Форма №Н-6.01

Міністерство освіти і науки України
Рівненський державний гуманітарний університет
Кафедра інформаційних технологій та моделювання

Кваліфікаційна робота
за освітнім ступенем «магістр»
на тему: «Віртуальне проектування рішень Інтернету речей з використанням інструментів Cisco Packet Tracer»

Виконав: магістр 2 курсу
групи М-КН-21
спеціальності 122 «Комп’ютерні науки»
Гуменний Дмитро Сергійович
Керівник:
к.т.н., доцент Шинкарчук Н.В.

Рівне – 2025

АНОТАЦІЯ ДО КВАЛІФІКАЦІЙНОЇ РОБОТИ

Гуменний Д.С. Віртуальне проектування рішень Інтернету речей з використанням інструментів Cisco Packet Tracer. Кваліфікаційна робота на здобуття ступеня «Магістр» за спеціальністю 122 «Комп'ютерні науки» – Рівненський державний гуманітарний університет. Рівне, 2025. 63 с.

Останнім часом, дедалі популярнішою стає концепція Інтернету речей (IoT). Це повноцінна технологічна парадигма, що трансформує повсякденне життя та промислові процеси, об'єднуючи мільярди фізичних пристроїв у єдину глобальну мережу. Такі рішення гарантують автоматизацію процесів, підвищення ефективності систем та новий рівень взаємодії між об'єктами. Це дозволяє організаціям та кінцевим користувачам економити ресурси, забезпечуючи при цьому високу ефективність моніторингу та управління.

Організація систем IoT передбачає наявність інфраструктури, що складається з сенсорів для збору даних (наприклад, температури, вологості, руху), виконавчих механізмів (актуаторів) для виконання дій, мікроконтролерів для локальної обробки та мережевих шлюзів і серверів для централізованого управління та аналізу даних.

В кваліфікаційній роботі проведено огляд концепції Інтернету речей та детально проаналізовано багаторівневу екосистему і архітектуру IoT. Досліджено інструментальну програмну складову симулятора Cisco Packet Tracer як засіб для проєктування та тестування IoT-рішень, детально описано процедуру розробки, програмної реалізації і тестування трьох практичних моделей: системи клімат-контролю, автоматизованої системи пожежогасіння та автономної системи автоматичного поливу.

ANNOTATION TO THE QUALIFICATION PAPER

Humennyi D.S. Virtual Design of Internet of Things Solutions Using Cisco Packet Tracer Tools. Master's thesis for the degree of «Master» in Specialty 122 «Computer Science» – Rivne State Humanitarian University. Rivne, 2025. 63 p.

Recently, the concept of the Internet of Things (IoT) has become increasingly popular. It is a full-fledged technological paradigm that transforms everyday life and industrial processes by uniting billions of physical devices into a single global network. Such solutions ensure process automation, enhanced system efficiency, and a new level of interaction between objects. This allows organizations and end-users to conserve resources while ensuring high efficiency in monitoring and management.

The organization of IoT systems involves an infrastructure consisting of sensors for data collection (e.g., temperature, humidity, motion), actuators for executing actions, microcontrollers for local processing, and network gateways and servers for centralized management and data analysis.

The qualification paper provides an overview of the Internet of Things concept and a detailed analysis of the multi-level ecosystem and architecture of IoT. The software tools of the Cisco Packet Tracer simulator were investigated as a means for designing and testing IoT solutions. The procedure for the development, software implementation, and testing of three practical models is described in detail: a climate control system, an automated fire suppression system, and an autonomous automatic irrigation system.

ЗМІСТ

ВСТУП

Актуальність роботи. Актуальність віртуального проєктування рішень Інтернету речей визначається потребами сучасного ІТ-ринку та необхідністю моделювання і тестування безпечних та ефективних IoT-систем ще до їхнього реального впровадження, для чого використовуються спеціалізовані симулятори, як-от Cisco Packet Tracer.

Мета роботи: дослідити принципи побудови та функціонування систем Інтернету речей і реалізувати процес віртуального проєктування IoT-рішень з використанням симуляційного середовища Cisco Packet Tracer для моделювання взаємодії між пристроями, аналізу мережевих процесів та перевірки ефективності функціонування створеної системи.

Об’єктом дослідження є технологія Інтернету речей, яка проєктуються засобами інструментарію середовищі Cisco Packet Tracer.

Інструмент дослідження. Симулятор і моделювання мережі Cisco Packet Tracer.

Предметом дослідження є інструментальні і функціональні можливості середовища Cisco Packet Tracer. для проєктування рішень рішень Інтернету речей.

Завдання дослідження. Для досягнення поставлених цілей потрібно:

1. Описати концепцію, історію розвитку та сфери впровадження технології Інтернет речей.
2. Розглянути і описати функціональні можливості та інструментарій симулятора Cisco Packet Tracer для проєктування та програмування IoT-пристроїв.
3. Розробити та реалізувати три практичні моделі IoT-систем: систему клімат-контролю, систему пожежогасіння і систему автоматичного поливу.
4. Провести симуляцію та тестування розроблених моделей для перевірки коректності їхньої логіки та аналізу ефективності роботи.

Апробація результатів кваліфікаційної роботи. Результати виконання кваліфікаційної роботи, окремі її аспекти та одержані узагальнення і висновки були оприлюднені на XVIII Всеукраїнській науково-практичній конференції здобувачів вищої освіти та молодих учених «НАУКА, ОСВІТА, СУСПІЛЬСТВО ОЧИМА МОЛОДИХ» (м. Рівне, 2025), звітній науковій конференції викладачів, співробітників і здобувачів вищої освіти Рівненського державного гуманітарного університету за 2024 рік (м. Рівне, 2025).

Публікації. Результати, які були отримані в ході кваліфікаційного дослідження частково опубліковані у вигляді тез XVIII Всеукраїнської науково-практичної конференції здобувачів вищої освіти та молодих учених «НАУКА, ОСВІТА, СУСПІЛЬСТВО ОЧИМА МОЛОДИХ» у Рівненському державному гуманітарному університеті (Додаток А).

Структура роботи. Робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел. Перший розділ присвячений огляду технології Інтернет речей, історії її виникнення, сферам застосування та технологічним основам. У другому розділі розглянуто екосистему IoT і детально проаналізовано аспекти безпеки даних. Третій розділ містить огляд інструментарію Cisco Packet Tracer для моделювання IoT-інфраструктури, включаючи опис пристроїв, сенсорів, актуаторів та можливостей їх програмування. Четвертий розділ – це детальний опис процедури розробки, програмної реалізації та тестування трьох практичних IoT-проектів: системи клімат-контролю, системи пожежогасіння та системи автоматичного поливу. Список літератури містить двадцять два джерела.

РОЗДІЛ 1
КОНЦЕПЦІЯ ІНТЕРНЕТУ РЕЧЕЙ

1.1. Основи технології Інтернет речей

Інтернет речей або Internet of Things (IoT) англійською мовою є концепцією глобальної мережі, яка складається з численних фізичних пристроїв, що мають вбудовані датчики для збирання даних. Окрім датчиків в систему входять також виконавчі механізми, вбудовані у різні об’єкти, які здатні здійснювати певні дії на основі отриманої інформації (рис. 1.1). Основна мета цієї мережі – здійснення автоматичного обміну даними між фізичними об’єктами та комп’ютерними системами за допомогою стандартних протоколів зв’язку.

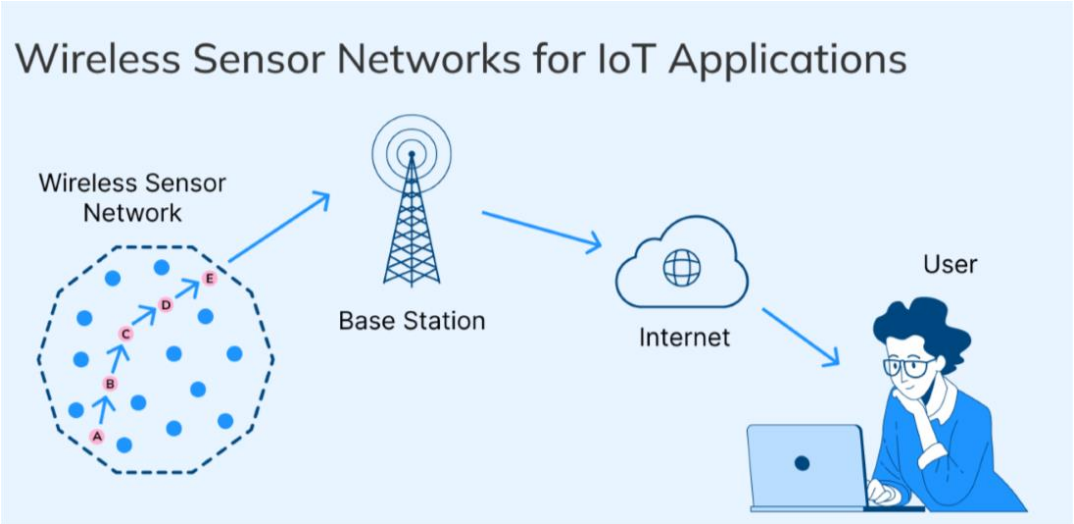

Рисунок 1.1. Приклад передачі даних в IoT

Мережа Інтернету речей включає не лише датчики, а й програмне забезпечення для обробки даних і керування пристроями, що забезпечує автоматизацію процесів. Пристрої можуть зчитувати інформацію, виконувати команди, а також здійснювати ідентифікацію та програмування. Взаємодія між пристроями здійснюється як через проводи, так і бездротовими технологіями, що забезпечує мобільність та ефективність. Концепція «Internet of Everything» розширює можливості підключення не лише стандартних пристроїв, а й фізичних об’єктів

Термін «Інтернет речей» вперше запропонував американський спеціаліст у галузі телекомунікацій Пітер Льюїс у 1985 році. Він описав цю концепцію як «інтеграцію людей, процесів та технологій з пристроями і датчиками, що забезпечують можливість віддаленого моніторингу, управління, оцінки стану та виявлення тенденцій роботи таких пристроїв» [1].

1.2. Галузі впровадження і мережеві технології Інтернету речей

IoT складається з набору різних інфокомунікаційних технологій, які забезпечують його роботу. Архітектура IoT показує, як різні технології пов'язані між собою.

Найнижчий рівень архітектури Інтернету речей складається з так званих «розумних» об'єктів, які підключені до сенсорів. Ці об'єкти здатні здійснювати збір та обробку даних в реальному часі, що дає можливість виконувати певні завдання. Розвиток мікропроцесорних технологій значно зменшив розміри апаратних сенсорів, що дозволило інтегрувати їх у велику кількість різноманітних пристроїв, роблячи ці технології доступними в повсякденному житті (рис. 1.2).

Рисунок 1.2. Приклад розумних пристроїв

Зазвичай ці «розумні» об’єкти з’єднані зі шлюзами, які у свою чергу, підключаються до локальних або глобальних комп’ютерних мереж для передачі та обробки отриманої інформації. Однак існують і самодостатні пристрої, які можуть працювати автономно, підключаючись до мереж стільникових операторів або через такі з’єднання, як Wi-Fi чи Ethernet. Шлюзи виступають як концентраційні пункти, які підтримують певні стандарти або протоколи, що забезпечують зв’язок між «речами» та більш складними мережевими системами. Наступним етапом у архітектурі IoT є рівень шлюзів і мереж. На цьому рівні забезпечується з’єднання різних мереж, створюючи єдину платформу для обміну даними між різнорідними технологіями та пристроями. Для того, щоб IoT функціонував ефективно і виконував широкий спектр завдань, необхідно забезпечити спільну роботу різних мереж і протоколів. Мережі доступу повинні відповідати певним вимогам щодо швидкості передачі даних, затримок, пропускної здатності та рівня безпеки. Рівень шлюзів має ключову роль, оскільки він відповідає за з’єднання різних мереж і дозволяє здійснювати взаємодію між користувачами або автоматизованими системами та кінцевими пристроями IoT. Третій рівень архітектури IoT – це сервісний рівень, який включає набір інформаційних послуг, необхідних для автоматизації різних технологічних і бізнес-операцій у рамках IoT. Цей рівень включає в себе не тільки підтримку операційної та бізнес-діяльності, але й різноманітні методи аналітичної обробки даних, зберігання інформації, забезпечення її безпеки, а також управління різними бізнес-процесами. Сервісний рівень відповідає за функціонування централізованої панелі управління «речами», що дозволяє ефективно керувати інфраструктурою IoT, оптимізувати процеси та забезпечувати моніторинг даних у реальному часі. Нарешті, четвертий рівень архітектури IoT – це рівень додатків. На цьому етапі реалізуються специфічні додатки, орієнтовані на різні промислові сектори та галузі. Ці додатки забезпечують інтеграцію та автоматизацію процесів у таких сферах, як охорона здоров’я, виробництво, транспорт, сільське господарство та багато інших. Різноманітність додатків дозволяє максимально ефективно використовувати потенціал IoT для вирішення конкретних завдань, що стоять перед кожною галуззю і забезпечує гнучкість і масштабованість всієї системи [2].

Для забезпечення безпеки та достовірності необхідно використовувати зашифровані канали автентифікації, щоб кожен об’єкт міг підтвердити свою унікальність і справжність. Заради безпеки системи повинні приймати дані та виділяти виробничі ресурси пристроям Інтернету речей тільки після авторизації останніх, адже це захищає від зламу та так званого «спуфінгу» підміни даних [3].

Для ідентифікації об’єктів в Інтернеті речей використовуються технології, такі як RFID-мітки (рис. 1.3), штрих-коди та «розумні» датчики. Базова RFID система складається з мітки з мікрочипом і пристрою для зчитування.

Рисунок 1.3. RFID-мітка

Технологія розвивається в двох напрямках: активні RFID системи та управління додатками RFID. RFID-мітки можна інтегрувати з бездротовими сенсорними мережами для відстеження об’єктів у реальному часі.

Для обробки та накопичення даних з сенсорів повинен використовуватися вбудований комп’ютер. Для цього використовуються мікроконтролери, які створюють розподілену мережу для зберігання та обробки даних, а також надають необхідні обчислювальні ресурси для управління системами без участі людини. Структуровані дані потім передаються в «хмару» через інтерфейси (API), де вони обробляються далі, в тому числі й вручну [3].

Технологія радіочастотної ідентифікації (RFID) є ключовою для підключення об’єктів до IoT. За допомогою недорогих міток без батарей можна ідентифікувати навіть пасивні об’єкти, працюючи на основі електромагнітного поля зчитувача. Ще одним прикладом подібної технології є NFC (Near Field Communication) яка активно

використовується в платіжних картках, але також має потенціал для інтеграції різних об’єктів в Інтернет речей [4].

Для бездротової передачі даних ключову роль у створенні Інтернету речей відіграють характеристики, такі як ефективність, надійність, адаптивність і здатність до самоорганізації. Важливим стандартом у цьому контексті є IEEE 802.15.4, який регулює доступ для створення енергоефективних персональних мереж. Цей стандарт є основою для таких протоколів, як ZigBee та 6LoWPAN.

ZigBee – це технологія бездротового зв’язку, яка базується на протоколі IEEE 802.15.4 і призначена для створення низькошвидкісних приватних мереж. Вона відома своїми характеристиками, такими як низьке споживання енергії, невелика швидкість передачі даних, економічність і висока пропускна здатність.

Wi-Fi у свою чергу, є технологією бездротової локальної мережі, яка працює на частотах 2,4 ГГц або 5 ГГц і ідеально підходить для передачі великих обсягів даних між пристроями. Однак вона має високе споживання енергії та обмежену пропускну здатність, що може призвести до швидкого розряду батарей. Тому при використанні Wi-Fi для таких цілей необхідно регулярно змінювати акумулятори в пристроях [5].

Однією з важливих технологій для розвитку Інтернету речей є PLC – передача даних через електричні мережі, що корисно для пристроїв, підключених до електромережі, таких як банкомати або розумні лічильники. Технологія 6LoWPAN (рис. 1.4),яка реалізує IPv6 через IEEE 802.15.4 або PLC, є стандартом IETF і важливим інструментом для IoT.

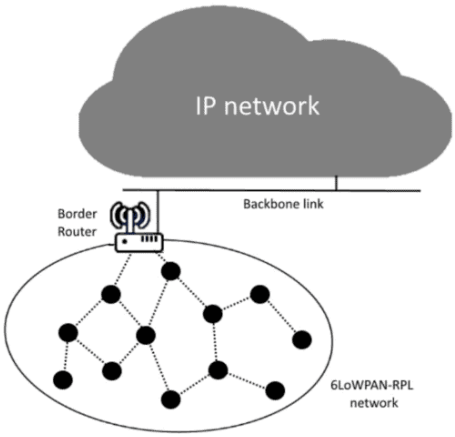

Рисунок 1.4. Технологія 6LoWPAN

1.3. Безпека Інтернету речей

У 2013 році були опубліковані результати дослідження стану безпеки в Інтернеті, яке проводив анонімний вчений. Це дослідження було здійснене в 2012 році, коли дослідник здійснював перевірку відкритих портів на усіх доступних IP-адресах. Через обсяг роботи, яку слід було виконати, він створив комп'ютерного черв'яка, який шукав пристрої без належного пароля або з простими паролями (наприклад, «root» або «admin»). В результаті його роботи був створений ботнет, що отримав назву «Carna» (рис. 1.5).

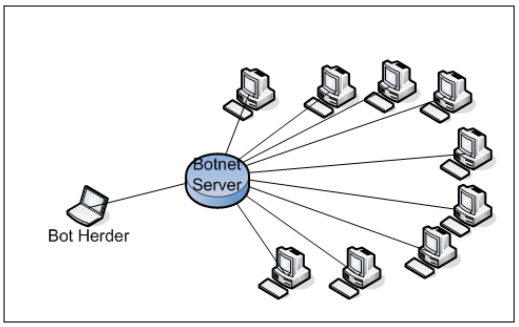

Рисунок 1.5. Структура Carna

Цей ботнет зібрав більше 9 ТБ даних, виконав 52 мільйони ICMP ping запитів, 180 мільярдів службових записів та 2,8 мільярда TCP SYN запитів на 660 мільйонів IP-адрес і здійснив опитування 71 мільярда портів. Його хробак спромігся поширитись на понад 400 тисяч пристроїв [6].

