатною для систем реального часу та edge-архітектур. Виконання прогнозування безпосередньо на локальному пристрої забезпечує автономність, відмовостійкість та мінімізує залежність системи від зовнішніх сервісів.

3.3.3 Локальний веб-інтерфейс

Локальний веб-інтерфейс, реалізований у модулі *app.py*, надає функцію доступу користувачеві до вимірянних та прогнозних даних. Додаток побудований на основі мікрофреймворку Flask, який забезпечує розгортання HTTP-сервера на одноплатному комп'ютері та формує інтерактивне середовище для взаємодії користувача із системою. Логіка роботи інтерфейсу базується на визначенні REST-маршрутів, що здійснюють запити до локальної бази даних та повертають структуровані результати у форматі JSON.

HTML-частина інтерфейсу формує єдину сторінку, на якій динамічно

відображаються графіки та ключові параметри електроспоживання. Побудова графіків здійснюється клієнтською частиною через бібліотеку Plotly, яка отримує дані з відповідних API-ендпоінтів та генерує інтерактивні діаграми у веб-браузері. Такий підхід забезпечує високу гнучкість інтерфейсу та дає змогу здійснювати аналітичний огляд даних безпосередньо на екрані користувача.

Важливою характеристикою розробленого веб-інтерфейсу є його незалежність від зовнішніх обчислювальних ресурсів. Усі обчислення, такі як прогнозування, агрегація та фільтрація даних виконуються на стороні серверної частини, що функціонує безпосередньо на Orange Pi. Це знижує навантаження на клієнтські пристрої та забезпечує стабільну роботу системи навіть у умовах обмеженого мережевого доступу. Таким чином, веб-інтерфейс є зручним інструментом для моніторингу та аналізу стану електроспоживання в реальному часі, інтегрованим у структуру локальної IoT-системи.

3.3.4 Автоматизація роботи системи

Автоматизація роботи системи відіграє ключову роль у забезпеченні її безперервного функціонування, стабільності та автономності в умовах edge-обчислень. Оскільки розроблена IoT-система призначена для тривалого збору й аналізу даних електроспоживання без постійного втручання користувача, важливою складовою її організації є формування механізмів автоматичного запуску, контролю виконання та періодичного оновлення вимірювальної та прогнозної підсистем. Для цього використано вбудовані можливості операційної системи DietPi на базі Linux, зокрема системний менеджер *systemd*, який забезпечує автоматичний запуск ключових програмних модулів під час завантаження пристрою, а також їхнє відновлення у випадку аварійного завершення роботи.

Автоматизація збору даних здійснюється шляхом створення окремого *systemd*-сервісу (Таблиця 3.1), що запускає модуль *data_acq.py* у фоновому режимі. Завдяки цьому забезпечується безперервне опитування датчика PZEM-004T та запис показників до бази даних з фіксованою періодичністю. Оскільки

сервіс працює незалежно від користувача, система продовжує виконувати свої функції навіть у випадках короткочасного перезавантаження або відсутності доступу до інтерфейсу керування.

Процедура автоматичного прогнозування реалізована за допомогою планувальника *cron*, що викликає модуль *forecast.py* через певні інтервали часу. Це забезпечує регулярне оновлення прогнозних значень без створення додаткового навантаження на основний процес збору даних. Інтеграція планувальника з серверною частиною дозволяє підтримувати актуальність інформації у веб-інтерфейсі та гарантує синхронізацію між фактичними вимірюваннями та результатами прогнозування.

Таблиця 3.1 Код конфігураційного файлу *systemd*-сервісу

```
[Unit]
Description=IoT Energy system
After=network.target

[Service]
User=pi
WorkingDirectory=/home/pi/iot_energy
ExecStart=/home/pi/iot_env/bin/python /home/pi/iot_energy/data_acq.py
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
```

Додатково у системі передбачено автоматичний запуск веб-сервера Flask через окремий сервіс *app.service* (Таблиця 3.2), що гарантує доступність інтерфейсу для користувача відразу після завантаження Orange Pi. Такий підхід мінімізує потребу у ручному адмініструванні та підвищує рівень відмовостійкості. Завдяки використанню механізмів автоматизації на рівні операційної системи розроблена IoT-платформа набуває властивостей автономності, самовідновлення та стабільної роботи упродовж тривалого часу

без обов'язкового нагляду оператора. Це, у свою чергу, значно підвищує практичну цінність системи та відповідає вимогам до сучасних edge-рішень у сфері моніторингу та прогнозування енергетичних параметрів.

Таблиця 3.2 Код конфігураційного файлу *app.service*

```
[Unit]
Description=IoT Energy API
After=network.target
[Service]

User=pi
WorkingDirectory=/home/pi/iot_energy
Environment=PATH=/home/pi/iot_env/bin
ExecStart=/home/pi/iot_env/bin/gunicorn -b 0.0.0.0:8000 app:app
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
```

А також після створення цих файлів конфігурації необхідно прописати команди (Таблиця 3.3) в системі, для того щоб запустити автоматизацію.

Таблиця 3.3 Команди для початку роботи автоматизації.

```
sudo systemctl daemon-reload
sudo systemctl enable iot_data.service
sudo systemctl start iot_data.service
sudo systemctl enable iot_api.service
sudo systemctl start iot_api.service
```

3.4 Готова система прогнозування та шляхи її реалізації

Розроблена IoT-система моніторингу та прогнозування електроспоживання на базі одноплатного комп'ютера Orange Pi Zero 3 становить собою комплексний інструмент для локального збирання, оброблення та аналізу

енергетичних даних. Її функціонал визначається взаємодією модулів збору інформації, системи збереження, аналітичних алгоритмів та інтегрованого веб-інтерфейсу, що разом забезпечують безперервний цикл оброблення енергетичних параметрів у режимі реального часу.

