

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Навчально-науковий
інститут електроенергетики
(інститут)
Факультет інформаційних технологій
(факультет)
Кафедра інформаційних технологій та комп'ютерної інженерії
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня бакалавра

студента Чередниченка Олексія Віталійовича
(ПІБ)

академічної групи 123-21ск-1
(шифр)

спеціальності 123 Комп'ютерна інженерія
(код і назва спеціальності)

за освітньо-професійною програмою 123 Комп'ютерна інженерія
(офіційна назва)

на тему «Комп'ютерна система ТОВ «ГРІН ФЬЮЧЕР ІНЖІНІРІНГ» з детальним опрацюванням побудови та налаштування корпоративної мережі та комплексу контролю живлення в гідропонній теплиці»
(назва за наказом ректора)

| Керівники | Прізвище, ініціали | Оцінка за шкалою | | Підпис |
|-------------------------------|--------------------|------------------|---------------|--------|
| | | рейтинговою | інституційною | |
| кваліфікаційної роботи | проф. Цвіркун Л.І. | | | |
| спеціальної частини | проф. Цвіркун Л.І. | | | |
| розділів: | | | | |
| розробка апаратної частини | доц. Бешта Д.О. | | | |
| розробка корпоративної мережі | ас. Панферова Я.В. | | | |

| | | | | |
|-----------|--|--|--|--|
| Рецензент | | | | |
|-----------|--|--|--|--|

| | | | | |
|----------------|--------------------|--|--|--|
| Нормоконтролер | проф. Цвіркун Л.І. | | | |
|----------------|--------------------|--|--|--|

Дніпро
2024

ЗАТВЕРДЖЕНО:

завідувач кафедри
інформаційних технологій
та комп'ютерної інженерії
(повна назва)

_____ Гнатушенко В.В.
(підпис) (прізвище, ініціали)
« _____ » _____ 2024 року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня бакалавр

студента Чередниченка О.В. академічної групи 123-21 ск-1
(прізвище та ініціали) (шифр)

спеціальності 123 Комп'ютерна інженерія
за освітньо-професійною програмою 123 Комп'ютерна інженерія
(офіційна назва)

на тему «Комп'ютерна система ТОВ «ГРІН ФЬЮЧЕР ІНЖИРІНГ» з детальним опрацюванням побудови та налаштування корпоративної мережі та комплексу контролю живлення в гідропонній теплиці»

затверджену наказом ректора НТУ «Дніпровська політехніка» від 29.04.2024 № 375-с

| Розділ | Зміст | Термін виконання |
|-------------------------------------|--|------------------|
| Стан питання та постановка завдання | На основі матеріалів виробничих практик, інших науково-технічних джерел розглянути призначення та завдання побудови мережі компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ» та системи комплексу контролю процесів живлення в гідропонній теплиці | 05.05.2024 |
| Розробка апаратної частини | Розробити вимоги до функцій, виконуваними корпоративною мережею та системою комплексу контролю процесів живлення в гідропонній теплиці | 12.05.2024 |
| Розробка корпоративної мереж | Побудувати в Packet Tracer модель корпоративної мережі компанії, виконати налаштування та перевірку роботи системи | 26.05.2024 |
| Розробка компонента системи | Розробити систему комплексу контролю процесів живлення в гідропонній теплиці | 09.06.2024 |

Завдання видано

_____ (підпис керівника)

_____ проф. Цвіркун Л.І.
(прізвище, ініціали)

Дата видачі 06.02.2024

Дата подання до екзаменаційної комісії 14.06.2024

Прийнято до виконання

_____ (підпис студента)

_____ Чередниченко О.В.
(прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 122 с., 47 рис., 10 табл., 2 додатка, 11 джерел.

МЕРЕЖА, ІНТЕРНЕТ РЕЧЕЙ, МАРШРУТИЗАТОРИ, КОМУТАТОРИ, CISCO, CISCO PACKET TRACER, NAT, VPN, DHCP, VLAN.

Об'єкт проектування — комп'ютерна система фірми "ГРІН ФЬЮЧЕР ІНЖИРІНГ", яка включає створення та конфігурацію корпоративної мережі, а також інтегрований комплекс для моніторингу та керування процесами живлення в гідропонічній теплиці.

Мета роботи — створення комп'ютерної системи компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ».