У вересні 2016 року після статті про угрупування, що продають послуги ботнетів для DDoS-атак, сайт журналіста Брайана Кребса став ціллю потужної атаки з пік трафіку 665 Гб/с, що зробило її однією з найбільших відомих атак. Атака була здійснена через ботнет з інфікованих «розумних» відеокамер. У жовтні того ж року були оприлюднені файли шкідливого ПЗ Mirai, що спричинило загрозу масового поширення таких атак. Ботнет став можливим через вразливість в «розумних» пристроях, що використовували однаковий заводський пароль. Зловмисники могли отримати доступ, перебираючи 61 варіант логін-пароля. Зломи систем IoT можуть налякати потенційних користувачів, особливо це стосується організацій, що працюють у таких сферах, як медицина, фінанси, логістика, торгівля та виробництво.

Крім того, пристрої IoT збирають конфіденційну інформацію про користувачів і її витік може мати катастрофічні наслідки. Впровадження заходів безпеки для систем IoT є складнішим, ніж для звичайних інтернет пристроїв, що підключаються до мережі. По-перше, маленькі датчики та мікропроцесори складніше захистити на апаратному рівні, по-друге, інструменти безпеки збільшать вартість і час, що витрачаються на їх виробництво, тоді як для великої кількості пристроїв важлива саме дешевизна [7].

1.4. Сучасний стан Інтернету речей

Розвиток Інтернету речей сприяє автоматизації різних сфер, включаючи будівництво, транспорт і охорону здоров'я. Компанія Juniper Research передбачає, що глобальний ринок IoT технологій, які використовують мобільні мережі, зросте з 31 мільярда доларів у 2022 році до 61 мільярда доларів до 2026 року, тобто майже подвоїться. Це зростання буде підтримуватися завдяки впровадженню технологій 5G і стільникового зв'язку з низьким енергоспоживанням для глобальних мереж (LPWA), що дозволить пристроям працювати в будь-якому місці [8].

«Розумні» міста використовують дані з датчиків для покращення інфраструктури та комунальних послуг (рис. 1.6).

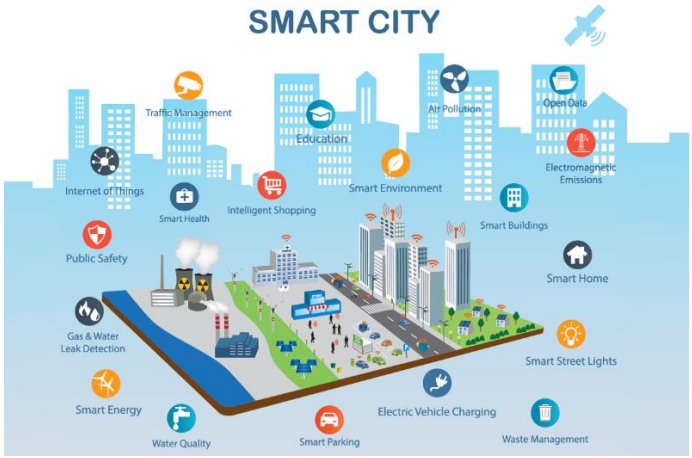

Рисунок 1.6. Складові розумного міста

Технології IoT у містах поділяються на три напрямки: транспорт, житлового комунального господарства та соціальні сервіси. До транспорту відносяться інтелектуальні системи, «розумні» паркінги і відеофіксація правил дорожнього руху;

до ЖКГ – «розумне освітлення», лічильники, управління сміттям; до соціальних сервісів підтримки екстрених служб, моніторинг безпеки, телемедицина та «розумні» школи. Інтелектуальні паркувальні системи на основі IoT зменшують час пошуку місця для паркування за допомогою датчиків, які стежать за станом майданчика та передають інформацію користувачу. Це дозволяє водіям швидко знаходити вільні місця для паркування, не витрачаючи час на безцільні пошуки. Крім того, система налаштована так, щоб відчиняти автомобільні ворота за умови, що на паркувальному місці є вільні місця [9].

У Сеулі (Південна Корея) для покращення ефективності вивезення сміття запустили програму Clean, в рамках якої сміттєві баки з датчиками рівня наповненості підключені до хмарної платформи Clean City Networks. Завдяки автоматизованим маршрутам для сміттєвозів, які формуються на основі даних про наповненість контейнерів, вдалося повністю усунути випадки переповнення баків. Завдяки цьому вдалося зменшити пробіг спецтехніки на 66% і скоротити витрати на 83% [10].

«Розумний» транспорт включає підключені до інтернету транспортні засоби, що дозволяють керування і доступ до даних (рис. 1.7). Одним із ключових напрямків є автоматизована взаємодія між автомобілями, зокрема в сфері «connected». Це стосується легкових автомобілів, громадського транспорту та інфраструктури для моніторингу і навігації.

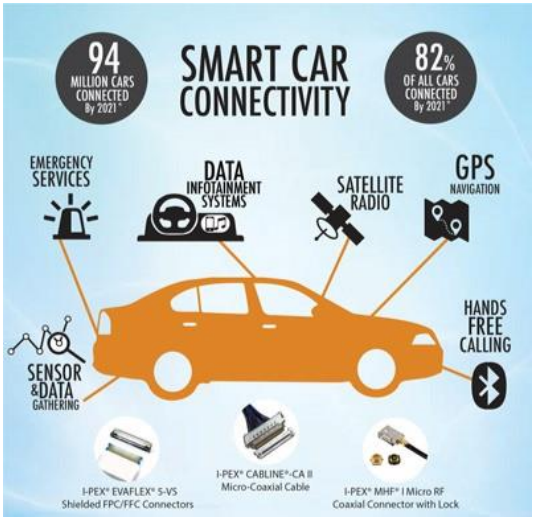

Рисунок 1.7. Складові розумного транспорту

Наприклад, автомобілі Hyundai можуть самостійно маневрувати та мають систему моніторингу водія, що попереджає про потребу в відпочинку.

РОЗДІЛ 2

ЕКОСИСТЕМА ТА БЕЗПЕКА ДАНИХ ІНТЕРНЕТУ РЕЧЕЙ

2.1. Екосистема Інтернету речей

У бізнес-контексті поняття екосистеми було введено Джеймсом після його досліджень біологічних наук про природну життєву екосистему, яка визначається як взаємодія організмів один з одним і з середовищем, де вони існують. За словами Мура, у бізнес-екосистемі можливості компанії спільно розвиваються навколо інновацій [11]. Оскільки Інтернет речей будується навколо концепції з'єднання фізичного світу з віртуальним світом Інтернету, технології, що сприяють IoT, такі як апаратні та програмні платформи, а також стандарти можуть стати ядром екосистеми [12]. Крім того, ядро екосистеми IoT може зосереджуватися на трьох ключових технічних областях: підключені пристрої або сприйняття підключення, або мережа і програми або послуги [13]. Проте пропонуються різні архітектури IoT, такі як трирівнева архітектура, п'ятирівнева архітектура, шестирівнева архітектура і навіть семирівнева архітектура. Найбільш основною серед цих архітектур є трирівнева архітектура. Подібно до багаторівневої архітектури (рис. 2.1), компонентна архітектура складається з трьох логічних рівнів: пристрої IoT, граничні пристрої та інфраструктура/хмара.

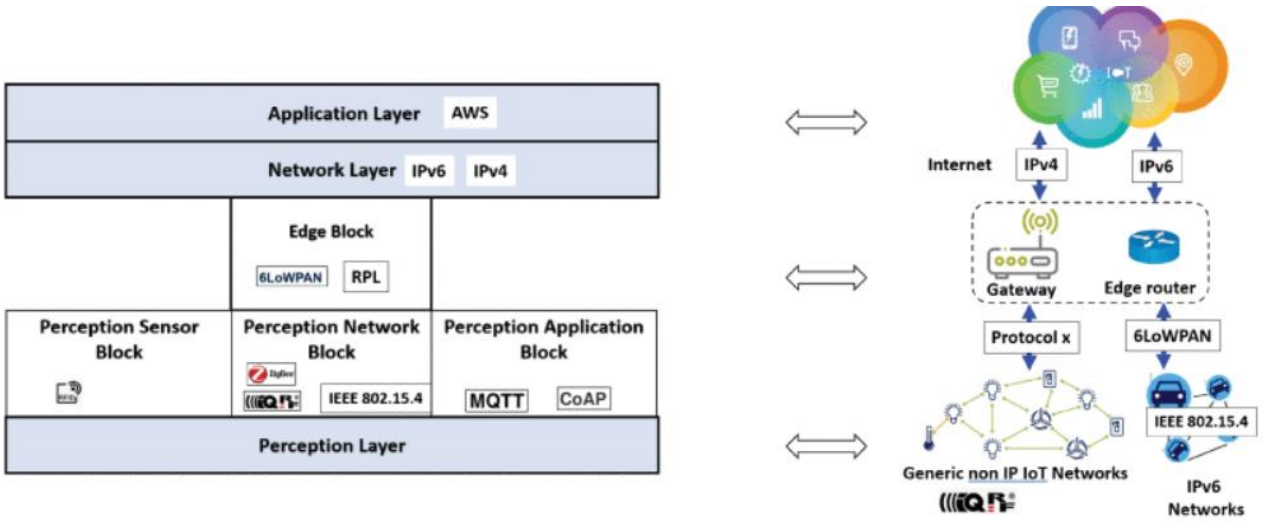

Рисунок 2.1. Приклад багаторівневої архітектури IoT

Рівень сприйняття відомий як сенсорний рівень екосистеми. Цей рівень є інформаційним джерелом додатків Інтернету речей, таких як розумні будинки, розумні електромережі та розумні міста, визначені на прикладному рівні [14].

Мережевий рівень є основним рівнем архітектури екосистеми IoT. Цей рівень також називають транспортним, оскільки маршрутизація інформації є основною функцією цього рівня.

Прикладний рівень охоплює високорівневі IoT-додатки, які забезпечують надійну та надійну взаємодію між пристроями та людиною. Прикладний рівень допомагає обробляти дані та надавати послуги, які запитують кінцеві користувачі. Після збору пристроями Інтернету речей дані аналізуються на прикладному рівні для прийняття рішень у різних сферах застосування, включаючи, але не обмежуючись, розумну охорону здоров'я, розумні міста, розумний дім, розумне сільське господарство та боротьбу зі стихійними лихами. Програми в IoT можна розділити на п'ять категорій, таких як персональні та соціальні програми, домашні програми, транспорт і логістика, програми охорони здоров'я та розумне середовище [15].

2.2. Безпека даних Інтернету речей

Екосистема IoT являє собою дуже гнучкий спосіб організації розумних додатків і побудови орієнтованої на споживача інфраструктури. Однак існує низка проблем, які впливають на безпеку та конфіденційність залучених сторін, коли мова йде про малопотужні пристрої IoT. Часто існує компроміс між впровадженням заходів безпеки та підтримкою операцій у межах заданих допусків (рис. 2.2).

Security Mechanisms			AES-CCM-128	AES-CCM-64	AES-CCM-32	AES-CBC-128	AES-CBC-64	AES-CBC-32	AES-CTR	AES-ECB-128	DTLS	IPSec	TLS	SSL	UID & PWD
IoT Perception Layer Enabling Technologies	Perception Sensor Block	RFID													
	Perception Application Block	CoAP									*				
		MQTT											* R		*
	Perception Network Block	IEEE 802.15.4	*	*	*	*	*	*	*						
		ZigBee	* ANL												
		IQRF				*N				*AU					
	Edge Block	6LoWPAN													
		RPL	*												
IoT Network Layer Enabling Technologies		IPv6										*			
		IPv4										* R			
IoT Application Layer Enabling Technologies		AWS										*	*	*	*

Рисунок 2.2. Механізм безпеки IoT

1. Perception Sensor Block.

Система RFID складається з двох частин. Електронна мітка RFID, що також називається транспондером, яка містить мікрочіп для зберігання інформації та антену (RFID-мітка може бути прикріплена до об'єкта з метою ідентифікації, відстеження або моніторингу). Зчитувач RFID, який також називають запитувачем. Зчитувач передає, зчитує або записує (опитує) тег через радіохвилі [16]. Пасивні мітки зчитуються на відстані кількох сантиметрів, тоді як активні мітки можна зчитувати з більшої відстані залежно від програми. Коли зчитувач RFID оснащена відповідним протоколом зв'язку, що дозволяє підключати його до Інтернету, розподілені зчитувачі RFID можуть ідентифікувати, відстежувати та контролювати помічені об'єкти в усьому світі [17].

2. Perception Application Block Security.

CoAP – це спеціалізований веб-протокол передачі на основі UDP для використання з обмеженими вузлами та обмеженими мережами, наприклад мережами з низьким енергоспоживанням і мережами з втратами (рис. 2.3.). Крім того, цей протокол призначений для взаємодії з протоколом передачі гіпертексту (HTTP), що дозволяє додаткам зі спеціальними вимогами, такими як дуже низькі накладні витрати, підтримка багатоадресної передачі та простота середовища обмежень для взаємодії з Інтернетом. Однак CoAP не використовує сліпо HTTP, натомість він

розроблений як підмножина архітектури RESTful, яка оптимізована для спілкування M2M [18].

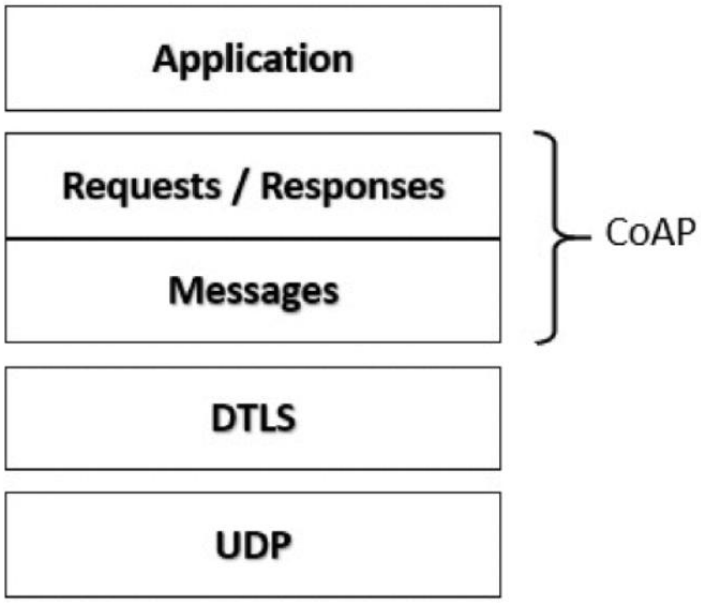

Рисунок 2.3. Абстрактне шарування DTLS-Secured CoAP

MQTT – це транспортний протокол прикладного рівня, який підходить для спілкування M2M та IoT. Цей відкритий протокол публікації/підписки клієнт-сервера, стандартизований OASIS, розроблений як простий і легкий протокол обміну повідомленнями, який підходить для використання пристроями з обмеженими можливостями в мережах з низькою пропускною здатністю та ненадійними мережами. Протокол MQTT був широко впроваджений у різноманітних галузях з моменту його розробки. Архітектура MQTT складається з видавця, передплатника та брокера [19]. Цей протокол рекомендує деякі рішення безпеки, які можна реалізувати за допомогою MQTT, такі як SSL/TLS. Він також рекомендує для реалізацій, які використовують SSL/TLS як параметр безпеки, використовувати порт користувача TCP 8883 як захищений порт (secure-mqtt), призначений IANA навпроти порту 1883, який не використовує протокол TLS [20].

3. Perception Network Block Security.

Стандарт IEEE 802.15.4 визначає рівень MAC і PHY багатьох протоколів мережевих специфікацій, включаючи але не обмежуючись ними 6LoWPAN, Zigbee, WirelessHART, WiSUN, MiWi [21]. Цей стандарт розроблено, щоб забезпечити

структуру та нижні рівні моделі OSI, фізичний (PHY) рівень і підрівень керування доступом до середовища (MAC) для недорогих, низькошвидкісних і малопотужних бездротових мереж

ZigBee – це відкритий стандарт, що належить ZigBee Alliance і побудований на основі рівня PHY і підрівня MAC стандарту IEEE 802.15.4. Стандарт ZigBee описує специфікації для вищих рівнів протоколу, рівня мережі ZigBee (ZNTW) і рівня застосування ZigBee (ZAPL). Рівень ZNTW надає функціональні можливості для забезпечення правильної роботи підрівня MAC IEEE 802.15.4 і для забезпечення відповідного інтерфейсу обслуговування до прикладного рівня. Рівень ZAPL складається з трьох основних компонентів. Підрівень підтримки додатків (APS), який діє як інтерфейс між рівнем ZAPL і рівнем ZNTW.

Ключ рівня ZNTW – це тимчасовий унікальний 128-бітний ключ AES, який генерується центром довіри щоразу, коли він застарів і надається всім пристроям у мережі за допомогою старого ключа [22].

4. Edge Block Security.

IP версії 6 (IPv6) дозволяє безпосередньо звертатися до кожного периферійного пристрою мережі. Отже, це вирівнює ієрархію адресації, усуває потребу в складних шлюзах і спрощує модель підключення. Крім того, малопотужна бездротова персональна мережа (LoWPAN) є простою, недорогою бездротовою мережею зв’язку в програмах з обмеженою потужністю та невибагливими вимогами до пропускної здатності. 6LoWPAN – це визначення протоколу, що об’єднує дві концепції, IPv6 і LoWPAN, щоб дозволити мережам, включаючи обмежені пристрої з обмеженими можливостями обчислення, зберігання та зв’язку, мати справу з вимогами IPv6.

РОЗДІЛ 3
ОГЛЯД ІНСТРУМЕНТІВ CISCO PACKET TRACER ДЛЯ ПРОЄКТУВАННЯ РІШЕНЬ ІНТЕРНЕТУ РЕЧЕЙ

3.1. Огляд середовища Cisco Packet Tracer

Cisco Packet Tracer – це потужний інструмент симуляції віртуальної мережі, який використовується для навчання та розуміння різних концепцій в комп'ютерних мережах. Система надає користувачам можливість проектувати та моделювати мережу за допомогою віртуальної мережі такі пристрої, як концентратор, маршрутизатор, комутатори тощо. У Cisco Packet Tracer симуляція працює без наявної фізичної мережі.