Система дозволяє здійснювати автоматичне вимірювання напруги, струму, миттєвої потужності та спожитої енергії за допомогою датчика PZEM-004T з подальшим збереженням отриманих значень у локальній базі SQLite. Завдяки реалізації алгоритмів очищення та валідації даних система забезпечує високий рівень достовірності підсумкових параметрів. Інтегрована модель прогнозування на основі методу ковзного середнього створює можливість формування короткострокових оцінок електроспоживання, що дозволяє користувачеві аналізувати тенденції та зміну навантаження. Веб-інтерфейс на Flask надає доступ до актуальних, історичних та прогнозних даних у структурованому та візуалізованому вигляді, що суттєво підвищує зручність взаємодії.

3.4.1 Можливості, переваги та недоліки готового рішення

Серед основних переваг системи слід відзначити передусім її автономності, яка досягається завдяки використанню концепції edge computing. Усі обчислення виконуються локально, без потреби в зовнішніх серверних ресурсах, що зменшує залежність від мережевої інфраструктури й підвищує надійність функціонування. Низька вартість апаратної платформи та простота застосованих методів роблять систему доступною для широкого кола користувачів, включаючи побутові й навчальні застосування. Важливою перевагою є модульна структура, що дозволяє легко змінювати або оновлювати окремі компоненти системи без суттєвих змін у загальній архітектурі.

Разом із тим, система має й певні недоліки, зумовлені як її апаратними обмеженнями, так і вибраними методами. Обчислювальні можливості Orange Pi Zero 3 є обмеженими, що знижує потенціал реалізації складніших алгоритмів прогнозування, таких як ARIMA, LSTM чи Prophet. Використання методів

простого згладжування не дає високої точності у випадках значних коливань навантаження або наявності сезонних компонентів у сигналі.

Крім того, точність вимірювань суттєво залежить від коректності роботи датчика PZEM-004T, який може мати систематичні похибки й потребує періодичної калібрації.

Незважаючи на зазначені недоліки, система повністю забезпечує виконання визначених функцій та може розглядатися як ефективне рішення для реалізації локального моніторингу та прогнозування у малих енергетичних мережах, лабораторних установках та побутових умовах.

3.4.2 Отримання вигоди завдяки використанню системи

Використання локальної системи моніторингу та прогнозування електроспоживання дає змогу отримати низку практичних вигод, що проявляються як у оптимізації енергоспоживання, так і в загальному підвищенні ефективності використання електричної мережі. Завдяки регулярному накопиченню даних критично важливою перевагою є можливість детального аналізу профілю навантаження, що дозволяє ідентифікувати періоди пікового споживання, нераціонального використання енергії або потенційних перевантажень мережі. Зменшення пікових навантажень та оптимізація режимів роботи електроприладів може прямо вплинути на зменшення побутових витрат, а також витрат домогосподарства або підприємства.

Прогнозування електроспоживання відкриває можливість впровадження адаптивних стратегій керування навантаженням. Наприклад, при прогнозуванні підвищеного споживання користувач може перенести роботу енергоємних приладів на інший час, що зменшить навантаження на мережу та покращить енергоефективність. Такий підхід особливо актуальний у контексті тарифів, диференційованих за періодами доби, що дозволяє безпосередньо зменшити витрати за рахунок планування споживання у години меншого тарифного навантаження.

Система також сприяє підвищенню надійності локальних енергетичних

мереж. Регулярний моніторинг дає змогу своєчасно виявляти аномалії, такі як падіння напруги, нестабільні коливання струму або раптові стрибки потужності, що можуть сигналізувати про пошкодження електропроводки, несправність обладнання, небезпечні режими його роботи або нестабільність у енергетичній системі. Завдяки цьому користувач має можливість завчасно вживати заходів щодо діагностики та ремонту, запобігаючи аварійним ситуаціям.

Для малих підприємств або приватних домогосподарств система може слугувати інструментом аудиту енергоспоживання. Її застосування дозволяє проводити оцінку ефективності встановлених електроприладів, аналізувати доцільність їх роботи в певні періоди, а також приймати рішення щодо модернізації енергетичного обладнання.

Таким чином, практична вигода від системи проявляється як у зменшенні витрат, так і у підвищенні рівня безпеки, енергоефективності та плановості у використанні електричної енергії.

3.4.3 Перспективи для використання та масштабування рішення

Розроблена система має значний потенціал масштабування та може бути адаптована до різних сфер застосування. Її модульна структура, відсутність прив'язки до конкретного контексту та використання відкритих технологій роблять архітектуру гнучкою і придатною для подальшого розвитку. У перспективі система може бути розширена шляхом інтеграції декількох датчиків електроспоживання, що дозволить здійснювати моніторинг не лише загального навантаження, але й окремих електричних контурів, окремих приміщень або навіть конкретних споживачів енергії.

Важливим напрямом масштабування є удосконалення алгоритмів прогнозування. Використання методів машинного навчання, таких як ARIMA, SARIMA або нейронні мережі типу LSTM, дозволить суттєво підвищити точність прогнозів, зокрема у випадках сезонних або нерегулярних коливань навантаження. Хоча такі моделі потребують більших обчислювальних ресурсів, можливим шляхом розвитку є перенесення частини аналітичних обчислень на

хмарні платформи або використання кращих за продуктивністю edge-пристроїв наступного покоління, наприклад Orange Pi 5 або Raspberry Pi 5.

Система також може бути інтегрована в більш широкі енергетичні екосистеми, включаючи розумні мережі (Smart Grid), системи домашньої автоматизації та керування енергоспоживанням (Home Energy Management Systems). У цьому випадку вона стає складовою частиною розподіленої інфраструктури, яка забезпечує інтелектуальне керування ресурсами на рівні будинку, кварталу чи підприємства. Можлива реалізація автоматичного керування окремими приладами на основі прогнозних даних, що відкриває шлях до створення повністю автономних енергетичних систем.