Було розроблено комп'ютерну мережу, яка може гнучко змінювати свій зовнішній вигляд і набір функцій шляхом перепрограмування. Ця мережа орієнтована на застосування в компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ» з метою забезпечення ефективної роботи корпоративної інфраструктури.

Для компанії також було розроблено комплекс контролю процесів живлення в гідропонічній теплиці. Цей комплекс включає можливості автономної роботи, відстеження станів датчиків та оперативного реагування на зміни середовища. Завдяки цьому система забезпечує постійний моніторинг і оптимізацію умов для вирощування рослин.

Розроблені системи надають можливість проведення як технічної, так і програмної модернізації, що дозволяє легко адаптувати їх до нових вимог та умов експлуатації. Комп'ютерна мережа була створена відповідно до завдань, поставлених для кваліфікаційної роботи бакалавра, забезпечуючи високу продуктивність та надійність у використанні.

ЗМІСТ

| | |
|--|----|
| ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК, ОДИНИЦЬ І ТЕРМІНІВ..... | 7 |
| ВСТУП | 8 |
| 1 СТАН ПИТАННЯ І ПОСТАНОВКА ЗАВДАННЯ | 9 |
| 1.1 Стисла характеристика галузі та умов застосування комп'ютерної системи, що проектується. | 9 |
| 1.2 Характеристика і структура компанії «ГРІН ФЬЮЧЕР ІНЖИНІРІНГ» | 10 |
| 1.3 Принципи, технічні способи та математичні методи інформаційного забезпечення підприємства | 10 |
| 1.4 Огляд існуючих інженерних рішень КС в галузі та визначення можливих напрямків рішення поставлених завдань | 11 |
| 1.5 Розробка схеми організаційної структури підприємства..... | 14 |
| 1.6 Завдання і мета роботи | 16 |
| 1.7 Визначення можливих напрямків рішення поставлених завдань..... | 17 |
| 1.8 Обґрунтування вибраного напрямку інженерного рішення..... | 19 |
| 2 РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ КОМП'ЮТЕРНОЇ АБО КІБЕРФІЗИЧНОЇ СИСТЕМИ | 21 |
| 2.1 Технічні вимоги до комп'ютерної системи компанії та кіберфізичної системи | 21 |
| 2.1.1 Вимоги до систем в цілому..... | 21 |
| 2.1.1.1 Вимоги до структури і функціонуванню системи..... | 21 |
| 2.1.1.1.1 Перелік підсистем, їхнє призначення й основні характеристики, вимоги до числа рівнів ієрархії та ступені централізації Системи | 21 |
| 2.1.1.1.2 Вимоги до способів і засобів зв'язку між компонентами систем..... | 23 |
| 2.1.1.1.3 Вимоги до характеристик взаємозв'язків створюваної системи із суміжними системами..... | 24 |

| | | |
|-----------|---|----|
| 2.1.1.1.4 | Вимоги до режимів функціонування систем | 24 |
| 2.1.1.1.5 | Вимоги до діагностування системи | 24 |
| 2.1.1.1.6 | Перспективи розвитку, модернізації системи..... | 25 |
| 2.1.1.2 | Вимоги до показників призначення..... | 26 |
| 2.1.1.3 | Вимоги до патентної чистоти | 26 |
| 2.1.1.4 | Додаткові вимоги | 27 |
| 2.1.2 | Вимоги функцій, виконуваним системою..... | 29 |
| 2.1.3 | Вимоги до видів забезпечення комп'ютерної системи | 31 |
| 2.1.3.1 | Вимоги до інформаційного забезпечення | 31 |
| 2.1.3.2 | Вимоги до лінгвістичного забезпечення | 33 |
| 2.1.3.3 | Вимоги до технічного забезпечення | 35 |
| 2.1.3.4 | Вимоги до організаційного забезпечення | 36 |
| 2.1.3.5 | Вимоги до методичного забезпечення..... | 38 |
| 2.1.3.6 | Вимоги до задач (налаштувань), які виконує КС | 39 |
| 2.2 | Розробка апаратної частини комп'ютерної системи | 40 |
| 2.2.1 | Розробка загальної архітектури мережі підприємства..... | 40 |
| 2.2.2 | Вибір і обґрунтування структурної схеми комплексу технічних засобів комп'ютерної системи..... | 42 |
| 2.2.3 | Розробка специфікації апаратних засобів комп'ютерної системи.. | 43 |
| 2.2.4 | Розрахунок інтенсивності трафіку вихідного трафіку найбільшої локальної мережі підприємства | 48 |
| 3 | РОЗРОБКА КОРПОРАТИВНОЇ МЕРЕЖІ | 50 |
| 3.1 | Проектування логічної топології мережі..... | 50 |
| 3.2 | Вибір та опис мережного обладнання..... | 52 |
| 3.3 | Розрахунок схеми адресації корпоративної мережі | 52 |
| 3.4 | Базове налаштування конфігурації пристроїв..... | 57 |
| 3.5 | Налаштування DHCP | 59 |
| 3.6 | Налаштування маршрутизації..... | 59 |
| 3.7 | Налаштування доступу в Інтернет | 61 |
| 3.8 | Налаштування мереж VLAN, маршрутизації між VLAN | 62 |