Cisco Packet Tracer має два робочих простори: один є фізичним, а інший логічним. Логічне дозволяє користувачеві розміщувати та підключати віртуальні мережеві пристрої, тоді як фізичний перегляд надає графічне представлення пристроїв віртуальної мережі. У фізичному вигляді пристроїв користувачі можуть додати додаткові модулі до доступного слота в пристроїв (рис. 3.1).

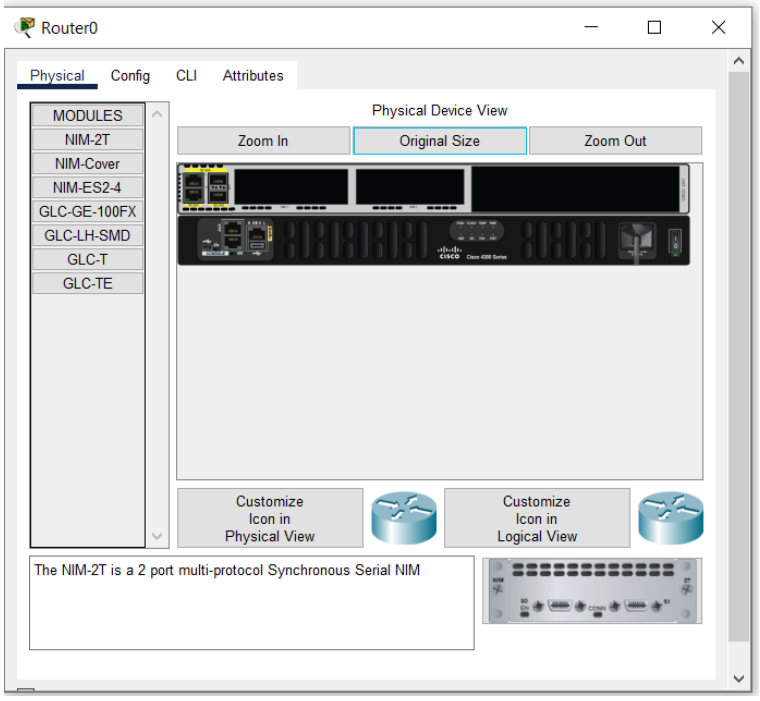

Рисунок 3.1. Фізичний вигляд маршрутизатора

Інструмент також забезпечує два режими: режим реального часу та режим симуляції. У реальному часі користувачі можуть мати чітке уявлення про те, як поводяться пристрої. У цьому режимі пристрої ведуть себе як справжні пристрої. З іншого боку, режим моделювання допомагає користувачам зрозуміти основну концепції мережевих операцій.

Робоча область Cisco Packet Tracer складається з кількох основних частин:

1. Головне меню та панелі інструментів – містить інструменти для створення, збереження, відкриття проєктів, а також для налаштувань середовища.
2. Область моделювання (Workspace) – центральна частина інтерфейсу, де розміщуються мережеві пристрої, кабелі та з'єднання.
3. Панель пристроїв (Device-Type Selection) та панель компонентів (Device-Specific Selection) – дозволяють вибирати мережеві пристрої, такі як комутатори, маршрутизатори, сервери, датчики IoT тощо.
4. Консоль налаштування (Config/CLI Window) – використовується для конфігурації пристроїв за допомогою графічного інтерфейсу або командного рядка (CLI).
5. Симуляційний режим (Simulation Panel) – дозволяє переглядати передачу пакетів у мережі, діагностувати проблеми та аналізувати їх.

*Головне меню та панелі інструментів (рис. 3.2)***.**

Меню Cisco Packet Tracer розташоване у верхній частині вікна програми і містить кілька основних пунктів.

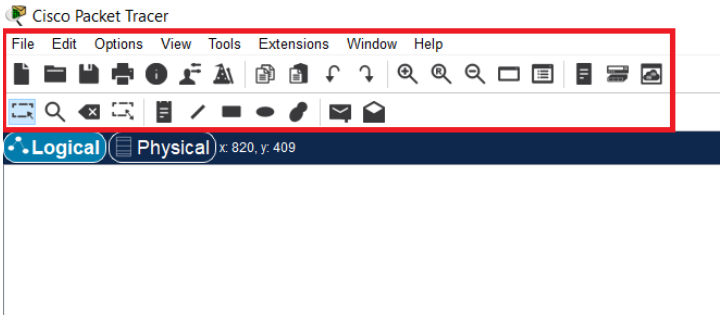

Рисунок 3.2. Головне меню та панелі інструментів

- Меню *File* відповідає за управління файлами проєкту.
- Меню *Edit* містить базові команди для редагування об'єктів.

- Меню *Options* дозволяє змінювати налаштування програми.
- Меню View керує відображенням інтерфейсу та панелей програми.
- Меню Tools містить додаткові можливості для роботи з мережею.
- Меню Extensions дозволяє підключати додаткові функції та модулі.
- Меню Help (Допомога) надає доступ до документації та довідкової інформації.

Область моделювання Workspace (рис. 3.3).

Область моделювання Workspace розташована в центральній частині інтерфейсу програми і є основним місцем для роботи з мережею. У цій області користувач може додавати пристрої, з'єднувати їх за допомогою кабелів та налаштовувати різні параметри мережі, створюючи та тестуючи мережеві топології. Область підтримує два режими роботи: Real-time (Реальний час), де мережа функціонує у звичайному режимі з реальним часом для налаштування і перевірки мережевих з'єднань і Simulation (Симуляція), що дозволяє відслідковувати передачу пакетів даних між пристроями в уповільненому режимі для детального аналізу процесу комунікації. Наприклад, додавши маршрутизатор, комутатор і комп'ютер, а також підключивши їх Ethernet-кабелями є можливість активувати симуляційний режим, щоб наочно побачити, як дані передаються між пристроями, що є корисним для вивчення та тестування мережевих налаштувань.

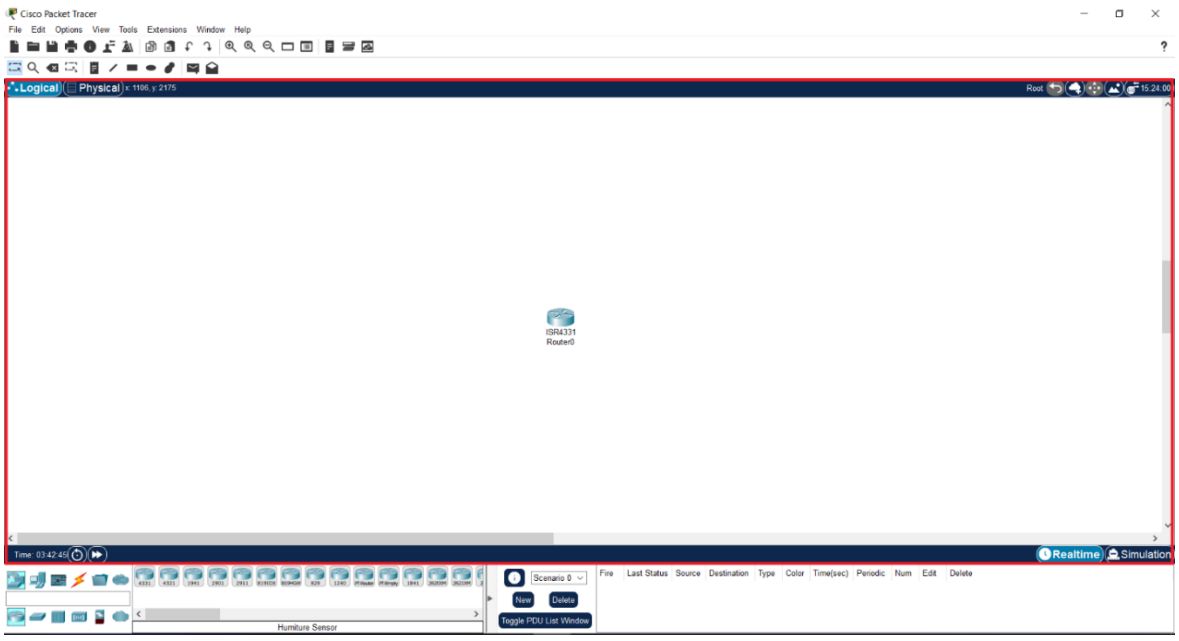

Рисунок 3.3. Робочий простір Cisco Packet Tracer

Панель пристроїв та панель компонентів.

Панель пристроїв (рис. 3.4.) розташована в нижній частині інтерфейсу та включає кілька категорій для вибору пристроїв, які можна додавати на робочу область:

1. Network Devices містить маршрутизатори, комутатори, точки доступу та інші мережеві елементи.
2. End Devices включає комп'ютери, сервери, смартфони та інші пристрої, які можуть бути кінцевими точками в мережі.
3. IoT Devices надає доступ до різних пристроїв IoT, таких як датчики, камери, лампи та мікроконтролери.
4. Connections містить різноманітні кабелі, які використовуються для з'єднання пристроїв, включаючи Ethernet, серійні та оптоволоконні кабелі.

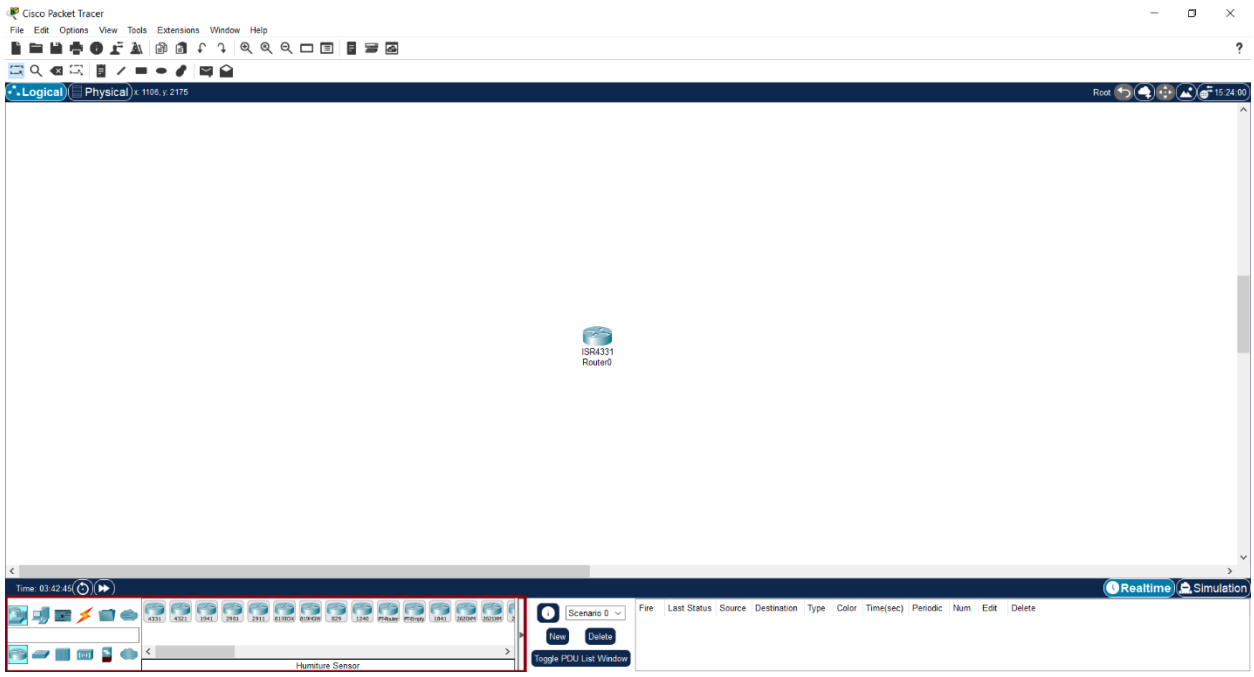

Рисунок 3.4. Панель пристроїв

Категорія Network Devices.

Категорія мережевих пристроїв містить основні компоненти, необхідні для побудови мережевої інфраструктури, включаючи маршрутизатори, комутатори, точки доступу та інші пристрої.

Маршрутизатори використовуються для з’єднання різних мереж та управління передачею даних між ними (рис. 3.5). Вони підтримують кілька інтерфейсів для підключення до інших пристроїв і здійснюють маршрутизацію пакетів за допомогою таблиць маршрутизації. Дозволяють передавати дані між різними мережами та сегментами.

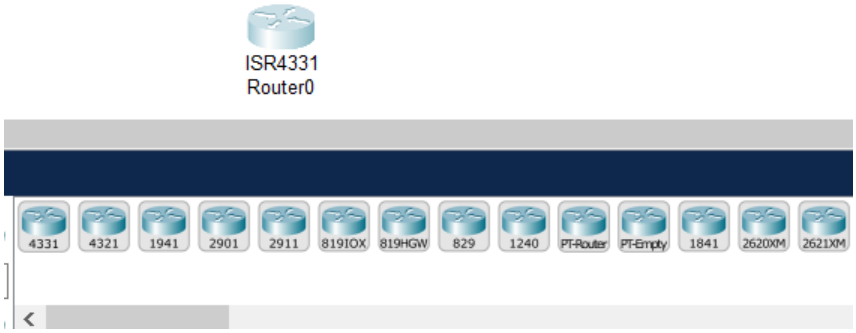

Рисунок 3.5. Вкладка маршрутизатори

Припустимо, користувач створює мережу, що містить два маршрутизатори, підключені між собою через два Ethernet-кабелі. Щоб маршрутизатори могли передавати дані між двома мережевими сегментами, потрібно налаштувати їхні інтерфейси, додати IP-адреси та створити відповідні записи в таблицях маршрутизації. Для цього у CLI кожного маршрутизатора можна виконати такі команди (рис. 3.6):

```
Router(config)# interface GigabitEthernet0/0
Router(config-if)# ip address 192.168.1.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# ip route 192.168.2.0 255.255.255.0 192.168.1.2
```

Рисунок 3.6. Команди у CLI

Комутатори (Switches) є ключовими пристроями локальних мереж, що забезпечують швидке та ефективне з’єднання між пристроями в межах однієї мережі (рис. 3.7). Вони працюють на другому рівні моделі OSI (канальному рівні) та передають дані, використовуючи MAC-адреси пристроїв.

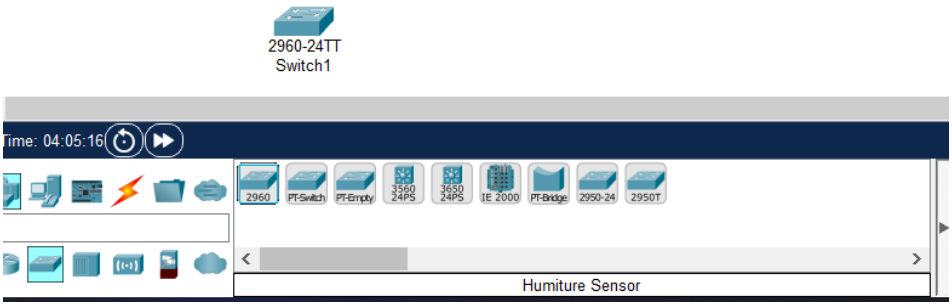

Рисунок 3.7. Комутатори

Припустимо є мережа з кількома комп'ютерами, які потрібно підключити для забезпечення швидкої передачі даних. Користувач використовує керований комутатор Cisco 2960 та налаштовуєте VLAN, щоб розділити мережу на логічні сегменти (рис. 3.8).

```
Switch(config)# vlan 10
Switch(config-vlan)# name Sales
Switch(config-vlan)# exit
Switch(config)# interface FastEthernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# exit
```

Рисунок 3.8. Налаштування Switch

Безпровідні точки доступу (WAP) є ключовими пристроями для організації Wi-Fi-мереж (рис. 3.9). Вони забезпечують бездротове з'єднання між кінцевими пристроями (смартфонами, ноутбуками, планшетами) та дротовою інфраструктурою мережі. Дозволяють підключення бездротових пристроїв до мережі через Wi-Fi.

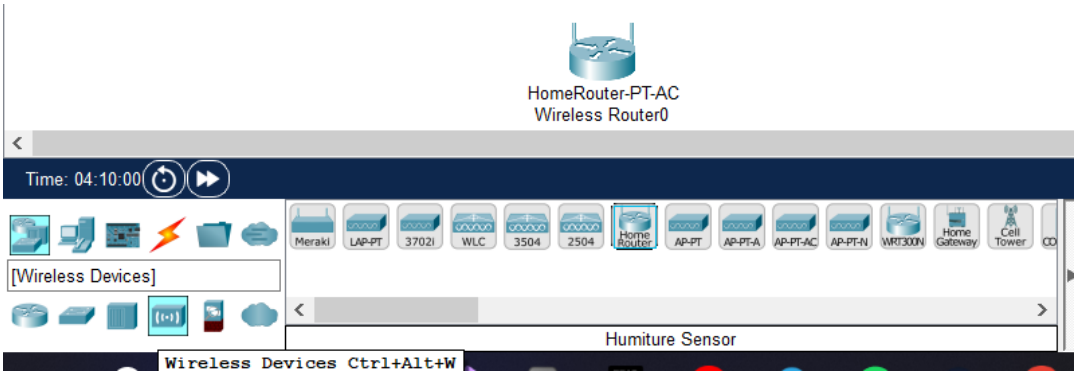

Рисунок 3.9. Розділ WAP

Припустимо, користувач хоче забезпечити бездротовий доступ в офісі. Він додає WAP Cisco Aironet до мережі, підключає її до комутатора через Ethernet-кабель і налаштовує основні параметри (рис. 3.10).

```
AP(config)# interface Dot11Radio0
AP(config-if)# ssid OfficeWiFi
AP(config-if-ssid)# authentication open
AP(config-if-ssid)# authentication key-management wpa
AP(config-if-ssid)# wpa-psk ascii StrongPass123
AP(config-if-ssid)# exit
AP(config-if)# no shutdown
AP(config)# interface GigabitEthernet0
AP(config-if)# no shutdown
AP(config-if)# exit
```

Рисунок 3.10. Налаштування WAP

Ця конфігурація створює Wi-Fi-мережу OfficeWiFi із захистом WPA-PSK, що дозволяє співробітникам безпечно підключатися до мережі.

Категорія Connections.

У Cisco Packet Tracer, категорія Connections містить інструменти для підключення різних пристроїв у мережі (рис. 3.11). Вона дозволяє налаштовувати фізичні з'єднання між пристроями, такими як маршрутизатори, комутатори, комп'ютери, сервери та інші мережеві компоненти.