Окремим напрямом розвитку є комерціалізація рішення. Завдяки низькій вартості компонентів та простоті розгортання система може бути використана як готовий продукт для приватних споживачів, малих підприємств, освітніх установ або енергетичних компаній. Її функціонал може бути доповнений модулем енергетичних рекомендацій, аналізом ефективності роботи обладнання або системами сповіщення про аномальні режими. Таким чином, розроблена система має значні перспективи подальшого застосування, розвиток яких може суттєво розширити її функціональність та практичну цінність у сфері моніторингу й керування електроспоживанням.

ВИСНОВКИ

У ході роботи було обґрунтовано актуальність та сформульовано архітектуру edge-орієнтованої IoT-системи, що забезпечує збір, локальне збереження, аналіз і прогнозування параметрів електроспоживання на базі одноплатного комп'ютера Orange Pi Zero 3 та вимірювального модуля PZEM-004T. Вибір апаратної платформи та сенсора базувався на критеріях доступності, економічності й сумісності з інтерфейсами збору даних. Зокрема в роботі показано, що Orange Pi Zero 3 має достатній набір периферійних інтерфейсів (UART, GPIO, Wi-Fi) і обчислювальних ресурсів для виконання базових edge-функцій, а PZEM-004T задовольняє вимоги за інструментальною точністю для побутових сценаріїв моніторингу.

Практична реалізація прототипу охоплювала розробку та інтеграцію основних програмних компонентів: модулю збору даних *data_acq.py*, що взаємодіє з датчиком через Modbus/Serial (бібліотека *minimalmodbus*) та допускає режим симуляції для налагодження, модуля роботи з локальною базою даних *db.py* на основі SQLite для ефективного збереження часових рядів, модуля прогнозування *forecast.py*, що реалізує метод простого ковзного середнього (SMA) для формування короткострокових оцінок, а також веб-інтерфейсу *app.py*, що надає REST-API та інтерактивні графіки через Plotly/Flask. Така модульна організація забезпечує розділення обов'язків, простоту тестування та можливість поетапного розгортання у польових умовах.

Результати апробації підтвердили практичну доцільність обраної концепції edge-обробки у контексті завдань короткострокового прогнозування. Метод SMA обрано як компроміс між простотою реалізації, малими витратами обчислювальних ресурсів та достатньою стабільністю прогнозів для побутових сценаріїв. Функція *compute_and_store()* у модулі *forecast.py* демонструє, що навіть на малопотужному пристрої можливе регулярне формування прогнозів і їхнє збереження у базі для подальшого аналізу й відображення у веб-інтерфейсі. Водночас проведений аналіз показав обмеження цього підходу: SMA не враховує сезонні чи циклічні компоненти та найменше точна при різких

флуктуаціях навантаження, що потребує обережного трактування прогнозних результатів у критичних застосуваннях.

Практичне застосування розробленої системи дає кілька очевидних переваг. По-перше, автономність обробки на периферії знижує витрати на мережевий трафік і мінімізує залежність від хмарних сервісів, що є важливою перевагою в умовах нестабільного інтернет-зв'язку. По-друге, доступ до локальної бази даних (SQLite) і веб-інтерфейсу надає простий і зрозумілий інструментарій для моніторингу та оперативних рішень (наприклад, планування роботи енергоємних приладів). По-третє, відносно низька собівартість апаратної частини робить рішення придатним для масового впровадження у житловому секторі, навчальних закладах та малому бізнесі, де немає потреби у складних моделях і великій обчислювальній інфраструктурі. Такий набір вигод знайшов відображення у проведених аналітичних розділах і практичних рекомендаціях для впровадження системи.

Разом з тим, у роботі детально проаналізовано обмеження та ризики готового рішення. Найважливішими з них є апаратні обмеження платформи Orange Pi Zero 3, що обмежують застосування ресурсомістких моделей (наприклад ARIMA у великих вибірках, LSTM-мережі), а також чутливість результатів до якості вимірювань датчика PZEM-004T, який потребує періодичної калібрації та перевірки коректності підключення. Практично важливо також враховувати ризики збою сервісів та пошкодження SD-карт як єдиного носія даних. У зв'язку з цим в роботі рекомендовано використовувати *systemd*-сервіси для забезпечення автоматичного запуску та відновлення критичних процесів (*iot_data.service*, *iot_api.service*) і передбачити механізми регулярного бекапу бази даних.

На базі проведених досліджень і апробації запропоновано конкретні практичні рекомендації для впровадження та подальшого розвитку системи. По-перше, перед розгортанням у виробничих умовах рекомендується виконати калібрування PZEM-модулів та верифікацію серійного інтерфейсу, а також налаштувати права доступу до портів UART/USB для безперебійного читання даних. По-друге, для підвищення точності прогнозування доцільно застосувати

гібридну стратегію: обчислення базових короткострокових прогнозів локально (SMA) і періодична передача зведених даних у централізовану систему або хмару для навчання більш складних моделей та надання коригувальних прогнозів. По-третє, слід передбачити політики збереження та архівації даних (ротація SQLite-файлу, збереження архівів на зовнішньому ресурсі) та моніторингу стану сервісів для підвищення надійності.

Перспективи подальшого розвитку охоплюють апаратне та алгоритмічне масштабування: перехід на більш продуктивні платформи, наприклад Orange Pi 5 для виконання ресурсомістких алгоритмів на периферії, інтеграцію додаткових сенсорів, таких як контури, потужності окремих пристроїв для детального енергомоніторингу та застосування методів машинного навчання (ARIMA/SARIMA, LSTM) у гібридній архітектурі з хмарним компонентом. Реалізація таких напрямів дозволить підвищити точність прогнозів, впровадити автоматичні правила керування навантаженням та створити комерційно привабливі сервіси енергоменеджменту.

Підсумовуючи, проведене дослідження підтверджує, що запропонована edge-IoT архітектура і реалізований прототип становлять збалансоване рішення для локального моніторингу та короткострокового прогнозування електроспоживання в умовах обмежених ресурсів. Рекомендована гібридна стратегія подальшого розвитку гарантує збереження переваг автономності й конфіденційності при одночасному підвищенні точності прогнозів за рахунок централізованих обчислень. Отримані результати мають практичну цінність для впровадження у побутових та малих комерційних застосуваннях в Україні, де питання оперативності, вартості і незалежності від хмарних сервісів є критично важливими.