| | |
|---|-----|
| 3.9 Захист інформації в комп'ютерній системі від несанкціонованого доступу | 66 |
| 3.9.1 Налаштування маршрутизаторів на підтримку служби AAA | 66 |
| 3.9.2 Налаштування віртуальної приватної мережі VPN | 67 |
| 3.10 Перевірка компонентів корпоративної мережі компанії | 68 |
| 4 РОЗРОБКА КОМПОНЕНТА СИСТЕМИ..... | 78 |
| 4.1 Інженерне рішення розробки гідропонної системи теплиці..... | 78 |
| 4.2 Застосовані рішення, програмне забезпечення та інженерні рішення .. | 78 |
| 4.3 Розробка компонентів системи..... | 80 |
| 4.3.1 Вибір мікроконтролера, датчиків та пристроїв для системи..... | 80 |
| 4.3.2 Розробка компонента керуванням станів наповнення резервуарів | 82 |
| 4.3.3 Розробка компонента підтримки живильного розчину та його процесу циркуляції | 85 |
| 4.3.4 Розробка компонента сповіщення користувача та збереження станів | 88 |
| 4.3.5 Розробка допоміжних компонентів..... | 91 |
| 4.4 Тестування системи | 92 |
| ВИСНОВКИ..... | 96 |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ | 97 |
| ДОДАТОК А..... | 99 |
| ДОДАТОК Б | 102 |

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК, ОДИНИЦЬ І ТЕРМІНІВ

ПК – персональний комп'ютер
СБД – система баз даних
UPS – джерело безперебійного живлення
AI – штучний інтелект
API – програмний інтерфейс додатків
DNS – система доменних імен
HTTP – протокол передачі гіпертексту
IoT – Інтернет речей
IP – Інтернет-протокол
ISP – Інтернет-провайдер
LAN – локальна обчислювальна мережа
SQL – мова структурованих запитів
UML – уніфікована мова моделювання
VLAN – віртуальна локальна мережа
VPN – віртуальна приватна мережа
WAN – глобальна обчислювальна мережа
°C – градус Цельсія
% – відсоток
Ω – ом (одиниця електричного опору)
mA – міліампер
ГГц (GHz) – гігагерц
ГБ (GB) – гігабайт
МГц (MHz) – мегагерц
мс (ms) – мілісекунда
Вт (W) – ват
м² (m²) – квадратний метр

ВСТУП

В сучасному світі комп'ютерні системи стають необхідним елементом для ефективного управління та контролю в різних галузях промисловості. У цьому контексті компанія «ГРІН ФЬЮЧЕР ІНЖИРІНГ» вирішила використати передові комп'ютерні технології для оптимізації та підвищення продуктивності власного виробництва. Одним з ключових напрямків розвитку побудова комплексу контролю процесів живлення в гідропонній теплиці.

Цей амбітний проект передбачає впровадження інноваційних рішень для автоматизації та оптимізації управління різноманітними аспектами виробництва в гідропонній теплиці. Від побудови мережевої інфраструктури до розробки програмного забезпечення для контролю параметрів середовища, всі етапи цього проекту спрямовані на покращення ефективності та якості виробництва.

У даній кваліфікаційній роботі ретельно досліджено процес побудови та налаштування корпоративної мережі для потреб компанії, а також розроблено комплекс системи контролю для гідропонних теплиць. Важливим аспектом є забезпечення надійності та безпеки системи, що вимагає використання передових технологій та врахування найкращих практик у галузі інформаційної безпеки.