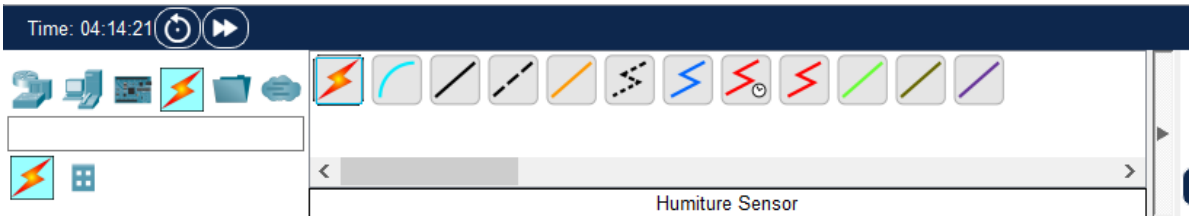

Рисунок 3.11. Категорія Connections

Основні елементи, доступні в категорії Connections, включають:

- Straight-Through Cable: використовується для з'єднання пристроїв, таких як комп'ютери та комутатори, або маршрутизатори та комутатори.
- Crossover Cable: використовується для з'єднання схожих пристроїв,
- Console Cable: кабель використовується для підключення комп'ютера або терміналу до консольного порту маршрутизатора чи комутатора, щоб здійснювати конфігурацію пристроїв через командний рядок.

- Fiber Optic Cable: використовується для з’єднання пристроїв на великих відстанях.
- Serial Cable: використовується для підключення маршрутизаторів або для з’єднання між пристроями в рамках WAN-мережі.

Консоль налаштування пристроїв (Config).

Панель налаштувань забезпечує два основні способи взаємодії з мережевим обладнанням: через Config (рис. 3.12), CLI (рис. 3.13).

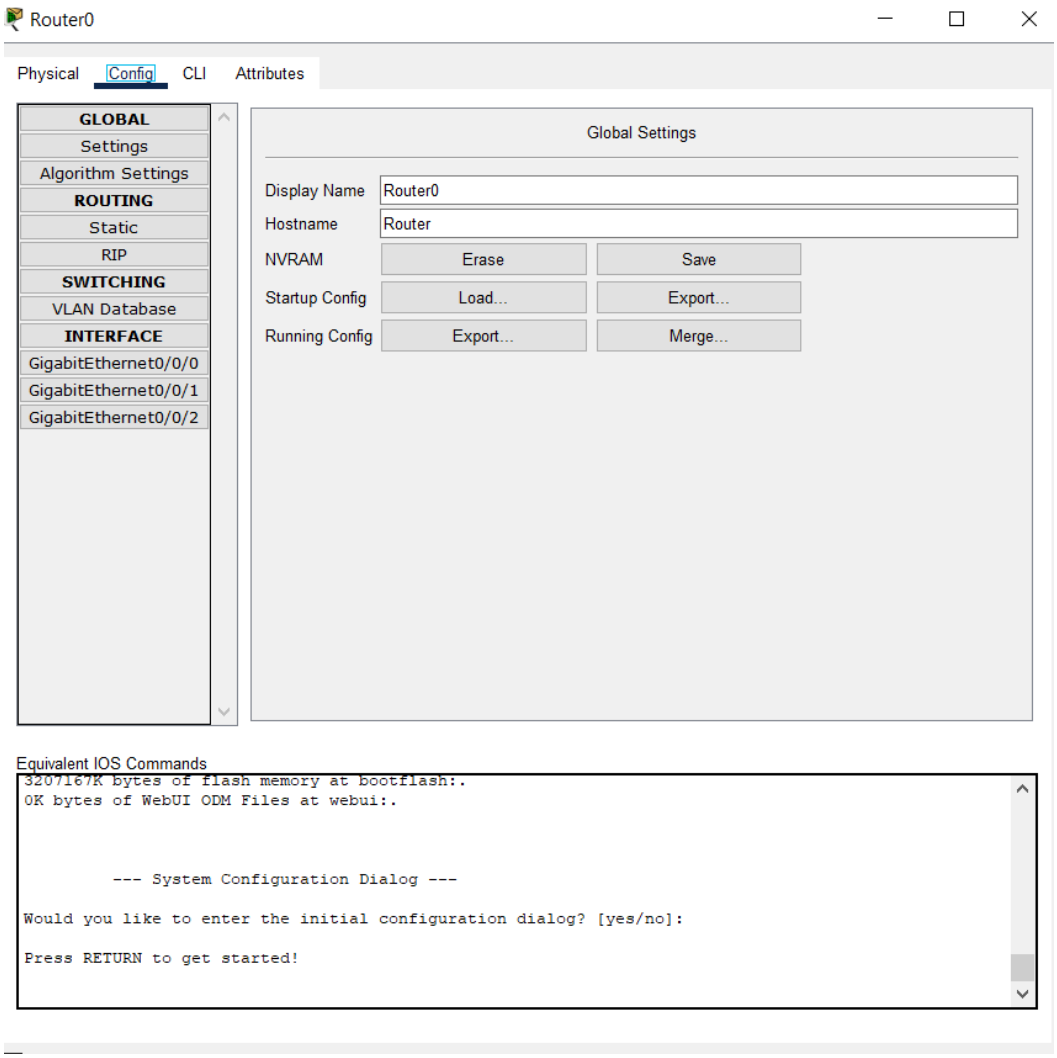

Рисунок 3.12. Вкладка конфігурації Cisco Packet Tracer

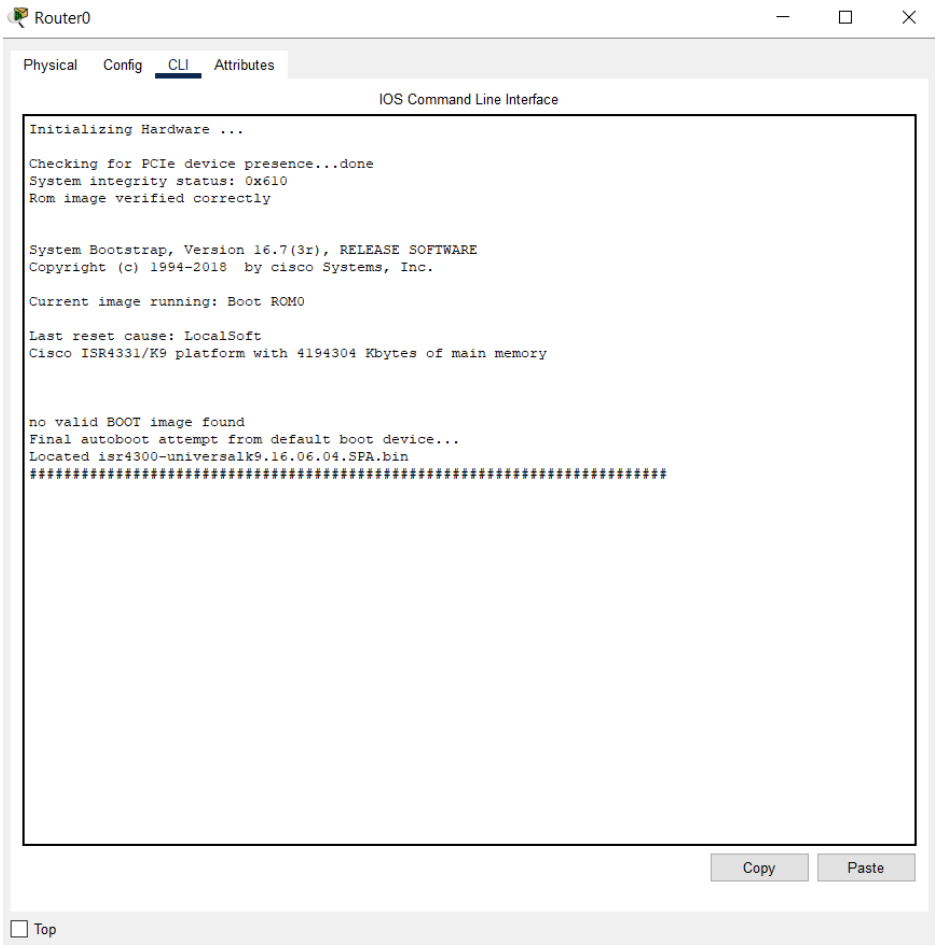

Рисунок 3.13. Вкладка інтерфейсу командного рядка Cisco Packet Tracer

CLI дає більше контролю над конфігурацією пристроїв за допомогою введення текстових команд. Користувач можете вводити стандартні команди Cisco, такі як enable, configure terminal, interface, ip address, ip route для налаштування мережевих інтерфейсів, маршрутів, безпеки та інших параметрів. Це дозволяє більш детально налаштувати мережу, оскільки кожна команда точно вказує, які дії виконуються.

Наприклад, якщо вибрати маршрутизатор, то можна відкрити CLI та ввести команди для налаштування IP-адреси на інтерфейсі (рис. 3.14)

```
Router> enable
Router# configure terminal
Router(config)# interface gigabitEthernet 0/0
Router(config-if)# ip address 192.168.1.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# exit
Router# write memory
```

Рисунок 3.14. Команди для налаштування IP-адреси в CLI

Це призначить IP-адресу інтерфейсу та активує його, даючи змогу маршрутизатору функціонувати в мережі. Після цього команда write memory зберігає конфігурацію. Інший приклад конфігурації може бути налаштування маршруту (рис. 3.15).

```
Router# configure terminal
Router(config)# ip route 0.0.0.0 0.0.0.0 192.168.1.254
Router(config)# exit
Router# write memory
```

Рисунок 3.15. Налаштування маршруту в CLI

Симуляційний режим (Simulation Panel).

Панель симуляції розташована в нижній частині праворуч і є важливим інструментом для аналізу процесу передачі даних у мережі (рис. 3.16).

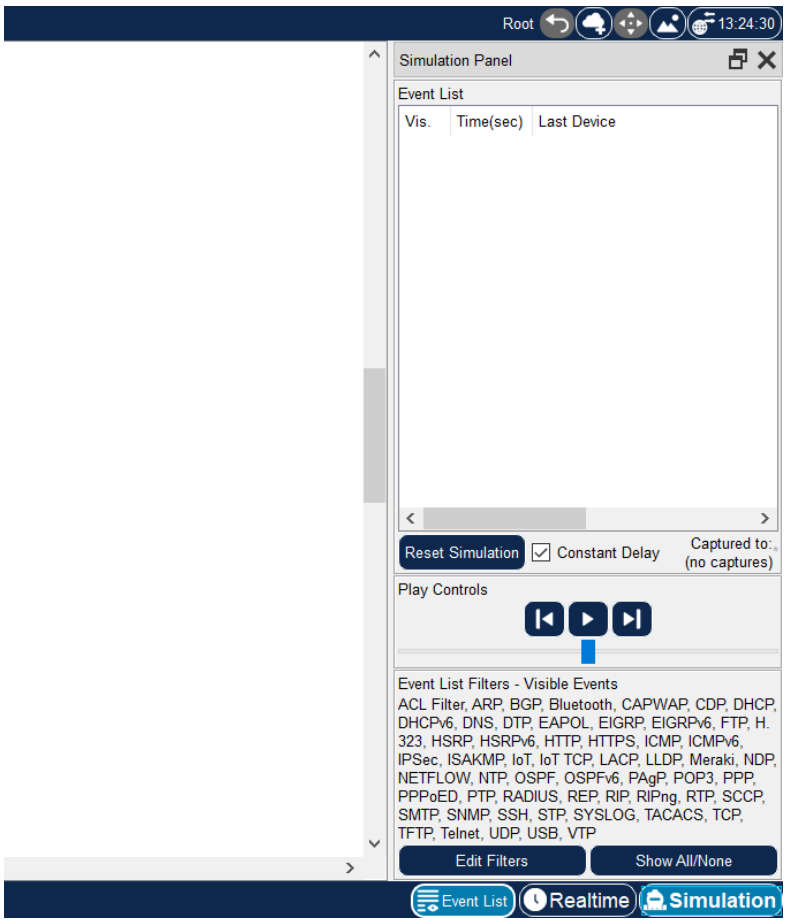

Рисунок 3.16. Simulation Panel

Панель симуляції дозволяє спостерігати, як пакети передаються між пристроями, даючи можливість детально досліджувати мережеву активність. Панель відображає різні рівні моделі OSI, що дозволяє користувачам аналізувати, як кожен рівень моделі взаємодіє з іншими під час передачі даних.

3.2. Реалізація IoT-рішень у Cisco Packet Tracer

У Cisco Packet Tracer доступні функції, які дозволяють проектувати та тестувати рішення Інтернету речей:

1. Віртуальні пристрої: включають датчики (температури, вологості, руху), розумні лампи, двері, розетки, камери тощо.
2. Програмування пристроїв: використання Python для автоматизації роботи пристроїв та створення інтерактивних сценаріїв.
3. Підключення IoT до мережі: моделювання підключення пристроїв до Wi-Fi-роутерів, шлюзів і серверів.
4. Підтримка протоколів**:** включає MQTT, CoAP, HTTP та інші протоколи, які широко використовуються в IoT для передачі даних.
5. Об’єднання фізичних та віртуальних середовищ: симуляція взаємодії пристроїв із традиційними мережевими елементами, такими як маршрутизатори, комутатори та сервери.

Остання версія 8.2 програми містить деякі нові функції, які допомагають з симуляцією Інтернету речей. Цими новими функціями є розумні пристрої, датчики, приводи та мікроконтролер. Деякі з цих інтелектуальних пристроїв, включених у систему – це розумні вікна, розумний вентилятор, розумний світло, сигнальна сирена. Також є наявні деякі датчики, такі як рівень води, температура, вологість, вуглекислий газ. Найважливіша річ у новій версії полягає в тому, що всі пристрої можна програмувати з використанням різних мов програмування, таких як phyton, javascript і blocky.

Категорія End Devices та пристроїв Інтернету речей.

Категорія End Devices у Cisco Packet Tracer містить пристрої, що імітують реальні кінцеві пристрої, використовувані для підключення до мережі, обробки даних і керування (рис. 3.17.). До них належать:

- PC (Персональний комп'ютер) – стаціонарний пристрій із підтримкою Ethernet, Wi-Fi та серійного підключення.
- Laptop – мобільний аналог PC із підтримкою Wi-Fi.
- Server (Сервер) – пристрій для надання мережевих сервісів.
- Printer (Принтер) – мережевий принтер.
- IP Phone (IP-телефон) – пристрій для VoIP-зв'язку з підтримкою Ethernet і PoE.
- Tablet (Планшет) – мобільний пристрій із підключенням через Wi-Fi, що імітує використання бездротових технологій у мережі.
- TV (Телевізор) – смарт-пристрій із підтримкою Ethernet.
- Smartphone (Смартфон) – мобільний пристрій із підключенням через Wi-Fi.
- Home Gateway (Домашній шлюз) – маршрутизатор для підключення пристроїв до Інтернету з підтримкою Ethernet, Wi-Fi, NAT і DHCP.

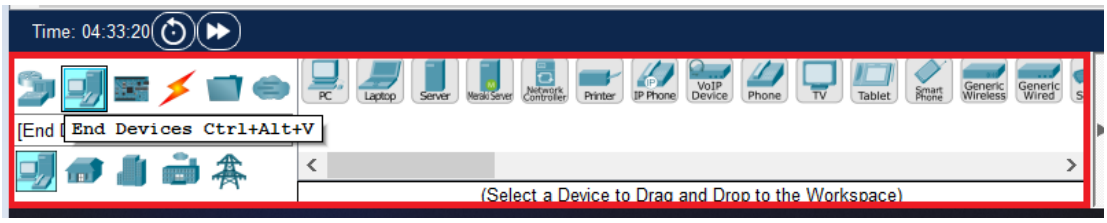

Рисунок 3.17. Категорія End Devices

Вкладка Home містить пристрої для моделювання домашніх мереж (рис. 3.18):

- Smart Devices – включають розумні лампи, дверні замки, термостати та інші IoT-пристрої, що підключаються через Wi-Fi або ZigBee.
- Home Router (Домашній маршрутизатор) – забезпечує бездротовий доступ до Інтернету для кінцевих пристроїв, підтримує DHCP, NAT та брандмауер.
- Smart Speaker (Розумна колонка**)** – підключений пристрій для голосових команд та управління іншими IoT-пристроями в мережі.

- Surveillance Camera (Камера спостереження) – підключена до мережі IP-камера для моніторингу території.
- Smart Plug (Розумна розетка) – електронний пристрій для дистанційного вмикання та вимикання електроприладів через Wi-Fi.

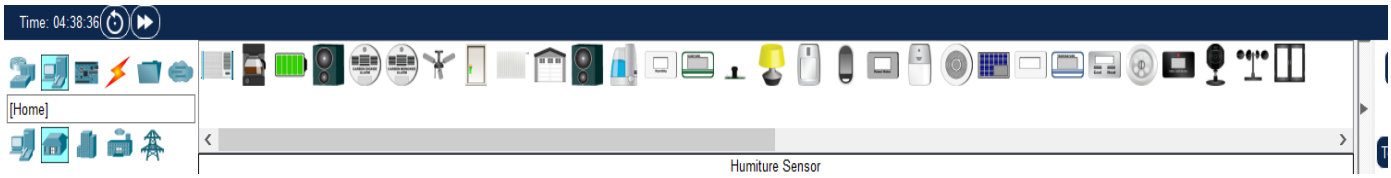

Рисунок 3.18. Вкладка Home

Ці пристрої дозволяють змоделювати роботу сучасних розумних будинків і протестувати їхню взаємодію з мережею.

Вкладка Smart City містить пристрої для моделювання інфраструктури розумного міста (рис. 3.19):

- Traffic Light (Світлофор) – керований пристрій для моделювання регулювання руху на перехрестях.
- Parking Sensor (Датчик парковки) – сенсор для визначення наявності вільних місць на паркінгу.
- Smart Street Light (Розумний ліхтар) – пристрій для моделювання автоматичного освітлення вулиць.
- Environmental Sensor (Екологічний датчик) – використовується для збору інформації про рівень забруднення, температуру та вологість.
- Microcontroller Board (Мікроконтролер) – використовується для програмування та управління IoT-пристроями.
- RFID Sensor (RFID-зчитувач) – використовується для сканування RFID-карток.