ПЕРЕЛІК ПОСИЛАНЬ

1. Gillis A., Yasar K. What is IoT (internet of things). *techtarget*. URL: <https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT>.
2. Internet of things market size, share & outlook. *mordorintelligence*. URL: <https://www.mordorintelligence.com/industry-reports/internet-of-things-iot-market>.
3. Rosenberg C. Coke machine. *cs.cmu.edu*. URL: <https://www.cs.cmu.edu/~coke/>.
4. Jena S. Architecture of internet of things (iot). *geeksforgeeks*. URL: <https://www.geeksforgeeks.org/computer-networks/architecture-of-internet-of-things-iot/>.
5. Decentralized iot networks – how blockchain is transforming device interactions. *medium*. URL: <https://medium.com/@RocketMeUpIO/decentralized-iot-networks-how-blockchain-is-transforming-device-interactions-efe00a9243ad>.
6. Almudayni Z. Energy inefficiency in iot networks: causes, impact, and a strategic framework for sustainable optimisation. *MDPI*. URL: <https://www.mdpi.com/2079-9292/14/1/159>.
7. Bornare A. Difference between edge computing and cloud computing - geeksforgeeks. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/cloud-computing/difference-between-edge-computing-and-cloud-computing/>.
8. Lambropoulos G. Single-Board computer, edge computing and virtualization technology. *encyclopedia*. URL: <https://encyclopedia.pub/entry/55359>.
9. lattepanda. SBC - edge computing: 6 uses cases for industrial iot. *lattepanda*. URL: <https://www.lattepanda.com/blog-308811.html>.
10. Pérez-Bedmar J. Four edge computing applications in the electrical sector. *Barbara*. URL: <https://www.barbara.tech/blog/four-edge-computing-applications-energy-sector>.
11. Rgcllegg. Autoregressive integrated moving average. *Wikipedia*. URL: https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average.

12. Hyndman R., Athanasopoulos G. ARIMA vs ETS. *otexts*. URL: <https://otexts.com/fpp2/arima-ets.html>.
13. Serana. Time series modelling – ARIMA and ETS. *medium*. URL: <https://medium.com/@serana.ai/time-series-modelling-arima-and-ets-cafc904b9183>.
14. Gao Y. A comparative study of ARIMA and ETS models for time series forecasting. *researchgate*. URL: https://www.researchgate.net/publication/387716971_A_Comparative_Study_of_ARIMA_and_ETS_Models_for_Time_Series_Forecasting.
15. Comparison of ARIMA, ETS, NNAR, TBATS and hybrid models to forecast the second wave of COVID-19 hospitalizations in Italy - The European Journal of Health Economics. *SpringerLink*. URL: <https://link.springer.com/article/10.1007/s10198-021-01347-4>.
16. Wikimedia. Exponential smoothing. *Wikipedia*. URL: https://en.wikipedia.org/wiki/Exponential_smoothing.
17. Hyndman R., Athanasopoulos G. What is the difference between ARIMA and ARMA model?. *GitHub*. URL: <https://r4ds.github.io/bookclub-fpp/what-is-the-difference-between-arima-and-arma-model.html>.
18. Jain G., Mallick B. A study of time series models ARIMA and ETS. *mecs-press*. URL: <https://www.mecs-press.org/ijmecs/ijmecs-v9-n4/IJMECS-V9-N4-7.pdf>.
19. Wikimedia, Elpupiok. Autoregressive moving-average model. *Wikipedia*. URL: https://en.wikipedia.org/wiki/Autoregressive_moving-average_model.
20. Using ARIMA and ETS models for forecasting water level changes for sustainable environmental management / T. Agaj et al. *Nature*. URL: <https://www.nature.com/articles/s41598-024-73405-9>.
21. World energy outlook 2025. *iea*. URL: <https://www.iea.org/reports/world-energy-outlook-2025>.
22. Eurostat. Energy statistics - an overview. *ec.europa*. URL: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Energy_statistics_-_an_overview.
23. Ritchie H., Rosado P., Roser M. Energy production and consumption. *Our*

- World in Data*. URL: <https://ourworldindata.org/energy-production-consumption>.
24. Rangelova K., Wiatros-Motyka M. Global electricity mid-year insights 2025. *ember-energy*. URL: <https://ember-energy.org/latest-insights/global-electricity-mid-year-insights-2025/>.
25. Campilho R. Energy consumption analysis and efficiency enhancement in manufacturing companies. *mdpi*. URL: <https://www.mdpi.com/2075-1702/11/10/939>.
26. Filipenco D. Global energy consumption patterns: statistics and analysis. *DevelopmentAid*. URL: <https://www.developmentaid.org/news-stream/post/187892/global-energy-consumption>.
27. Global energy review 2025. *iea*. URL: <https://www.iea.org/reports/global-energy-review-2025/global-trends>.
28. A detailed analysis of electricity consumption. *sciencedirect*. URL: <https://www.sciencedirect.com/science/article/pii/S0378778823002761>.
29. Ukraine efficiency demand. *iea*. URL: <https://www.iea.org/countries/ukraine/efficiency-demand>.
30. Dixi Group. Step-by-Step plan for energy efficient reconstruction in ukraine. *irf*. URL: <https://www.irf.ua/en/pokrokovyj-plan-energoefektyvnoyi-vidbudovy-v-ukrayini/>.
31. Chen Y.-T. The factors affecting electricity consumption and the consumption characteristics in the residential sector - A case example of taiwan. *mdpi*. URL: <https://www.mdpi.com/2071-1050/9/8/1484>.
32. Naveen I., Naveen E. An examination of factors affecting residential energy consumption using a multiple discrete continuous approach. *sciencedirect*. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0378778821002188>.
33. Factors affecting urban and rural household electricity demand in south sulawesi province, indonesia. *econjournals*. URL: <https://econjournals.com/index.php/ijeep/article/view/21226/9363>.
34. Bandarra P., Valdez M., Pereira A. Solutions for monitoring and analysing for