Робота враховує можливість масштабування та розвитку системи у майбутньому, з огляду на швидкі зміни в технологічному середовищі та потреби компанії. В цілому, дана робота є важливим кроком у напрямку оптимізації виробничих процесів та підвищення конкурентоспроможності компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ» на ринку.

1 СТАН ПИТАННЯ І ПОСТАНОВКА ЗАВДАННЯ

1.1 Стисла характеристика галузі та умов застосування комп'ютерної системи, що проектується.

Галузь застосування комп'ютерної системи, яка проектується – агронома та конкретно, автоматизація вирощування культур. Система підприємства призначена для використання у вирощуванні рослин у контрольованому середовищі, що забезпечує оптимальні умови для росту та розвитку рослин та збору інформації о поточних станах. Вона включає в себе автономний контроль за поливом, вентиляцією, освітленням та іншими аспектами середовища, необхідними для здоров'я та зростання рослин. Але всі ці функції включає у себе автономні IoT системи.

Дана галузь охоплює різні ключові напрямки.

Оптимізація середовища росту рослин. Включає в себе розробку систем контролю вологості, температури, освітлення та інших параметрів, що забезпечують оптимальні умови для зростання рослин. Це може включати в себе використання технологій гідропоніки, аеропоніки та інших методів безґрунтового вирощування.

Дослідження та впровадження нових технологій. Важливо вивчати та впроваджувати передові технології у галузі сільського господарства, такі як інтернет речей (IoT), штучний інтелект, дрони та інші, для покращення ефективності та продуктивності теплиць.

Генетичне вдосконалення рослин. Включає в себе роботу над створенням нових сортів рослин, які були б стійкими до шкідників, хвороб або негативних погодних умов, а також мали б покращені смакові та якісні характеристики.

Створення екологічно чистих систем вирощування. Важливо розробляти та впроваджувати екологічно чисті методи вирощування рослин, які б не лише забезпечували високу врожайність, але й мінімізували вплив на навколишнє середовище [1].

Створення ефективних систем управління та моніторингу. Включає в себе розробку систем управління та моніторингу, які дозволяють оперативно реагувати на зміни в умовах вирощування рослин, а також вести аналітику для оптимізації процесів [1].

1.2 Характеристика і структура компанії «ГРІН ФЬЮЧЕР ІНЖИНІРІНГ»

Об'єкт впровадження – компанія «ГРІН ФЬЮЧЕР ІНЖИРІНГ».

Компанія «ГРІН ФЬЮЧЕР» – інноваційний лідер у сфері сільського господарства, спеціалізується на розробці та впровадженні автоматизованих систем вирощування рослин у контрольованих середовищах. Завдяки поєднанню передових технологій та професійного підходу, вони забезпечують клієнтам надійність, ефективність та стабільність у вирощуванні рослин.

Компанія спеціалізується на розробці, виробництві та комерціалізації різноманітних продуктів для агросектору, включаючи системи автоматичного поливу, спеціалізовані LED фіто-світильники, стелажні гідропонні системи, а також компоненти, що сприяють прискореному росту, розвитку та дозріванню рослин, включно з живцями та розсадою. Виробничі можливості компанії охоплюють повний цикл від розробки і дослідження до виробництва та постачання готових "під ключ" рішень у сфері фіто-освітлення та гідропоніки [2].

1.3 Принципи, технічні способи та математичні методи інформаційного забезпечення підприємства

В аграрній сфері існує безліч технічних засобів та методів інформаційного забезпечення, які розвиваються.

Автоматизація процесів вирощування. Застосування автоматичних систем поливу, дозування поживних розчинів, регулювання вологості та освітлення дозволяє оптимізувати процес вирощування рослин та забезпечити стабільність умов у гідропонних системах.

Використання датчиків та сенсорів. Встановлення датчиків температури, вологості, рівня рН та інших параметрів дозволяє здійснювати постійний моніторинг умов вирощування та автоматично реагувати на зміни.

Системи автоматичного керування: Використання мікроконтролерів або комп'ютерних систем для програмного керування процесами в гідропонній системі, що базується на даних від датчиків та сенсорів.

Системи моніторингу та аналізу даних. Впровадження систем збору, аналізу та візуалізації даних, що дозволяють оперативно контролювати стан системи, виявляти аномалії та приймати відповідні заходи.

Системи віддаленого керування. Реалізація можливості керування гідропонними системами віддалено через Інтернет або мобільний додаток, що забезпечує зручність та доступність для операторів.