Рисунок 3.19. Вкладка Smart City

Вкладка Industrial містить пристрої для моделювання промислових мереж (рис. 3.20):

- PLC Controller (Програмований логічний контролер) – керує автоматизованими системами виробництва.
- Industrial Robot (Промисловий робот) – симулює автоматизовані виробничі процеси.
- SCADA System – використовується для моніторингу та управління промисловими процесами.
- Розумні кондиціонери – це пристрої для охолодження повітря, які можуть автоматично регулювати температуру, вологість і навіть якість повітря в кімнаті.
- Нагрівачі – використовуються для обігріву приміщень.
- Зволожувачі – підтримують оптимальний рівень вологості в приміщеннях, що особливо важливо в зимовий період, коли через опалення повітря стає сухим.
- Генератори сигналів – використовуються в різноманітних IoT-пристроях для створення або обробки електронних сигналів.
- Карбонові детектори – виявляють підвищену концентрацію вуглекислого газу (CO_2) у приміщенні.
- Сенсори температури – вимірюють температуру в приміщенні або зовні.
- Сенсори вологості – вимірюють рівень вологості в повітрі.
- Сенсори руху – використовуються для виявлення фізичної активності в кімнатах або зовні будівлі.

- Детектори диму – це пристрої, які виявляють дим, що може свідчити про початок пожежі.
- Контролери вогню – інтегрують інформацію від детекторів диму, температури та інших сенсорів для автоматичного реагування на пожежу.
- Сонячні панелі – перетворюють сонячну енергію на електричну.
- Вітрогенератори – перетворюють кінетичну енергію вітру на електричну.

Рисунок 3.20. Вкладка Industrial

Вкладка Power Grid містить пристрої для моделювання енергетичних мереж (рис. 3.21):

- Smart Meter (Розумний лічильник) – пристрій для віддаленого обліку споживання електроенергії.
- Power Generator (Генератор електроенергії) – використовується для моделювання джерел електроенергії.
- Power Transformer (Трансформатор) – пристрій для моделювання зміни напруги в електромережі.

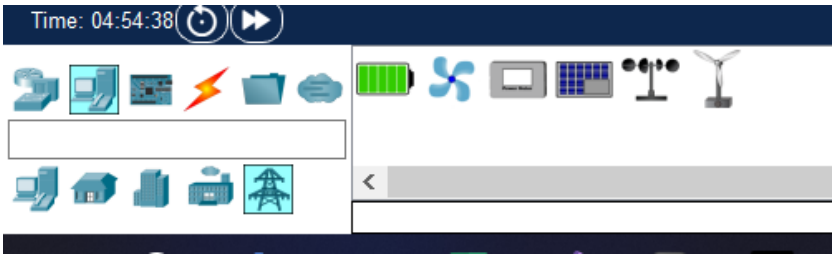

Рисунок 3.21. Вкладка Industrial

У Cisco Packet Tracer, категорія Components також містить підкатегорії, які спеціалізуються на розробці та моделюванні IoT (рис. 3.22). Ці підкатегорії дозволяють створювати сценарії, де мережеві пристрої взаємодіють з фізичними компонентами, такими як датчики та актуатори.

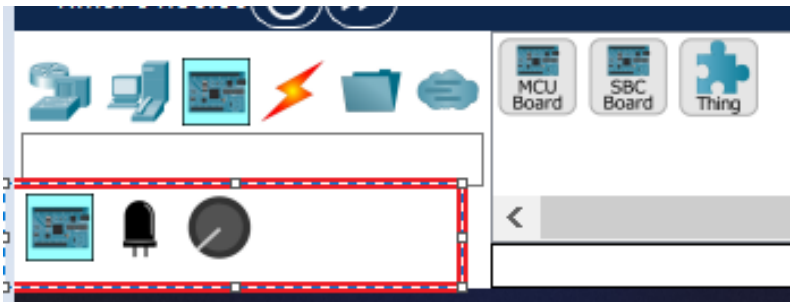

Рисунок 3.22. Категорія Components

Підкатегорія Board містить мікроконтролери та плати для розробки рішень IoT (рис. 3.23). Вони служать для підключення різних датчиків та актуаторів, щоб обробляти та передавати дані в мережу.

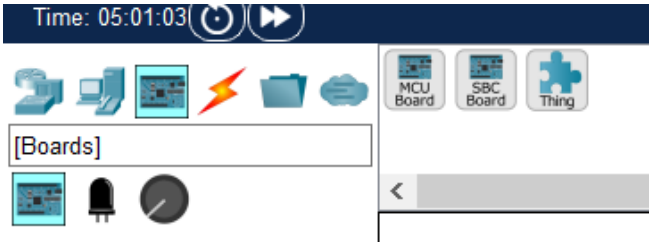

Рисунок 3.23. Підкатегорія Board

Актуатори – це пристрої, що виконують фізичні дії в реальному світі на основі отриманих сигналів або даних (рис. 3.24).

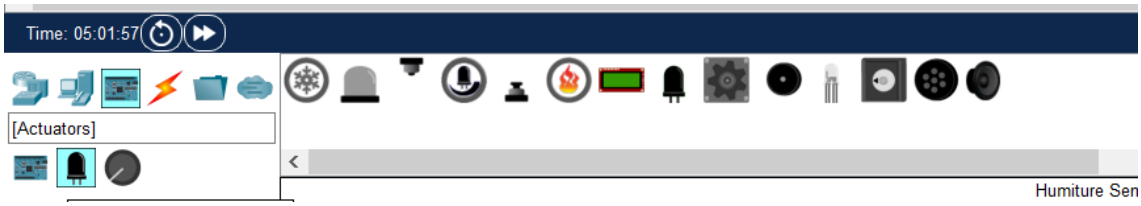

Рисунок 3.24. Категорія Актуатори

Вони зазвичай взаємодіють з контролерами та датчиками в IoT-системах. У Packet Tracer можуть бути представлені такі актуатори:

- Servo motor (Серво-мотор): використовується для точного керування рухами або положенням об’єктів.
- DC motor (Двигун постійного струму): використовується для руху об’єктів, наприклад, у роботах або механізмах.

- Lamp (Лампа): симулює фізичне включення/вимкнення світла на основі зовнішніх даних.
- Buzzer (Зуммер): використовується для сигналізації або сповіщення через звуковий сигнал.

Датчики збирають дані з фізичного світу та передають їх до контролерів або комп'ютерних систем для обробки. Вони є невід'ємною частиною систем IoT. У Packet Tracer доступні такі типи датчиків (рис. 3.25):

- Temperature sensor (Датчик температури): вимірює температуру навколишнього середовища.
- Light sensor (Датчик світла): реагує на рівень освітленості, може використовуватися для автоматичного регулювання освітлення.
- Moisture sensor (Датчик вологості): вимірює рівень вологи в ґрунті, що може бути корисно в аграрних або гідропонних системах.
- Motion sensor (Датчик руху): виявляє рух об'єктів у певній зоні.
- Gas sensor (Датчик газу): використовується для виявлення певних газів у навколишньому середовищі.

Рисунок 3.25. Категорія Sensor

Датчики взаємодіють з контролерами, щоб забезпечити відповідні реакції в мережі або активувати актуатори.

3.3. Програмування IoT-пристроїв засобами Cisco Packet Tracer

Cisco Packet Tracer дозволяє програмувати поведінку IoT-пристроїв через вкладку «Programming» (рис. 3.26). Основні підходи:

- Використання мов програмування – підтримуються JavaScript та Python, що дозволяє створювати складні логічні алгоритми керування IoT-пристроями.

- Обробка подій – пристрої можуть реагувати на зміну параметрів середовища, наприклад, включати освітлення при виявленні руху або регулювати температуру в приміщенні.
- Автоматизація процесів – можливість використання умовних операторів (if-else), циклів та логічних функцій для створення автономних систем.
- Зберігання та обробка даних – отримані з датчиків дані можуть зберігатися та аналізуватися для прийняття рішень.
- Взаємодія з хмарними сервісами – підтримується підключення до IoT Cloud Server, що дозволяє керувати пристроями дистанційно та отримувати дані через інтернет.

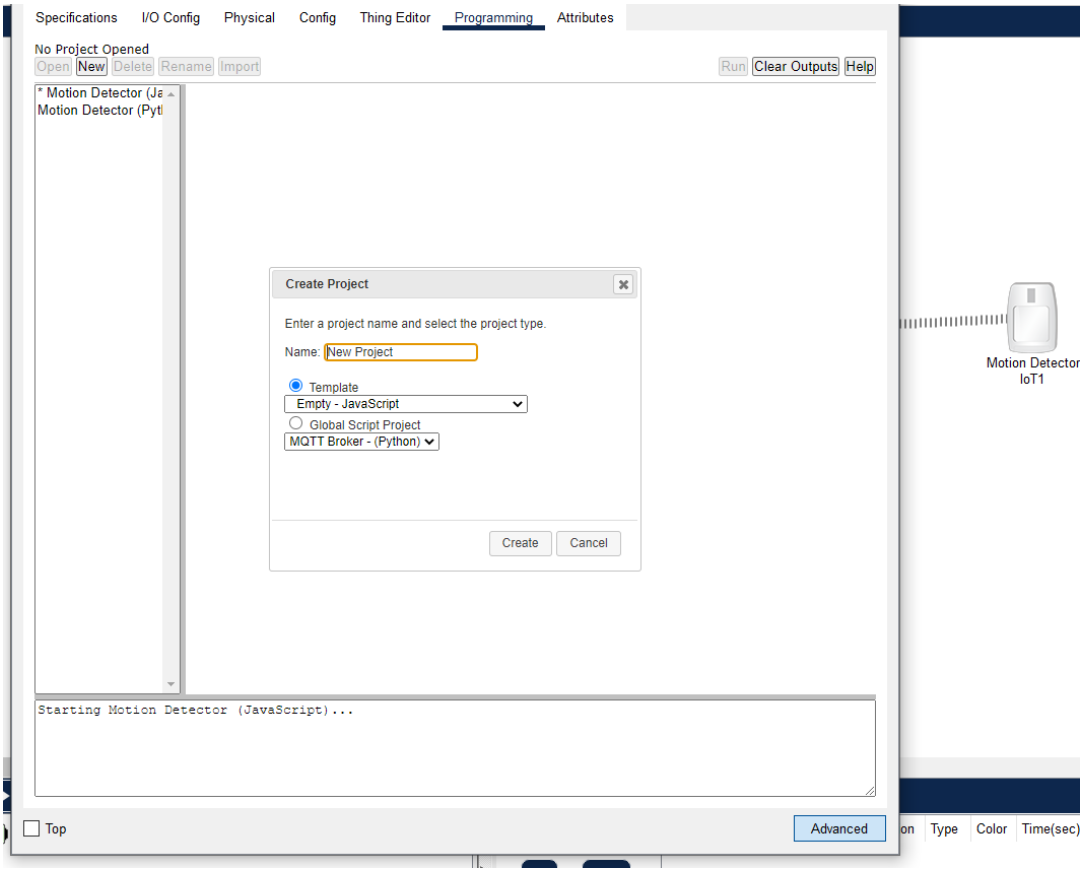

Рисунок 3.26. Вкладка «Programming»

Програмування IoT-пристрій у Cisco Packet Tracer відбувається таким чином:

- Вибрати IoT-пристрій у розділі (End Devices / Smart Devices).

- Вибрати пристрій, що підтримує програмування, наприклад: Smart Light (розумна лампа), Temperature Sensor (датчик температури), Microcontroller Board (IoT Board) – гнучкий пристрій, що може взаємодіяти з іншими IoT-девайсами.
- Перетягни пристрій у робочу область.
- Підключити пристрій до мережі.
- Додати хаб або маршрутизатор.
- Підключити IoT-пристрій через Wi-Fi або Ethernet.
- Переконатися, що пристрій має IP-адресу (перевір у вкладці "Config").
- Клацнути на пристрій, вибрати вкладку Programming.
- У списку Scripts обрати один із вбудованих скриптів або створити власний.
- Обрати New або відредагувати наявний скрипт.
- Запустити симуляцію.
- Зміни параметри датчика (наприклад, піднеси об’єкт до датчика руху або збільши температуру).
- Переконатися, що пристрій правильно реагує.
- Якщо потрібно змінити код повертаємося у вкладку Programming.

Packet Tracer дозволяє створювати інтерактивні сценарії, де IoT-пристрої можуть взаємодіяти з іншими елементами мережі. Взаємодія з мережевими пристроями дозволяє створювати більш складні IoT-системи, де пристрої не тільки виконують локальні операції, а й можуть обмінюватися інформацією через мережу або хмару.

В Packet Tracer є вбудовані інструменти для моніторингу та налагодження програм IoT-пристроїв. Користувач можете переглядати виведені значення з датчиків, переглядати статус пристроїв і відстежувати, як вони реагують на зміни в оточуючому середовищі. Програмування підтримує JavaScript та Python. JavaScript використовується для базових сценаріїв взаємодії.

РОЗДІЛ 4
РОЗРОБКА ТА РЕАЛІЗАЦІЯ ПРОЕКТІВ ІНТЕРНЕТУ РЕЧЕЙ ЗАСОБАМИ CISCO PACKET TRACER

4.1. IoT-ситема клімат-контролю «Розумного будинку»

Метою даної системи є створення інтегрованої моделі клімат-контролю, яка здатна автоматично підтримувати комфортні умови в приміщенні, орієнтуючись на поточні показники навколишнього середовища. У межах даного проєкту використано середовище Cisco Packet Tracer, що надає широкі можливості для візуального моделювання мереж, взаємодії між датчиками, виконавчими пристроями, шлюзами зв'язку та центральним сервером. Це дозволяє реалізувати повноцінну симуляцію роботи системи, перевірити логіку взаємодії компонентів і проаналізувати ефективність алгоритмів автоматичного керування.

Структура системи клімат-контролю включає кілька інтелектуальних компонентів, об'єднаних у єдину мережу через Home Gateway – центральний вузол, який забезпечує комунікацію між усіма пристроями та сервером Інтернету речей. До складу системи входить датчик температури, який постійно відстежує поточний стан мікроклімату в приміщенні, а також виконавчі пристрої – вентилятор, піч та автоматизоване вікно. Отримані від датчиків дані передаються до шлюзу, де обробляються згідно з визначеними правилами.

На основі цих даних система приймає рішення про необхідність вмикання або вимикання певних пристроїв: якщо температура перевищує встановлене значення – вмикається вентилятор або відкривається вікно для охолодження повітря, якщо ж температура опускається нижче комфортного рівня – активується піч або система опалення. Такий підхід дозволяє підтримувати стабільний мікроклімат у приміщенні без участі людини, а при цьому мінімізує витрати енергії.

У таблиці 4.1. подано компоненти системи клімат-контролю.

Таблиця 4.1. Компоненти системи клімат-контролю

Компонент	**Тип пристрою**	**Функціональне призначення**	**Технічні характеристики**
Home Gateway	IoT-шлюз	Передача даних між локальною мережею та сервером	IP-з’єднання, сумісність з сервером
Temperature Meter	Аналоговий сенсор	Вимірювання температури середовища в реальному часі	Діапазон вимірювання: −20 °C – +60 °C
Smart Fan	Актуатор (вентилятор)	Активне охолодження приміщення при перевищенні температурного порогу	Напруга живлення – 5 В
Furnace	Актуатор (піч)	Нагрів приміщення при зниженні температури нижче порогового значення	Потужність – 1.2 кВт; цифрове керування
Window	Актуатор (вікно)	Відкривання для вентиляції при підвищенні температури; закривання при охолодженні	Електропривід, цифровий сигнал керування
Tablet	Користувацький пристрій	Віддалене керування та моніторинг через Monitor	Підключення через Wi-Fi
IoT Server	Хмарна платформа	Обробка даних від Home Gateway, збереження сценаріїв автоматизації	Web-інтерфейс, підтримка MQTT-протоколу

Архітектура та алгоритм роботи системи.

Функціональна схема системи клімат-контролю побудована за принципом централізованого керування через Home Gateway, який об’єднує всі пристрої в єдину

мережу (рис. 4.1). Датчик Temperature Meter зчитує поточну температуру повітря та передає дані на шлюз, який у свою чергу взаємодіє з IoT Server (рис. 4.2). Сервер аналізує показники та активує відповідні пристрої за заданими правилами:

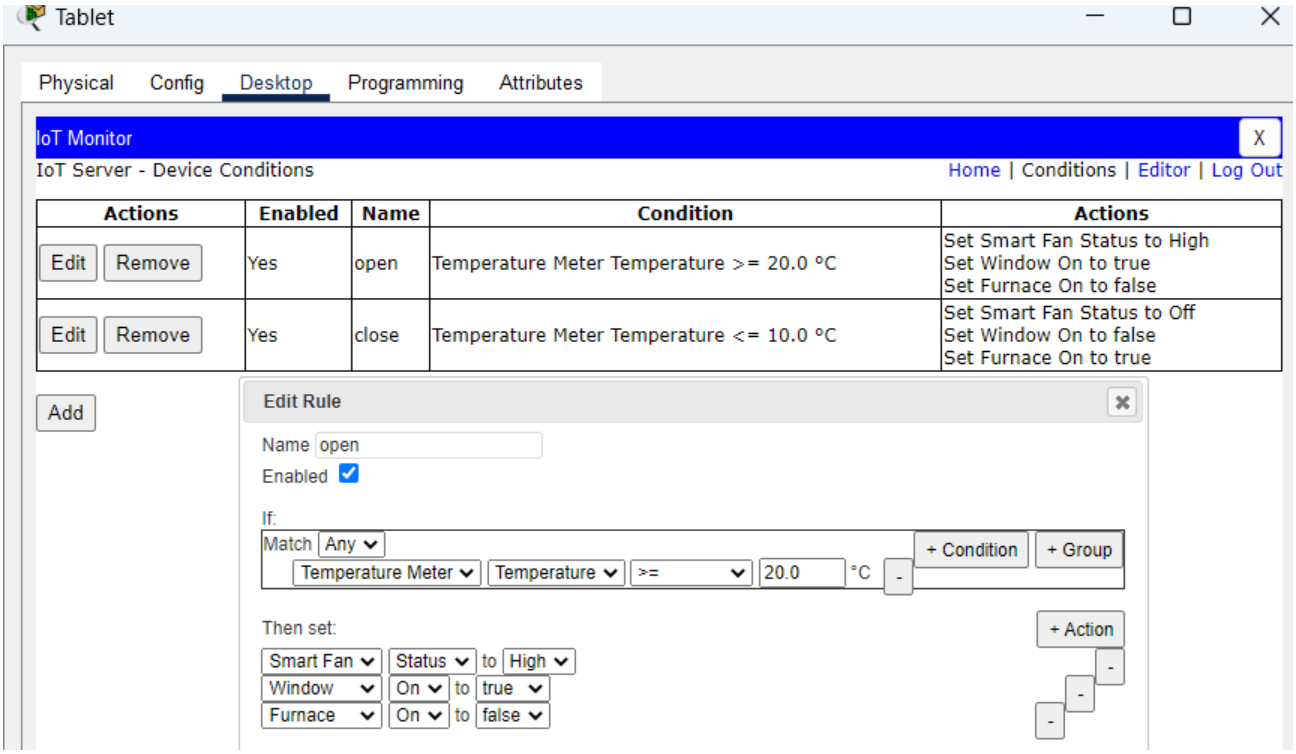

Рис. 4.1. Алгоритм роботи системи термо-контролю

- якщо температура перевищує 20 °C – відкривається вікно та вмикається вентилятор;
- якщо температура знижується нижче 10 °C – вікно зачиняється, вентилятор вимикається, а піч активується;
- якщо температура знаходиться в діапазоні 10–20 °C – система переходить у режим очікування.