- energy consumption – Energy management systems. *researchgate*. URL: https://www.researchgate.net/publication/321986431_Solutions_for_monitoring_and_analysing_for_energy_consumption_-_Energy_management_systems.
35. Energy monitoring system - iot based devices - power saving solutions. *Energyly*. URL: <https://www.energyly.com/energy-monitoring.html?srsltid=AfmBOoqL5-YzoMPb2CdCUtshhj6RG62ea1HLRQ5RKxoxVSJNsgy1mpeJ>.
36. Energy monitoring & management software | E360 by sanalife. *Sanalife*. URL: <https://www.sanalifeenergy.com/energy-monitoring-software>.
37. Wikimedia. ClearVUE.Zero. *Wikipedia, the free encyclopedia*. URL: <https://en.wikipedia.org/wiki/ClearVUE.Zero>
38. Ucar M., Kaygusuz A. Short-Term energy consumption forecasting analysis using different optimization and activation functions with deep learning models. *mdpi*. URL: <https://www.mdpi.com/2076-3417/15/12/6839>.
39. Comparison of short-term forecasting methods of electricity consumption in microgrids / V. Shendryk et al. *ric*. URL: <https://ric.zntu.edu.ua/article/view/274449>.
40. Comparative analysis of models for short-term forecasting of electricity consumption / M. Korablyov et al. *openarchive*. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/cc1b7967-5866-4746-abe8-5843f3eb6b87/content>.
41. Misiurek K., Olkuski T., Zyśk J. Review of methods and models for forecasting electricity consumption. *mdpi*. URL: <https://www.mdpi.com/1996-1073/18/15/4032>.
42. Tokhmetov A. Comparative analysis of various forecast models of electricity consumption in smart buildings. *researchgate*. URL: https://www.researchgate.net/publication/390472367_COMPARATIVE_ANALYSIS_OF_VARIOUS_FORECAST_MODELS_OF_ELECTRICITY_CONSUMPTION_IN_SMART_BUILDINGS.

ДОДАТКИ

```

# db.py – база даних
import sqlite3
from datetime import datetime

def init_db(path):
    conn = sqlite3.connect(path, check_same_thread=False)
    cur = conn.cursor()
    cur.execute('PRAGMA journal_mode=WAL;')
    cur.execute("""
CREATE TABLE IF NOT EXISTS readings (
    ts TEXT PRIMARY KEY,
    voltage REAL,
    current REAL,
    power REAL,
    energy REAL
)""")
    cur.execute("""
CREATE TABLE IF NOT EXISTS forecasts (
    ts TEXT,
    horizon INTEGER,
    forecast_value REAL,
    method TEXT,
    PRIMARY KEY (ts, horizon, method)
)""")
    conn.commit()
    return conn

def insert_reading(conn, ts_iso, voltage, current, power, energy):
    cur = conn.cursor()
    cur.execute("""
INSERT OR REPLACE INTO readings(ts, voltage, current, power, energy)
VALUES (?, ?, ?, ?, ?)
""", (ts_iso, voltage, current, power, energy))
    conn.commit()

def get_last_n_powers(conn, n):
    cur = conn.cursor()
    rows = cur.execute("""
SELECT ts, power FROM readings
ORDER BY ts DESC
LIMIT ?
""", (n,)).fetchall()
    return rows[::-1] # повернути у хронологічному порядку

```

```

def insert_forecast(conn, base_ts, horizon, value, method):
    cur = conn.cursor()
    cur.execute("""
        INSERT OR REPLACE INTO forecasts(ts, horizon, forecast_value, method)
        VALUES (?, ?, ?, ?)
    """, (base_ts, horizon, value, method))
    conn.commit()

def get_latest_reading(conn):
    cur = conn.cursor()
    row = cur.execute('SELECT ts, voltage, current, power, energy FROM readings
ORDER BY ts DESC LIMIT 1').fetchone()
    return row

def get_history(conn, limit=1440):
    cur = conn.cursor()
    rows = cur.execute('SELECT ts, voltage, current, power, energy FROM readings
ORDER BY ts DESC LIMIT ?', (limit,)).fetchall()
    return rows[::-1]

```

```

# forecast.py – ковзне середнє
import numpy as np
from db import get_last_n_powers, insert_forecast
from datetime import datetime, timedelta

def sma_forecast(series, window, steps):
    """
    series: list of numeric values (стапі -> нові)
    window: N для SMA
    steps: кількість кроків вперед
    повертає list length=steps
    """
    if len(series) < window:
        # якщо замало даних — використовуємо середнє всіх наявних
        avg = float(np.mean(series)) if len(series) > 0 else 0.0
        return [avg] * steps
    avg = float(np.mean(series[-window:]))
    return [avg] * steps

def compute_and_store(conn, sma_window, steps, method_name="SMA"):
    # беремо останні sma_window значень або трохи більше
    rows = get_last_n_powers(conn, max(sma_window, 24))
    series = [r[1] for r in rows] # r = (ts, power)
    base_ts = datetime.utcnow().isoformat()
    preds = sma_forecast(series, sma_window, steps)
    # зберігаємо кожен прогноз з індексом горизонту

```

```

for h, val in enumerate(preds, start=1):
    insert_forecast(conn, base_ts, h, float(val), method_name)
return base_ts, preds

```

```

# data_acq.py - збір даних з PZEM
import time
import yaml
from datetime import datetime
from db import init_db, insert_reading
import random

# optional hardware lib
try:
    import minimalmodbus
    HAS_MODBUS = True
except ImportError:
    HAS_MODBUS = False

# завантажити конфіг
with open('config.yaml') as f:
    cfg = yaml.safe_load(f)

DB_PATH = cfg['database']['path']
SERIAL_PORT = cfg['device']['serial_port']
BAUDRATE = cfg['device'].get('baudrate', 9600)
SLAVE_ADDR = cfg['device'].get('slave_addr', 1)
SENSOR_MODE = cfg['device'].get('sensor_mode', 'auto')
INTERVAL = cfg['sampling']['interval_seconds']

conn = init_db(DB_PATH)

def read_pzem_sim():
    v = 230.0 + random.uniform(-5.0, 5.0)
    i = random.uniform(0.1, 2.0)
    p = v * i
    energy = random.uniform(0.0, 0.01)
    return round(v,2), round(i,3), round(p,2), round(energy,4)

def read_pzem_modbus(port, slave):
    """
    Простий приклад з minimalmodbus.
    Значч: карти регістрів PZEM бувають різні. Якщо результати дивні - треба
    змінити адресу регістрів.
    Тут наведено базовий приклад. Якщо minimalmodbus не встановлено або
    пристрій інший - користуйся simulate.
    """

```

```

inst = minimalmodbus.Instrument(port, slave)
inst.serial.baudrate = BAUDRATE
inst.serial.timeout = 1
# !!! Перевір і відкоригуй адреси регістрів під свій модуль PZEM !!!
# Нижче - типовий приклад (потрібно протестувати):
try:
    voltage = inst.read_register(0, 1) # регістр 0 -> напруга (у одиницях 0.1)
    current = inst.read_register(1, 3) # регістр 1 -> струм (може мати множник)
    power = inst.read_register(3, 1) # регістр 3 -> потужність
    energy = inst.read_register(4, 1) # регістр 4 -> енергія
    return float(voltage), float(current), float(power), float(energy)
except Exception as e:
    print("Modbus read error:", e)
    raise

def read_pzem():
    if SENSOR_MODE == 'simulate' or not HAS_MODBUS:
        return read_pzem_sim()
    else:
        return read_pzem_modbus(SERIAL_PORT, SLAVE_ADDR)

def main():
    print("Starting data_acq, interval:", INTERVAL)
    while True:
        try:
            v, i, p, e = read_pzem()
            ts = datetime.utcnow().isoformat()
            # базова валідація
            if p < 0 or v <= 0:
                # ігноруємо помилкові значення
                print("Bad reading, skipping:", v, i, p, e)
            else:
                insert_reading(conn, ts, v, i, p, e)
                print(f"{ts} | V={v} V, I={i} A, P={p} W, E={e} kWh")
        except Exception as ex:
            print("Error reading sensor:", ex)
        time.sleep(INTERVAL)

if __name__ == "__main__":
    main()

```

```

# app.py - веб-інтерфейс Flask
from flask import Flask, jsonify, render_template_string, request
import yaml
from db import init_db, get_latest_reading, get_history, get_last_n_powers
from forecast import sma_forecast

```

```

from datetime import datetime
import os

with open('config.yaml') as f:
    cfg = yaml.safe_load(f)

DB_PATH = cfg['database']['path']
SMA_WINDOW = cfg['prediction']['sma_window']
FORECAST_STEPS = cfg['prediction']['forecast_steps']

app = Flask(__name__)
conn = init_db(DB_PATH)

@app.route('/api/latest')
def api_latest():
    row = get_latest_reading(conn)
    if not row:
        return jsonify({})
    ts, v, i, p, e = row
    return jsonify({'ts': ts, 'voltage': v, 'current': i, 'power': p, 'energy': e})

@app.route('/api/history')
def api_history():
    limit = int(request.args.get('limit', 1440))
    rows = get_history(conn, limit)
    data = [{'ts': r[0], 'voltage': r[1], 'current': r[2], 'power': r[3], 'energy': r[4]} for r in
rows]
    return jsonify(data)

@app.route('/api/forecast')
def api_forecast():
    rows = get_last_n_powers(conn, max(SMA_WINDOW, 24))
    series = [r[1] for r in rows]
    preds = sma_forecast(series, SMA_WINDOW, FORECAST_STEPS)
    base_ts = datetime.utcnow().isoformat()
    # повертаємо кроки з часовими мітками (припускаємо інтервал агрегації у
хвилинах)
    interval_minutes = int(cfg['sampling']['aggregation_seconds'] // 60) or 1
    times = []
    from datetime import timedelta
    t0 = datetime.utcnow()
    for h in range(1, FORECAST_STEPS+1):
        times.append((t0 + timedelta(minutes=h*interval_minutes)).isoformat())
    return jsonify({'base_ts': base_ts, 'times': times, 'preds': preds})

# Простий HTML дашборд з plotly (з pull даних через API)

```

```

DASH_HTML = """
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>IoT Energy Dashboard</title>
  <script src="https://cdn.plot.ly/plotly-latest.min.js"></script>
</head>
<body>
  <h2>Поточне споживання</h2>
  <div id="current"></div>
  <h2>Графік потужності (історія)</h2>
  <div id="hist" style="width:100%;height:400px;"></div>
  <h2>Прогноз (SMA)</h2>
  <div id="forecast" style="width:100%;height:300px;"></div>

<script>
async function loadData(){
  const latest = await fetch('/api/latest').then(r=>r.json());
  document.getElementById('current').innerText = JSON.stringify(latest);

  const hist = await fetch('/api/history?limit=200').then(r=>r.json());
  const x = hist.map(r=>r.ts);
  const y = hist.map(r=>r.power);
  Plotly.newPlot('hist',[{x:x,y:y,type:'scatter',name:'power'}],{margin:{t:20}});

  const fc = await fetch('/api/forecast').then(r=>r.json());
  Plotly.newPlot('forecast',[
    {x:fc.times,y:fc.preds,type:'scatter',name:'forecast'}
  ],{margin:{t:20}});
}

loadData();
setInterval(loadData, 30000); // оновлювати кожні 30 сек
</script>
</body>
</html>
"""

@app.route('/')
def index():
  return render_template_string(DASH_HTML)

if __name__ == '__main__':
  app.run(host=cfg['web']['host'], port=int(cfg['web']['port']), debug=False)

```

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)

Актуальність теми: У сучасному світі в останній час зростає попит на електроенергію. Інтеграція відновлюваних джерел енергії та необхідність підвищення енергоефективності ставлять перед людьми завдання точного моніторингу, аналізу та прогнозування енергоспоживання. Україна що переживає періоди нестабільності енергопостачання та активну модернізацію інфраструктури, проблеми прогнозування і оперативного реагування набувають особливого значення. Вони пов'язані з підвищенням надійності постачання, оптимізацією тарифів та зниженням витрат на енергоресурси.