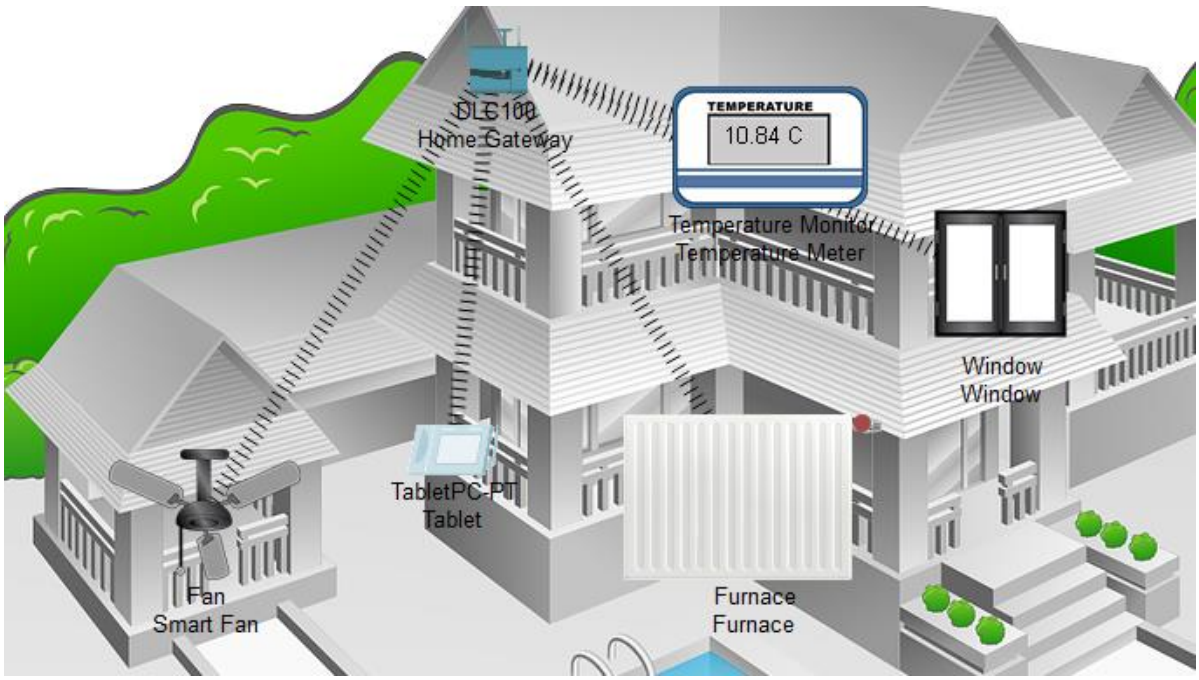

Рис. 4.2. Схема взаємодії компонентів системи
клімат-контролю в середовищі Cisco Packet Tracer

У таблиці 4.2. подано алгоритм роботи системи термо-контролю

Таблиця 4.2. Алгоритм роботи системи термо-контролю

Умова (If Condition)	**Дія (Then Action)**
Temperature ≥ 20	Window = ON; Fan = ON; Furnace = OFF
Temperature ≤ 10	Window = OFF; Fan = OFF; Furnace = ON
10 < Temperature < 20	Усі пристрої в режимі Idle

Тестування та результати роботи.

Після побудови моделі системи клімат-контролю було проведено етап тестування у середовищі Cisco Packet Tracer, який мав на меті перевірити правильність взаємодії між усіма компонентами, а також оцінити ефективність автоматичного регулювання температури (рис. 4.3).

Рис. 4.3. Вікно моніторингу IoT-Server із відображенням станів пристроїв системи клімат-контролю

У процесі експерименту система демонструвала стабільну роботу в динамічному режимі, оперативно реагуючи на коливання показників сенсора температури. Коли значення температури підвищувалося понад 20 °C, спрацьовували відповідні алгоритми автоматичного охолодження: вікно відкривалося, забезпечуючи природну вентиляцію, а вентилятор автоматично вмикався, створюючи спрямований потік повітря для прискореного зниження температури в кімнаті. Такий механізм дозволив ефективно стабілізувати мікроклімат без необхідності ручного втручання користувача.

У протилежному випадку, коли температура опускалася до 10 °C, система переходила у режим обігріву. Вентилятор автоматично вимикався, вікно зачинялося, щоб уникнути втрати тепла, після чого активувалася піч (Heating Unit), яка розпочинала підігрів повітря в приміщенні. Завдяки цьому забезпечувалося поступове повернення температури до комфортного рівня.

Усі дії системи супроводжувалися візуальними змінами в інтерфейсі Cisco Packet Tracer – зміна кольору індикаторів, активність піктограм пристроїв і динамічне

оновлення даних у Monitor. Останній дозволяв у реальному часі відстежувати поточні показники сенсора температури, стан виконавчих пристроїв (вентилятора, печі, вікна) та швидкість реакції системи.

За результатами випробувань було встановлено, що середній час реакції системи на зміну температури становив менше 2 секунд, що є відмінним показником для автоматизованих систем цього типу. Така швидкість забезпечує високу точність регулювання мікроклімату й гарантує комфорт користувачів навіть у разі різких коливань зовнішніх умов.

Аналіз ефективності та переваги системи.

Запропонована система клімат-контролю, побудована на основі технологій Інтернету речей має низку важливих переваг, що забезпечують її ефективність, надійність і гнучкість у використанні.

Автоматизація процесів. Система повністю усуває необхідність постійного ручного втручання користувача. Усі процеси – від зчитування показників температури до ввімкнення або вимкнення пристроїв – відбуваються автоматично, відповідно до встановлених порогових значень.

Завдяки інтелектуальній логікі керування пристрої (вентилятор, піч, автоматичне вікно) активуються лише за необхідності. Наприклад, при стабільній комфортній температурі система не витрачає енергію на додаткове охолодження чи нагрів. Це дозволяє істотно зменшити споживання електроенергії, продовжити термін служби обладнання й зробити систему екологічно безпечнішою.

У ході тестування доведено, що розроблена система забезпечує стабільне підтримання комфортної температури в приміщенні, демонструє високу точність реагування на зміни середовища та економно витрачає енергоресурси.

Програмна реалізація системи.

Програмна частина моделі реалізована за допомогою IoT Server у середовищі Cisco Packet Tracer. Для кожного пристрою (вентилятора, печі, вікна, сенсора температури) у вікні Monitor були створені окремі правила автоматизації, які визначають реакцію системи на зміни температури.

Кожне правило включає умову (якщо температура більше / менше встановленого значення) та відповідну дію (увімкнути або вимкнути певний пристрій). Завдяки цьому реалізується логіка автоматичного керування, що забезпечує узгоджену роботу всіх компонентів системи.

Даний сценарій демонструє прості, але ефективні алгоритми, що лежать в основі роботи клімат-контролю. Вони можуть бути розширені, наприклад, додаванням часових умов, пріоритетів або залежностей від інших сенсорів. Така гнучкість дозволяє адаптувати систему під будь-які потреби користувача та специфіку приміщення.

4.2. IoT-система протипожежного захисту складу

Забезпечення пожежної безпеки на промислових, складських та виробничих об’єктах є однією з ключових складових комплексної системи охорони життя та майна. У таких умовах надзвичайно важливим є своєчасне виявлення загоряння, адже навіть кілька секунд затримки можуть призвести до масштабних збитків або загрози життю працівників. Саме тому сучасні технології Інтернету речей відкривають нові перспективи у створенні інтелектуальних систем автоматичного пожежогасіння, здатних самостійно аналізувати стан середовища, виявляти появу диму, активувати спринклери й оперативно сповіщати користувачів про небезпеку.

Завдяки застосуванню IoT-технологій відбувається інтеграція сенсорів, виконавчих пристроїв і серверів моніторингу в єдину мережу, що дозволяє системі реагувати на надзвичайні ситуації практично миттєво. У порівнянні з традиційними пожежними сигналізаціями, такі рішення є більш гнучкими, масштабованими та здатними до самодіагностики.

Метою даного проєкту є створення інтегрованої моделі автоматизованої системи пожежогасіння для складського приміщення з використанням середовища Cisco Packet Tracer. Це програмне забезпечення дає змогу не лише відтворити віртуальну архітектуру мережі, але й перевірити її працездатність у різних сценаріях

– від нормальної роботи до аварійних ситуацій, з імітацією появи диму та активації відповідних виконавчих пристроїв.

Склад та архітектура системи.

Розроблена модель системи пожежогасіння базується на централізованій архітектурі керування, у якій усі елементи поєднані через Home Gateway – центральний комунікаційний вузол, що забезпечує обмін даними між пристроями, сервером та користувачем. Саме через цей шлюз відбувається передача сигналів від сенсорів до виконавчих пристроїв і системи оповіщення.

До складу системи входять такі основні компоненти:

1. Датчики диму (Smoke Detectors). Вони виконують функцію первинного моніторингу повітряного середовища. Кожен сенсор здатний фіксувати підвищення концентрації диму в повітрі, що є головною ознакою початку займання. У разі спрацювання датчик негайно надсилає сигнал на шлюз Home Gateway для подальшої обробки.
2. Спринклери (Fire Sprinklers). Це виконавчі пристрої, які активуються автоматично після підтвердження сигналу про задимлення. Спринклери імітують подачу води або іншого вогнегасного середовища, тим самим забезпечуючи локалізацію осередку займання до прибуття пожежної команди.
3. Сирена (Siren). Призначена для звукового сповіщення персоналу про виявлення пожежі. У момент активації система видає гучний звуковий сигнал, який дозволяє швидко евакуювати людей із небезпечної зони.
4. Ноутбук адміністратора (Laptop). Виконує функцію центру моніторингу та управління. За допомогою інтерфейсу Monitor користувач може в реальному часі відстежувати стан кожного пристрою, переглядати сповіщення, активувати або деактивувати окремі компоненти, а також проводити діагностику системи.
5. Home Gateway. Служить ключовою ланкою зв’язку між сенсорами, спринклерами, сиреною та ноутбуком адміністратора. Забезпечує обмін

даними в обох напрямках, дозволяючи системі функціонувати автономно навіть за відсутності прямого втручання людини.

Функціональна схема роботи системи побудована за принципом «виявлення – реагування – сповіщення». Після виявлення диму сенсор передає сигнал на шлюз, який автоматично активує спринклери для гасіння пожежі й одночасно запускає сирену для оповіщення персоналу. Паралельно на ноутбуці адміністратора з'являється повідомлення про інцидент, що дає змогу оперативно оцінити ситуацію та вжити необхідних заходів.

Усі пристрої працюють у рамках єдиної IoT-мережі, що забезпечує стабільність і синхронізацію дій (рис. 4.4). Такий підхід дозволяє створити систему, здатну до автономного функціонування, високої надійності та швидкої реакції на будь-яку загрозу займання.

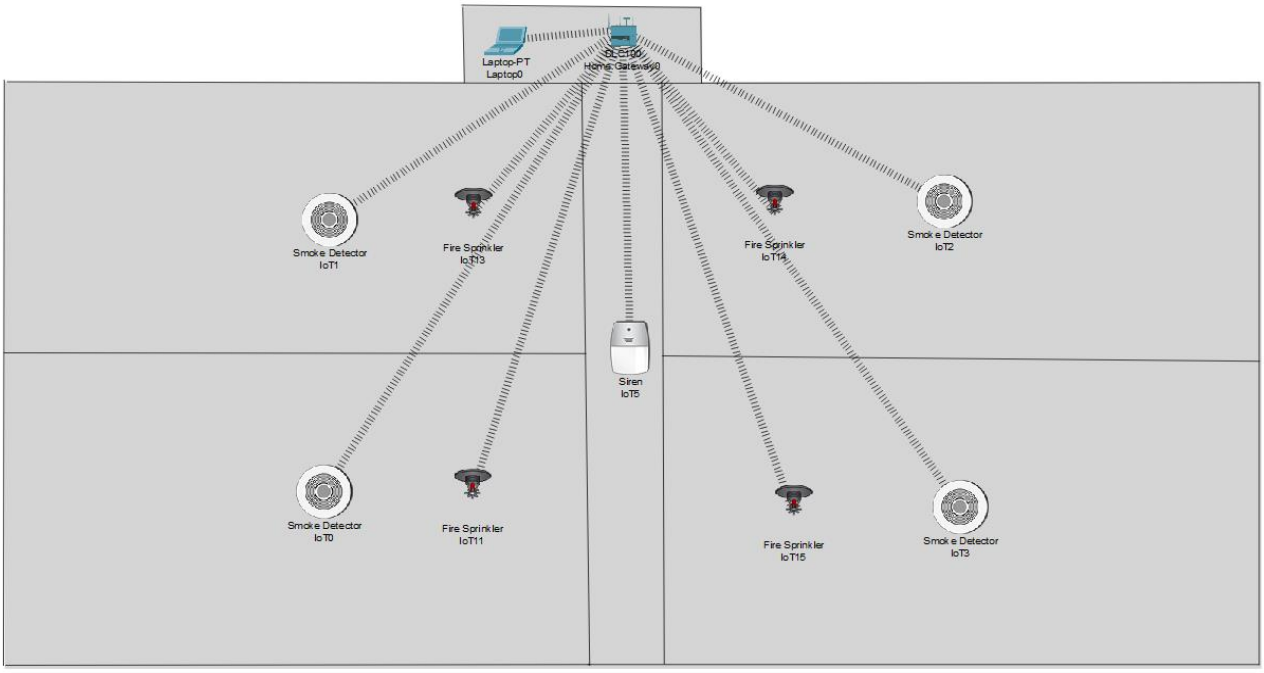

Рис. 4.4. Схема підключення компонентів системи пожежогасіння

Умовно склад поділений на дві секції: верхню і нижню, кожна з яких обладнана власними сенсорами та спринклерами. Датчики диму передають сигнали на Home Gateway, який через IoT Server активує спринклери та сирену в разі виявлення загоряння. У таблиці 4.3. подано компоненти системи пожежогасіння.

Таблиця 4.3. Компоненти системи пожежогасіння

Компонент	**Тип пристрою**	**Функціональне призначення**	**Технічні характеристики**
Smoke Detector	Сенсор (аналоговий)	Виявлення наявності диму в зоні контролю	Діапазон чутливості: 0 –1000 одиниць диму
Fire Sprinkler	Актуатор	Автоматичне розпилення води при спрацюванні сенсора диму	Напруга живлення: 12 В; цифрове керування (увімкнено/вимкнено)
Siren	Актуатор	Генерація звукового сигналу	Гучність: 100–110 дБ; цифрове керування
Home Gateway	IoT-шлюз	З’єднання всіх пристроїв системи через бездротову мережу Wi-Fi	Підтримка IoT-протоколів
Laptop-PT	Користувацький пристрій	Моніторинг стану сенсорів і керування системою	З’єднання через Wi-Fi
Old Car	Об’єкт середовища	Імітація джерела задимлення у складському приміщенні	Використовується для тестування роботи системи

Принцип роботи системи.

Система пожежогасіння працює за принципом зворотного зв’язку між сенсорами диму та виконавчими пристроями. У нормальному стані всі пристрої знаходяться в режимі очікування (рис. 4.5). У разі виявлення диму будь-яким із сенсорів система миттєво передає сигнал через Home Gateway на IoT Server, який виконує наступні дії:

- активує спринклери (Fire Sprinklers) для гасіння вогню;
- вмикає сирену (Siren) для сповіщення персоналу;
- надсилає повідомлення на ноутбук адміністратора для фіксації інциденту.

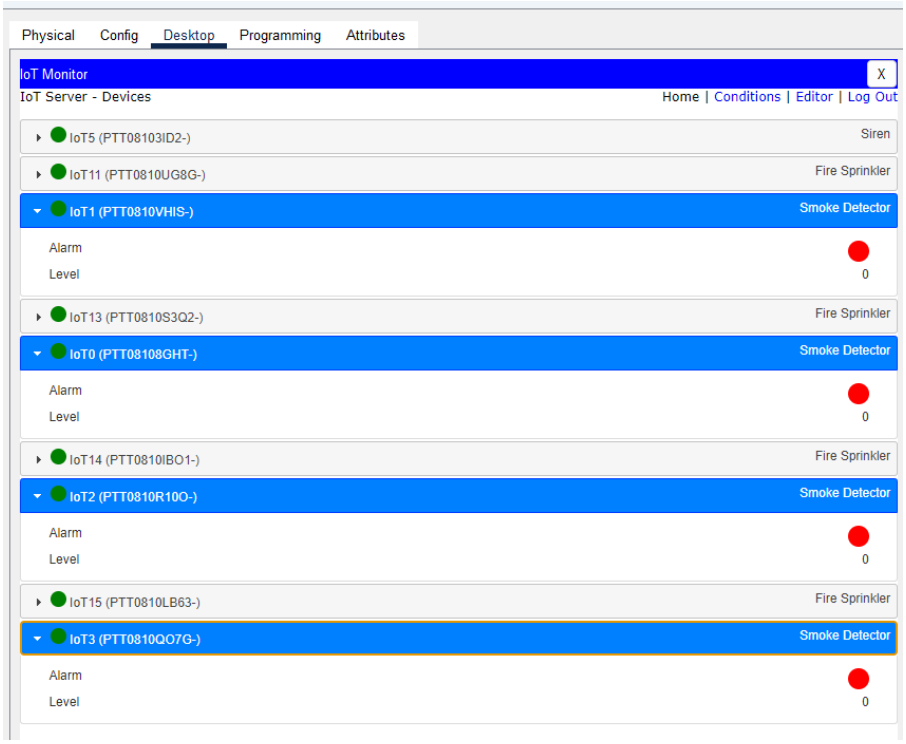

Рис. 4.5. Логічна взаємодія сенсорів, шлюзу та виконавчих пристроїв у системі пожежогасіння

Алгоритм автоматизації на IoT Server.