Наукова новизна: Наукова новизна роботи виражена в описі нового підходу до локального прогнозування енергоспоживання на edge-пристроях: поєднання моделей прогнозування, оптимізованих під ресурси одноплатних комп'ютерів з архітектурними рішеннями для апаратної обробки з можливістю перетворення в гібридну.

Об'єкт дослідження: Об'єктом дослідження є процеси збору, обробки та прогнозування даних про споживання електроенергії з використанням IoT-пристроїв.

Предмет дослідження: Предметом дослідження є архітектурні і алгоритмічні підходи до локальної обробки та прогнозування енергоспоживання, а також методи апробації їх ефективності на одноплатних комп'ютерах.

Мета дослідження: Розробити і обґрунтувати архітектуру IoT-системи для збору, аналізу та прогнозування споживання електроенергії із застосуванням edge computing, яка забезпечить локальну обробку даних, адекватну точність короткострокових, середньострокових прогнозів та можливість масштабування для застосування в побутових або малих комерційних середовищах.

Завдання дослідження:

1. Аналіз проблеми збору та прогнозування енергоспоживання, доступні апаратні й програмні платформи для edge-обчислень.
2. Огляд і порівняльний аналіз методів прогнозування часових рядів, придатних для виконання на edge-пристроях (ковзні середні, ARIMA, прості регресійні моделі, легкі ML-моделі).
3. Розробка архітектури IoT-системи на базі одноплатного комп'ютера для збору й локальної обробки даних з приладів обліку.

2

ПРОЕКТУВАННЯ ТА РОЗРОБКА СИСТЕМИ ПРОГНОЗУВАННЯ

Вибір апаратних та програмних компонентів для розробки

Плата - Orange pi zero 3

Переваги

- Розмір - компактна
- Ціна - недорога по співвідношенню ціна/продуктивність
- Програмовість - має чудові можливості для налаштувань
- Розширення - має порти для розширення функціонала

Недоліки

- Потужність - для більш складних моделей не є достатнім рішенням в області продуктивності
- Живлення - плата не має вмонтованого живлення і потребує зовнішнє



3

Компонти системи

Датчик струму



Для вимірювання електричного струму системі необхідний відповідний датчик. В якості такого виступить модуль вимірювання параметрів споживання змінного струму PZEM-004T, призначений для вимірювання вхідної напруги, споживаного навантаженням струму. В комплекті з датчиком поставляється кільце для виміру напруги. В даному випадку воно зі швидким доступом, що дозволить мати здатність вимірювати будь який пристрій.

Карта пам'яті

Для роботоспроможності системи необхідне встановлення програмного забезпечення. Оскільки плата orange pi zero 3 не має на борту вбудованої пам'яті нам необхідно буде використовувати карту пам'яті microSD. В даній кваліфікаційній роботі буде використовуватись карта пам'яті microSD від компанії SanDisk моделі Ultra об'ємом 32 gb. Вона має непогані характеристики і невисоку ціну, та стане чудовим вибором.



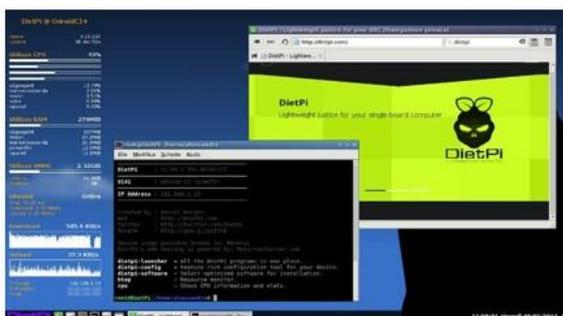
Живлення



Для функціонування системи потрібна електроенергія. Можна використати будь який блок живлення що працює від мережі. Головне щоб блок відповідав необхідним для системи характеристиками, а саме: штекер повинен бути, а також видавати напругу 5 V та струм 3 A, що в сумі дають 15 W. Альтернативою блоку живлення може стати power bank у зв'язці з кабелем type-c. Ще один спосіб заживити систему - це інтегрувати в неї аккумулятори 18650, підвищуючий перетворювач MT3608 та засунути це все в корпус.

4

Операційна система



Для функціонування системи на одноплатному комп'ютері потрібно встановити низку програмних компонентів, для подальшого запуску елементів прогнозування. Першим компонентом стане операційна система DietPI - дистрибутив заснований на linux полегшений та оптимізований для роботи саме на одноплатних комп'ютерів

Для того щоб система могла збирати, групувати та обробляти дані їй окрім операційної системи також необхідні інструменти. В якості такого виступить Python. Легка і ефективна мова програмування для реалізації системи.

Плюси

- Легка - не потребує багато ресурсів для функціонування
- Ефективна - добре розподіляє і оптимізує процеси
- Функціональна – підтримує всі необхідні компоненти

Мінуси

- Лінійкоідна – потребує початкового налаштування

5

Метод прогнозування

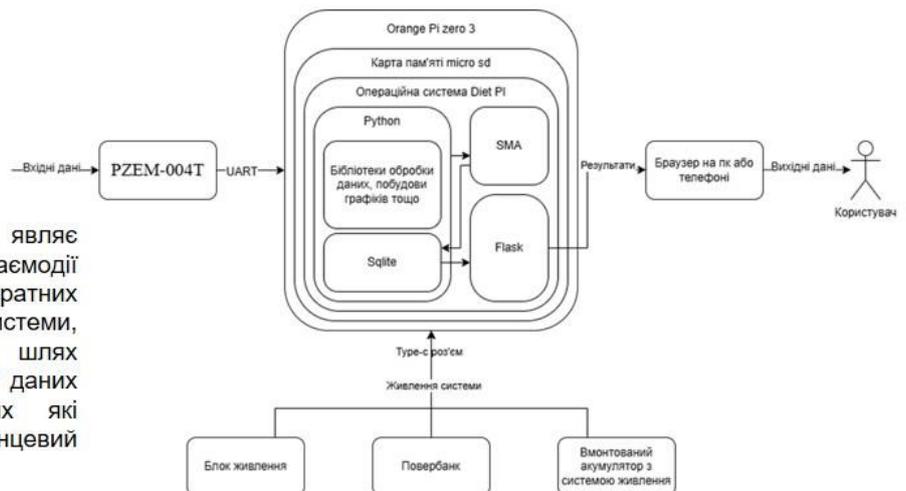
Для прогнозування в системі використовується SMA (Simple Moving Average) - це інструмент, що використовується для обчислення та аналізу точок даних шляхом створення серії середніх значень різних вибірок з повного набору даних. В даному випадку буде використовуватись простий підвид моделі. Його буде цілком достатньо для системи і своїми обчисленнями вона не буде перевантажувати систему.

$$SMA_k = \frac{1}{k} \sum_{i=n-k+1}^n p_i$$

6

Архітектурна схема

Архітектурна схема являє собою принцип взаємодії програмних та апаратних компонентів системи, показуючи повний шлях проходження вхідних даних до вихідних даних які отримує кінцевий користувач



7

Локальний веб-інтерфейс

Локальний веб-інтерфейс, реалізований у модулі `app.py`, надає функцію доступу користувачеві до вимірених та прогнозних даних. Додаток побудований на основі мікрофреймворку Flask. HTML-частина інтерфейсу формує єдину сторінку, на якій динамічно відображаються графіки та ключові параметри електроспоживання. Побудова графіків здійснюється клієнтською частиною через бібліотеку Plotly. Усі обчислення, такі як прогнозування, агрегація та фільтрація даних виконуються на стороні серверної частини, що функціонує безпосередньо на Orange Pi. Це знижує навантаження на клієнтські пристрої та забезпечує стабільну роботу системи навіть у умовах обмеженого мережевого доступу.



8

Автоматизація роботи системи

Оскільки розроблена IoT-система призначена для тривалого збору й аналізу даних електроспоживання без постійного втручання користувача, важливою складовою її організації є формування механізмів автоматичного запуску, контролю виконання та періодичного оновлення вимірювальної та прогнозної підсистем. Для цього використано вбудовані можливості операційної системи DietPi на базі Linux, зокрема системний менеджер `systemd`, який забезпечує автоматичний запуск ключових програмних модулів під час завантаження пристрою, а також їхнє відновлення у випадку аварійного завершення роботи.

Автоматизація збору даних здійснюється шляхом створення окремого `systemd`-сервісу, що запускає модуль `data_acq.py` у фоновому режимі.

Процедура автоматичного прогнозування реалізована за допомогою планувальника `cron`, що викликає модуль `forecast.py` через певні інтервали часу. Додатково у системі передбачено автоматичний запуск веб-сервера Flask через окремий сервіс `app.service`.



9

Можливості, переваги та недоліки готового рішення

У будь-якої системи сьогодні існують як переваги, так і недоліки. Основні з них які треба виділити такі:

Параметри	1	2	3	4
Переваги	Автономність, яка досягається завдяки використанню концепції edge computing	Усі обчислення виконуються локально, без потреби в зовнішніх серверних ресурсах, що зменшує залежність від мережевої інфраструктури	Низька вартість апаратної платформи та простота застосованих методів роблять систему доступною для широкого кола користувачів.	Модульна структура, що дозволяє легко змінювати або оновлювати окремі компоненти системи без суттєвих змін у загальній архітектурі.
Недоліки	Обчислювальні можливості Orange Pi Zero 3 є обмеженими, що знижує потенціал реалізації складніших алгоритмів прогнозування.	Використання методів простого згладжування не дає високої точності у випадках значних коливань навантаження або наявності сезонних компонентів у сигналі.	Точність вимірювань суттєво залежить від коректності роботи датчика PZEM-004T, який може мати систематичні похибки й потребує періодичної калібрації.	-

Крім того, Незважаючи на зазначені недоліки, система повністю забезпечує виконання визначених функцій та може розглядатися як ефективне рішення для реалізації локального моніторингу та прогнозування у малих енергетичних мережах, лабораторних установках та побутових умовах.

10

ВИСНОВКИ

- Під час написання дипломної роботи було детально розглянуто процес розробки та сформульовано архітектуру edge-орієнтованої IoT-системи, що забезпечує збір, локальне збереження, аналіз і прогнозування параметрів електроспоживання на операційній системі Diet Pi.
- Було проведено та проаналізовано всі етапи розробки, такі як: підбір апаратних та програмних елементів, і їх адаптація для роботи в зв'язці.
- Підсумовуючи, проведене дослідження підтверджує, що запропонована edge-IoT архітектура і реалізований прототип становлять збалансоване рішення для локального моніторингу та короткострокового прогнозування електроспоживання в умовах обмежених ресурсів.
- Рекомендована гібридна стратегія подальшого розвитку гарантує збереження переваг автономності й конфіденційності при одночасному підвищенні точності прогнозів за рахунок централізованих обчислень.
- Отримані результати мають практичну цінність для впровадження у побутових та малих комерційних застосуваннях в Україні.

Апробація

VII ВСЕУКРАЇНСЬКА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ «TELECOMMUNICATION: PROBLEMS AND INNOVATION», 30 грудня 2024 року, ДУІКТ - «Розробка енергоефективних алгоритмів для оптимізації навантаження у мультимарних середовищах»

Дякую за увагу)

11