У середовищі Cisco Packet Tracer було створено набір автоматизованих правил (conditions), які визначають поведінку системи пожежогасіння в різних ситуаціях. Ці правила виконують функцію логічного ядра системи – саме вони забезпечують автоматичну реакцію виконавчих пристроїв на зміну стану сенсорів.

Усі правила налаштовано через інтерфейс Server → Monitor, де кожен сценарій має чітко визначену умову (IF Condition) та дію (THEN Action). У таблиці 4.3. представленні правила налаштування сценаріїв

Таблиця 4.3. Правила налаштування сценаріїв

Назва сценарію	**Умова (IF Condition)**	**Дія (THEN Action)**	**Опис процесу**
fire_detected	Smoke Detector = TRUE	Siren = ON; Fire Sprinklers = ON	Виявлено дим – система активує сирену і спринклери
fire_clear	Smoke Detector = FALSE	Siren = OFF; Fire Sprinklers = OFF	Дим зник – система переходить у режим очікування

Наведений алгоритм забезпечує миттєву реакцію системи у разі виявлення ознак загоряння. У момент, коли сенсор диму (Smoke Detector) фіксує підвищення концентрації частинок диму в повітрі, відбувається передача сигналу до IoT Server, який одразу виконує визначені дії: вмикає сирену та активує спринклери. Коли ж сенсор більше не фіксує диму, система автоматично повертається до режиму очікування, вимикаючи всі активні пристрої.

Процес перевірки умов відбувається циклічно з інтервалом у 1 секунду, що гарантує високу оперативність реагування навіть при незначному появленні диму. Така частота оновлення дозволяє системі своєчасно розпізнавати загрозу та активувати захисні заходи ще до того, як вогонь розгориться.

Тестування системи.

Після створення та налаштування алгоритмів автоматизації було проведено повне тестування моделі системи пожежогасіння у середовищі Cisco Packet Tracer. Метою випробувань було перевірити коректність спрацьовування умов, стабільність роботи зв’язку між пристроями та реалістичність реакції системи в аварійній ситуації.

Для цього був змодельований сценарій загоряння автомобіля Old Car, розташованого у нижній частині складського приміщення. На початку експерименту всі пристрої перебували у стані спокою, тобто сирена й спринклери були вимкнені, а сенсори не фіксували диму.

У момент, коли датчик Smoke Detector IoT10 виявив появу диму, система спрацювала наступним чином:

1. На сервер автоматично надійшов сигнал «Smoke detected», який активував сценарій *fire_detected*.
2. Сирена IoT5 увімкнулася та почала видавати гучний звуковий сигнал, що слугує попередженням для персоналу.
3. Спринклери IoT11 та IoT15 активувалися, імітуючи процес розпилення води над зоною загоряння.
4. На екрані Laptop-PT у вкладці *Monitor* з'явилося повідомлення про тривогу, а індикатори пристроїв у віртуальній схемі змінили свій стан на активний (позначення зеленим або червоним кольором).

Після завершення симуляції й припинення дії диму сенсор повернувся до початкового стану (Smoke Detector = FALSE), і система автоматично деактивувала спринклери та сирену, переходячи у режим очікування.

Під час тестування було зафіксовано, що реакція системи займала не більше 1–2 секунд, що є відмінним показником для подібних автоматизованих IoT-моделей. Усі пристрої працювали узгоджено, без затримок або помилкових спрацювань, що свідчить про високу стабільність і надійність створеної системи.

Результати тестування підтвердили, що розроблена модель повністю відповідає поставленій меті: вона здатна ефективно виявляти загоряння, оперативно активувати системи пожежогасіння й забезпечувати інформування користувачів у реальному часі (рис. 4.6). Таким чином, система може бути використана як практичний прототип для впровадження у реальних умовах на промислових і складських об'єктах.

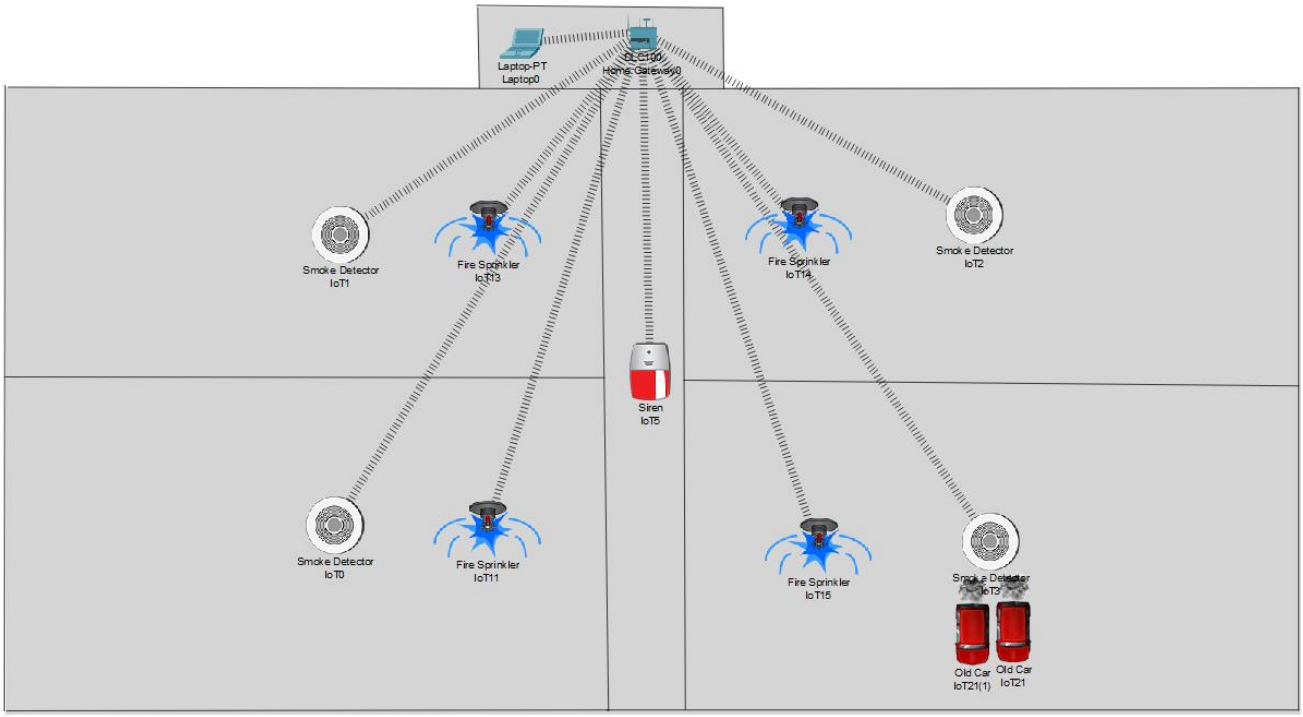

Рис. 4.6. Симуляція спрацювання сенсорів диму
та активації спринклерів при пожежі

Аналіз ефективності системи.

Розроблена система пожежогасіння характеризується високою надійністю, швидкодією та простотою реалізації. Основні переваги рішення:

- Автоматична робота – не потребує ручного втручання людини.
- Висока чутливість – спрацювання при мінімальній кількості диму.
- Енергоефективність – система активується лише в момент небезпеки.
- Можливість розширення – легко додати додаткові сенсори або виконавчі пристрої.
- Інтеграція з іншими системами, такими як відеоспостереження або контроль доступу.

Розроблена система пожежогасіння у середовищі Cisco Packet Tracer успішно продемонструвала здатність до автоматичного виявлення пожежі та миттєвої реакції у вигляді активації сирени та спринклерів.

4.3. IoT-система автоматичного поливу газону

Сучасні системи «Розумного будинку» (Smart Home) виходять далеко за межі контролю внутрішнього мікроклімату та управління освітленням. Одним із важливих напрямів розвитку таких технологій є автоматизація догляду за навколишньою територією, зокрема – за газонами, клумбами чи декоративними насадженнями.

Одним із ключових компонентів у цьому напрямі виступає система автоматичного поливу газону, яка дозволяє підтримувати оптимальний рівень вологості ґрунту без постійної участі людини. Завдяки використанню датчиків вологості, мікроконтролера та виконавчих пристроїв система самостійно визначає потребу у поливі та вмикає дощувальні установки лише тоді, коли це необхідно.

Таке рішення має низку переваг:

- Економія водних ресурсів – зрошення відбувається лише у разі потреби, що запобігає перевитраті води.
- Збереження здоров’я рослин – підтримується стабільна вологість, що сприяє рівномірному росту трав’яного покриву.
- Автономність і зручність – користувачу не потрібно вручну вмикати полив або контролювати стан ґрунту.
- Можливість масштабування – систему можна доповнювати новими зонами поливу, сенсорами чи алгоритмами оптимізації.

Архітектура системи.

Проєкт системи автоматичного поливу створено у середовищі Cisco Packet Tracer з використанням можливостей моделювання пристроїв Інтернету речей. На відміну від попередніх рішень, дана модель працює без шлюзу Home Gateway, тобто комунікація та управління всіма компонентами здійснюються безпосередньо через мікроконтролер MCU-PT, який виконує роль центрального обчислювального вузла.

Основні елементи системи включають:

1. Мікроконтролер (MCU-PT).

Є основним елементом керування, який отримує дані від сенсорів і відповідно до заданих умов керує виконавчими пристроями. Усі обчислення виконуються локально, без необхідності підключення до серверів чи шлюзів.

2. Датчик вологості ґрунту (Generic Environment Sensor).

Використовується для вимірювання рівня вологості у зоні поливу. Сенсор постійно передає значення до мікроконтролера. Якщо рівень вологості опускається нижче встановленого порогу система розпізнає це як сигнал до запуску поливу.

3. Дощувальні установки (Lawn Sprinklers).

У системі реалізовано чотири спринклери, підключені до цифрових виходів мікроконтролера. Вони вмикаються одночасно у разі потреби поливу та вимикаються після досягнення необхідного рівня вологості.

4. Датчик води (Water Detector).

Використовується для перевірки наявності подачі води до системи. Якщо датчик фіксує відсутність води у трубопроводі, мікроконтролер блокує запуск спринклерів, запобігаючи виходу системи з ладу або порожньому обертанню насосів.

Усі елементи з’єднані провідними лініями зв’язку, що забезпечує стабільність і відсутність затримок у передачі сигналів. Модель не вимагає мережевого підключення або зовнішніх серверів, тому її можна віднести до автономних рішень локального рівня.

Принцип роботи системи.

Принцип роботи базується на циклічному зчитуванні показників вологості та прийнятті рішень на основі встановлених порогових значень.

Алгоритм дії виглядає так:

1. Мікроконтролер отримує поточні дані від сенсора вологості.
2. Якщо виявлено, що значення нижче порогу (наприклад, <300), контролер увімкне всі спринклери, розпочавши полив.
3. Після підвищення вологості вище порогового рівня система автоматично вимикає спринклери, переходячи в режим очікування.
4. Якщо датчик води повідомляє про її відсутність, полив не активується, що підвищує безпеку системи.

Таким чином, система забезпечує повністю автоматизоване керування процесом поливу, реагуючи лише на реальні зміни стану середовища, без необхідності участі користувача (рис. 4.7).

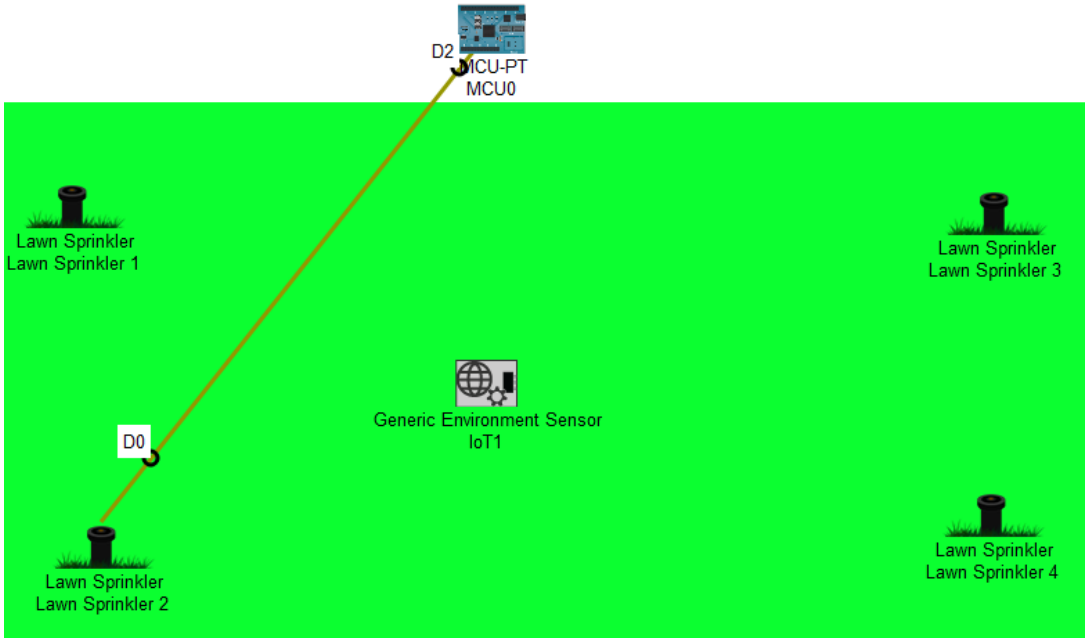

Рис. 4.7. Схема підключення компонентів системи автоматичного поливу газону у середовищі Cisco Packet Tracer

У таблиці 4.4. подано компоненти системи автоматичного поливу газону.

Таблиця 4.4. Компоненти системи автоматичного поливу газону

Компонент	**Тип пристрою**	**Функціональне призначення**	**Технічні характеристики**
MCU-PT	Мікроконтролер	Обробка сигналів від сенсора вологості та керування спринклерами	6 цифрових і 6 аналогових портів, частота 16 МГц
Generic Environment Sensor	Аналоговий сенсор	Вимірювання вологості ґрунту	Діапазон: 0–1023, чутливість ±5 %, аналоговий вихід

Компонент	Тип пристрою	Функціональне призначення	Технічні характеристики
Lawn Sprinkler 1–4	Актуатори	Розпилення води при зниженні вологості ґрунту нижче порогу	Напруга живлення 12 В, цифрове керування (ON/OFF)

Принцип роботи системи.

Система автоматичного поливу функціонує за принципом зворотного зв’язку між датчиком вологості та виконавчими пристроями. Датчик постійно вимірює вологість ґрунту та передає аналоговий сигнал на мікроконтролер. Якщо рівень вологості нижчий за порогове значення (300 одиниць), контролер активує всі спринклери. Коли показник зростає вище порогу або виявляється наявність води (через Water Detector, підключений до цифрового порту D0), система припиняє полив.

Алгоритм та програмна реалізація.

Для реалізації логіки роботи використано вбудоване середовище програмування мікроконтролера в Cisco Packet Tracer. Програмний код (рис.4.8) написано мовою JavaScript (файл main.js).

Програма реалізує циклічний контроль вологості з інтервалом у 1 секунду. Під час тестування в середовищі симулятора підтверджено, що при зниженні вологості нижче 300 од. усі спринклери активуються, а після досягнення оптимального рівня – автоматично вимикаються.

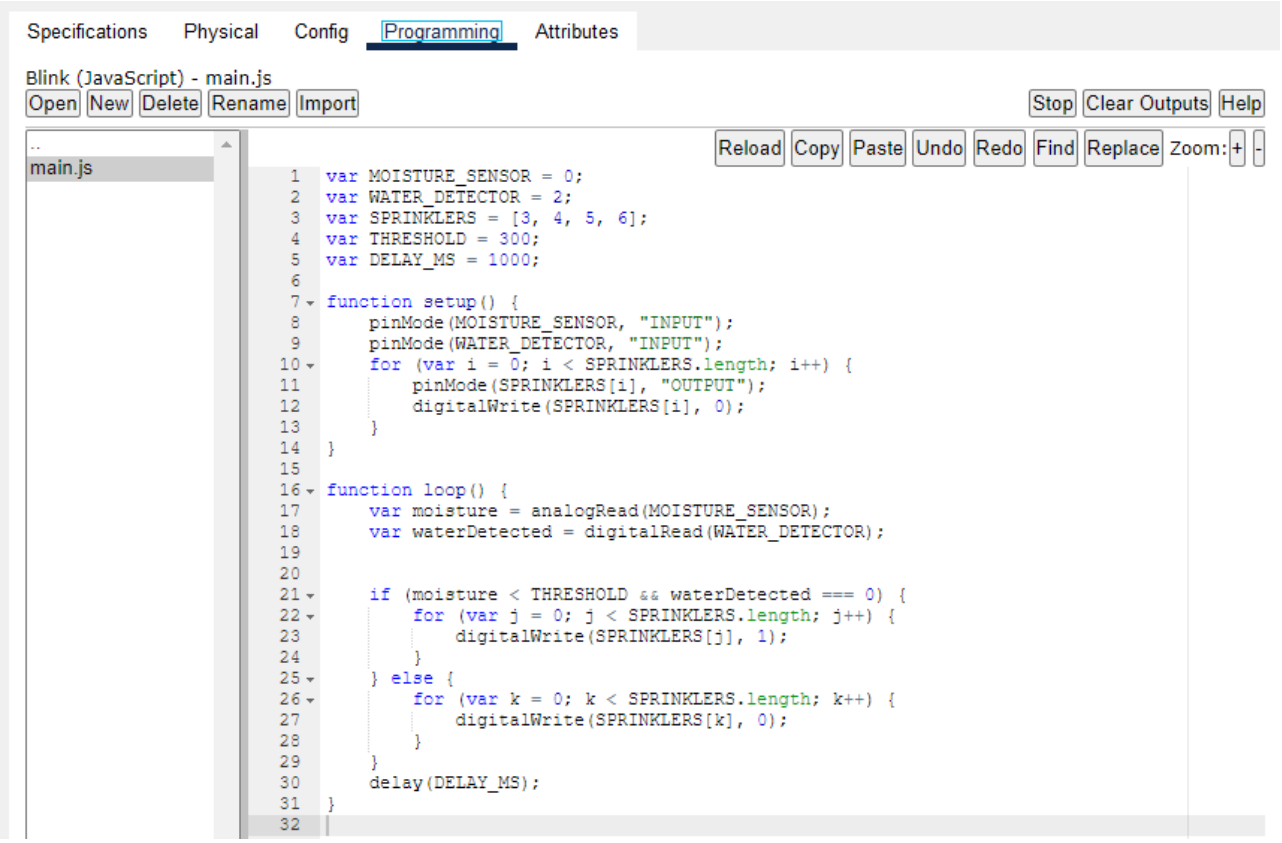

Рис. 4.8. Інтерфейс програмування мікроконтролера MCU-PT
та робота алгоритму автоматичного поливу

Результати моделювання та переваги системи.

Проведене моделювання показало, що система стабільно реагує на зміни вологості, забезпечує рівномірний полив і запобігає перевитраті води. Серед основних переваг розробленого рішення можна виділити:

- Повна автономність роботи – не потребує шлюзу або зовнішнього сервера.
- Точність керування – реакція лише за фактичним станом ґрунту.
- Енерго- та водоефективність – споживання ресурсів тільки за необхідності.
- Масштабованість – можливість підключення додаткових сенсорів або зон поливу.
- Простота програмної реалізації – короткий і надійний алгоритм із мінімальним споживанням ресурсів мікроконтролера.

Завдяки застосуванню мікроконтролера MCU-PT, датчика вологості та чотирьох спринклерів, система забезпечує повністю автономне керування процесом зрошення. Алгоритм реагує на зміни вологості в реальному часі та автоматично припиняє полив при досягненні оптимальних значень (рис. 4.9).

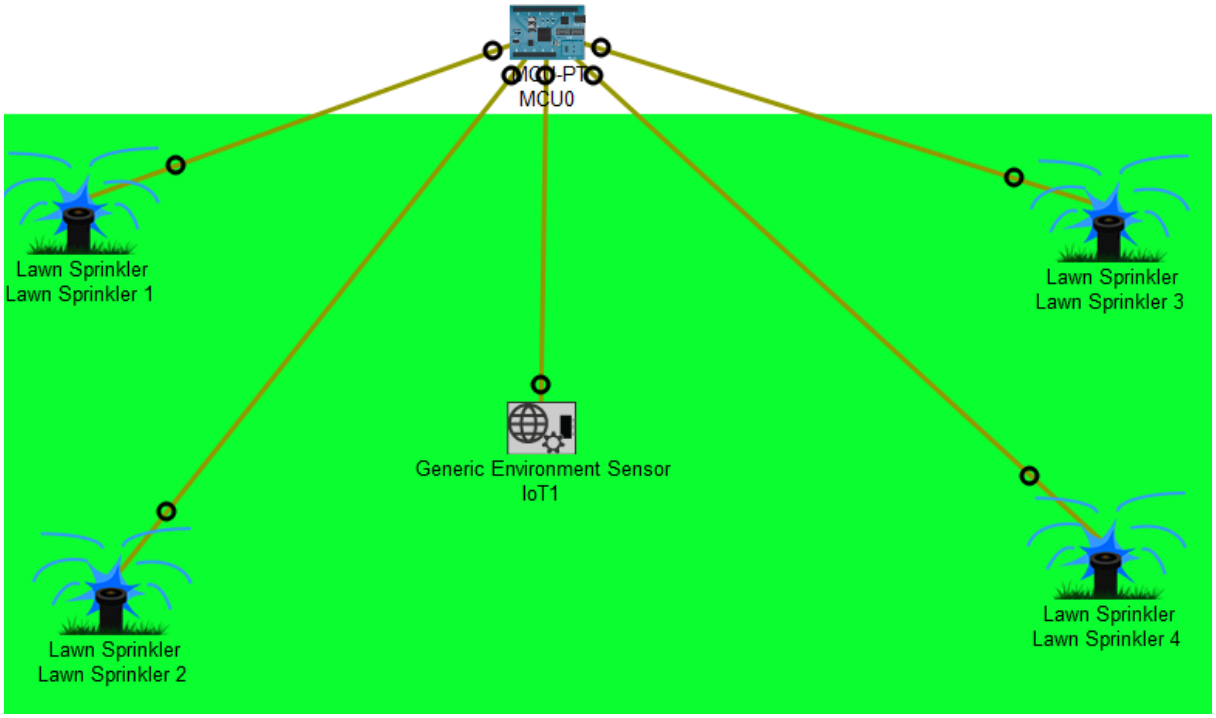

Рис. 4.9. Робота системи автополиву

Розроблена система автоматичного поливу може бути ефективно впроваджена у різних сферах господарської діяльності. Передусім вона підходить для приватних домогосподарств, де забезпечує автоматичний догляд за присадибними ділянками, клумбами чи газонами.

Таке рішення є не лише зручним, а й економічно вигідним, оскільки мінімізує витрати води та електроенергії, а також дозволяє запобігти пересушуванню або надмірному зволоженню рослин. Система може працювати цілодобово у повністю автономному режимі, що робить її особливо корисною у віддалених або важкодоступних місцях.

ВИСНОВКИ

Інтернет речей – концепція мережі, що складається із взаємозв'язаних фізичних пристроїв, які мають вбудовані датчики та програмне забезпечення, яке дозволяє здійснювати передачу і обмін даними в автоматичному режимі.

В першому розділі кваліфікаційної роботи досліджено основи та концепцію технології Інтернет речей.

В другому розділі детально досліджено та проаналізовано екосистему Інтернету речей та проблеми безпеки даних IoT. Розглянуто архітектурні рівні IoT та описано специфічні механізми захисту і вразливості на мережевому і прикладному рівнях.

В третьому розділі кваліфікаційної роботи досліджено програмну складову проєктування IoT, а саме: мережевий симулятор Cisco Packet Tracer. Розглянуто і описано його програмний та інструментальний інтерфейс, основні компоненти для моделювання. Детально проаналізовано вбудовані засоби програмування логіки IoT-пристроїв за допомогою мов JavaScript.

В четвертому розділі кваліфікаційної роботи розроблено та реалізовано три практичні моделі IoT-систем у середовищі Cisco Packet Tracer. Здійснено проєктування та налаштування: системи клімат-контролю для «Розумного будинку» на базі Home Gateway та IoT Server; автоматизованої системи пожежогасіння для складського приміщення, що також використовує IoT Server для моніторингу та реагування; автономної системи автоматичного поливу газону на базі мікроконтролера MCU-PT з локальною програмною логікою. Для кожної моделі було проведено тестування та симуляцію роботи, що підтвердило коректність розроблених алгоритмів та архітектурних рішень.

Отже, при виконанні кваліфікаційної роботи, було виконано наступні завдання: досліджено концепцію технології Інтернет речей, описано екосистему IoT і основні проблеми безпеки даних, розглянуто інструментарій симулятора Cisco Packet Tracer для проєктування та програмування IoT-пристроїв, розроблено та реалізовано три IoT-системи з різними архітектурними підходами (централізованим та автономним), проведено симуляцію і тестування та аналіз ефективності розроблених IoT-систем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. M. Weyrich, C. Ebert, «Reference architectures for the internet of things», IEEE Software, 2016, Vol. 33, № 1, P. 112 – 116.

2. Пиріг Ю. В., Кайдан М. В., Гордійчук-Бублівська О. В. Аналіз концепції інтернету речей та динаміки її розвитку в різних галузях. DOI 10.36994/2707-4110-2020-1-28-15 Вісник Унівеситету «Україна», № 1 (24), 2020, с. 171-185.

3. Літій-іонних батарей скоро не буде – топ-6 альтернатив. URL: https://focus.ua/uk/digital/681923 (дата звернення: 06.11.2024).

4. What is PaaS? Platform as a service definition and guide. URL: https://www.techtarget.com/searchcloudcomputing/definition/Platform-as-a-Service-PaaS (дата звернення: 08.11.2024).

5. Історія інтернету від перших мереж до сучасності. URL: https://ranok-portal.com.ua/news/istoriya-internetu-vid-pershyh-merezh-do-suchasnosti/ (дата звернення: 15.11.2024).

6. Varadharajan V., Bansal S. Data Security and Privacy in the Internet of Things (IoT) Environment. Connectivity Frameworks for Smart Devices, 2016, P. 261 – 281.

7. Enabling privacy and security in Cloud of Things: Architecture, applications, security & privacy challenges. URL: https://www.sciencedirect.com/science/article/pii/S2210832719302819 (дата звернення: 20.11.2024).

8. Li S., Tryfonas T., Li, H. The Internet of Things: a security point of view. Internet Research, 26(2), 2016, P. 337 – 359.

9. Кулик Я.А, Волошина В.О., Олійниченко А.Б. Інтернет речей. Його концепція та ідеї для застосування. Вінницький національний технічний університет URL:https://conferences.vntu.edu.ua/index.php/all-fksa/all fksa2023/paper/download/16981/14410 (дата звернення: 30.11.2024)

10. Климаш М.М., Пиріг Ю.В., Стригалюк Б.М. Наукоємні технології в інфокомунікаціях: обробка інформації, кібербезпека, інформаційна боротьба, колективна монографія, Харків: Лідер, 2017, 600 с.

11. M. James, The Death of Competition, New York, NY, USA: Harper Collins, 1996, P. 297-300.

12. S. Leminen, M. Westerlund, M. Rajahonka and R. Siuruainen, Towards IoT ecosystems and business models Internet of Things Smart Spaces and Next Generation Networking, 2012, P. 15-26.

13. O. Mazhelis, E. Luoma and H. Warma, Defining an Internet-of-Things ecosystem, Internet of Things Smart Spaces and Next Generation Networking, 2012, P. 1-14.

14. M. Zhang, F. Sun and X. Cheng, Architecture of Internet of Things and its key technology integration based on RFID, Proc. 5th Int. Symp. Comput. Intel. Design (ISCID), vol. 1, 2012, P. 294-297.

15. S. N. Swamy, D. Jadhav and N. Kulkarni, Security threats in the application layer in IoT applications, Proc. Int. Conf. I-SMAC (IoT Social Mobile Anal. Cloud) (I-SMAC), 2017, P. 477-480.

16. K. Singh and B. Patro, Elliptic curve sign encryption based security protocol for RFID, KSII Trans. Internet Inf. Syst., vol. 2020, P. 344-365.

17. P.Y.-F. Lam, Development of a multi-domain RFID security model for global supply chains and a practical framework for model adoption, 2020, P. 260-268.

18. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari and M. Ayyash, Internet of Things: A survey on enabling technologies protocols and applications, IEEE Commun. Surveys Tuts., vol. 17, no. 4, 2015 P, 2347-2376.

19. Shalaginov, O. Semeniuta and M. Alazab, MEML: Resource-aware MQTT-based machine learning for network attacks detection on IoT edge devices, Proc. 12th IEEE/ACM Int. Conf. Utility Cloud Computing. Companion, 2019 P. 123-128.

20. E. Rescorla and T. Dierks, The transport layer security (TLS) protocol version 1.3, 2018, P. 150-160.

21. IEEE Standard for Low-Rate Wireless Networks, Standard 802.15.4-2015, 2016, P. 684.

22. ZigBee Standards Organization, Davis, CA, USA, 2012, P. 565.

ДОДАТОК А

ВІРТУАЛЬНА РОЗРОБКА ПРОЄКТІВ ІНТЕРНЕТУ РЕЧЕЙ ЗАСОБАМИ CISCO PACKET TRACER

Гуменний Д. С., здобувач ступеня вищої освіти «магістр»
Шинкарчук Н. В., кандидат технічних наук, доцент кафедри інформаційних технологій та моделювання
Рівненський державний гуманітарний університет

Інтернет речей (IoT) є однією з ключових технологій сучасного цифрового світу. Його інтеграція в різні галузі дозволяє автоматизувати процеси, підвищити ефективність систем і забезпечити новий рівень взаємодії між пристроями та користувачами. В умовах активного впровадження IoT-рішень зростає потреба у їхньому моделюванні та тестуванні ще до фактичного розгортання. Віртуальне проєктування стає важливим етапом у створенні надійних, безпечних і ефективних систем. У цьому контексті особливе значення має використання Cisco Packet Tracer – інструмента, що дозволяє віртуально моделювати IoT-інфраструктуру.

Cisco Packet Tracer – це багатофункціональне середовище для мережевого моделювання, яке окрім класичних мережевих рішень, також підтримує розробку сценаріїв Інтернет речей [1]. Цей інструмент дозволяє створювати симуляції з підключенням різноманітних пристроїв, таких як датчики температури, дверні сенсори, лампи, розумні вимикачі, вентилятори та мікроконтролери. Користувач має змогу задавати логіку взаємодії між пристроями через умовні сценарії (IF-THEN), а також використовувати базові елементи програмування подій.

У Packet Tracer реалізовано підтримку IoT-кодування за допомогою вбудованого інтерпретатора JavaScript. Це дозволяє створювати складніші сценарії поведінки пристроїв та автоматизувати їхню взаємодію на основі отриманих даних. Наприклад, при перевищенні певного порогу температури може автоматично вмикатися вентилятор або надсилатися повідомлення адміністратору. Серед інших можливостей Packet Tracer – емуляція роботи з хмарними сервісами (Cloud) і підтримка MQTT-брокера для моделювання обміну даними між пристроями.

Однією з переваг Packet Tracer є його навчальна спрямованість. Інструмент дозволяє безкоштовно практикуватися в проєктуванні Інтернет речей студентам і початківцям, надаючи широкий спектр компонентів і можливостей для тестування мережевих топологій та рішень [2].

В процесі дослідження розроблено кілька прототипів IoT-систем із використанням середовища Packet Tracer.

1. Система моніторингу температури та вологості у теплиці.

Використовуються датчики температури та вологості, які передають дані на мікроконтролер. Якщо температура перевищує 30°C, вмикається вентилятор (рис. 1). Якщо вологість падає нижче 40%, надсилається повідомлення користувачу (рис. 2).

```
if (temperatureSensor.value > 30) {
  fan.setPower(true);
}
if (humiditySensor.value < 40) {
  sendNotification("Вологість у теплиці занадто низька");
```

Рис.1 Система моніторингу температури та вологості, фрагмент коду на JavaScript

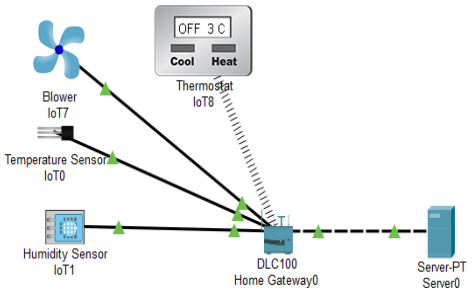

Рис.2 Топологія системи моніторингу температури та вологості у теплиці в Packet Tracer

2. Система автоматичного освітлення в «розумному будинку».

Датчики руху та освітленості регулюють вмикання світла (рис. 3). Світло вмикається лише за умови наявності руху та низького рівня освітлення (менше 20%) (рис. 4).

```
if (motionSensor.triggered && lightSensor.value < 20) {
  light.setPower(true);
} else {
  light.setPower(false);
```

Рис.3 Датчика руху та освітленості, фрагмент коду на JavaScript

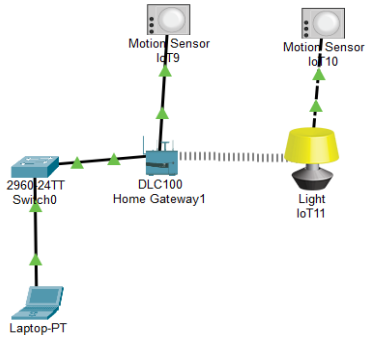

Рис.4 Топологія системи автоматичного освітлення в Packet Tracer

3. Система контролю доступу з використанням RFID.

Кожна RFID-картка має унікальний ID. Мікроконтролер порівнює з базою даних дозволених ID (рис. 5). Якщо ID авторизований – відкривається дверний замок і подія реєструється у журналі (рис. 6) [3].

```
if (rfidReader.lastCardId == "ABC123") {
  doorLock.setLocked(false);
  logEvent("Доступ дозволено: ABC123");
} else {
  logEvent("Спроба несанкціонованого доступу");
```

Рис.5 Система контролю доступу, фрагмент коду на JavaScript

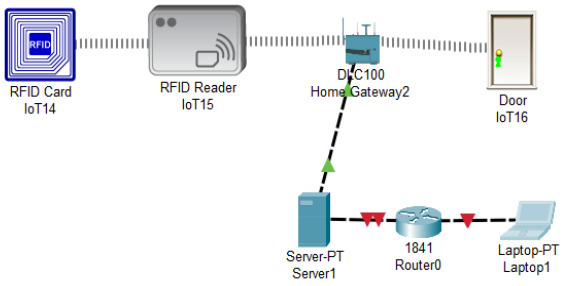

Рис.6 Топологія системи контролю доступу з використанням технології RFID в Packet Tracer

Кожен із цих прикладів демонструє ефективність підходу до віртуального проєктування, ще до реального впровадження можна оцінити функціональність системи, знайти й усунути потенційні помилки, протестувати різні варіанти взаємодії пристроїв.

Отже, віртуальне проєктування рішень Інтернету речей за допомогою Cisco Packet Tracer є важливою частиною у сучасній інженерній практиці. Cisco Packet Tracer забезпечує ефективну підготовку до реального розгортання систем Інтернет речей, дозволяє моделювати складні сценарії взаємодії між пристроями та формує основу для навчання й розвитку навичок у сфері цифрових технологій. Подальше удосконалення цього підходу сприятиме підвищенню якості проєктування IoT-рішень у різних галузях.

Список використаних джерел:

1. What is Cisco Packet Tracer? | Free Training and Download. URL: https://www.netacad.com/cisco-packet-tracer (дата звернення: 12.04.2025).

2. Packet Tracer - Wikipedia. URL: https://en.wikipedia.org/wiki/Packet_Tracer (дата звернення: 16.04.2025).

3. K. Singh, B. Patro Elliptic curve sign encryption based security protocol for RFID, KSII Trans. Internet Inf. Syst., 2020, vol. 14, no. 1, pp. 344-365.