Безпека та захист даних. Застосування методів шифрування, автентифікації та інших заходів забезпечення безпеки даних у гідропонних системах, що має велике значення для захисту від несанкціонованого доступу та втрати даних [2].

1.4 Огляд існуючих інженерних рішень КС в галузі та визначення можливих напрямків рішення поставлених завдань

Важливо враховувати, що ефективна обробка та передача інформації в гідропонних системах потребує використання надійних і масштабованих технологій. Сучасні IoT рішення дозволяють зібрати дані з різних сенсорів в реальному часі та забезпечити їхню швидку обробку і передачу для подальшого аналізу і управління.

Системи, які використовуються для таких завдань, зазвичай включають:
– протоколи передачі даних, такі як MQTT або CoAP, які спеціалізуються на мінімізації навантаження на мережу і підтримці високої надійності в IoT-додатках;

– хмарні платформи, такі як AWS IoT Core або Microsoft Azure IoT Hub, які забезпечують потужні інструменти для обробки даних, їх зберігання та аналізу;

– Edge Computing, що дозволяє проводити первинну обробку даних безпосередньо на пристроях, скорочуючи час реакції та зменшуючи обсяг переданих через мережу даних.

Побудова об'єкта впровадження включає архітектурні рішення, які забезпечують гнучкість, масштабованість та інтеграційну взаємодію системи. Принципи включають:

– модульність, яка дозволяє легко розширювати або модифікувати систему;

– мтандартизація компонентів для забезпечення їх заміності та сумісності;

– автоматизація процесів для підвищення ефективності та зниження людського фактору.

Відомі рішення у цій галузі включають технології та платформи, які демонструють успішну імплементацію аналогічних систем у схожих умовах.

Приклади включають:

– контрольні системи від компаній, як Hoogendoorn і Priva, які пропонують інтегровані рішення для автоматизації та управління в сільськогосподарських теплицях;

– розумні агротехнології, такі як ті, що розробляються компаніями AgriTech, які фокусуються на оптимізації виробництва за допомогою автоматизованого контролю клімату та живлення рослин.

Огляд важливих існуючих способів обробки та передачі інформації для корпоративної мережі ТОВ "ГРІН ФЬЮЧЕР ІНЖИНІРИНГ":

В обробці та передачі інформації ключовими є технології, що забезпечують швидкість, безпеку та стабільність мережевих операцій:

– Ethernet та Wi-Fi основні технології для забезпечення фізичного зв'язку у корпоративних мережах. Ethernet використовується для створення

провідних мереж з високою надійністю та пропускнуою здатністю, тоді як Wi-Fi надає гнучкість бездротового доступу;

- MPLS (Multiprotocol Label Switching) технологія, що використовується для підвищення ефективності передачі даних через створення віртуальних шляхів між джерелом та приймачем;

- VPN (Virtual Private Network) забезпечує безпечний зв'язок між користувачами та корпоративними ресурсами через інтернет, шифруючи всі передані дані.

Принципи побудови корпоративних мереж включають:

- скалярність та гнучкість: мережа повинна бути спроектована так, щоб легко адаптуватися до змін у кількості користувачів або вимогах до ресурсів без значних перебудов;

- резервування та відновлення: забезпечення високої доступності та мінімального часу простою шляхом імплементації резервних зв'язків та систем відновлення;

- безпека: впровадження сучасних заходів безпеки, таких як файєрволи, інтрузійні детекційні системи (IDS), і шифрування даних для захисту від зовнішніх та внутрішніх загроз.

На ринку існує ряд перевірених рішень та продуктів, що використовуються для створення ефективних корпоративних мереж:

- Cisco Systems один з лідерів у галузі мережевого обладнання, пропонує рішення від комутаторів та маршрутизаторів до комплексних безпекових систем.

- Juniper Networks відома своїми високопродуктивними мережевими рішеннями, особливо в області маршрутизації та безпеки.

- HP Enterprise постачає широкий спектр мережевих рішень, включаючи комутатори, маршрутизатори та технології бездротового зв'язку.

1.5 Розробка схеми організаційної структури підприємства

В організаційній структурі підприємства є наступні відділи: департамент досліджень та розробок, виробничий, відділ продажів та маркетингу, фінансовий відділ, відділ кадрів та адміністрування.

Департамент досліджень та розробок відповідає за створення та вдосконалення технологій і продуктів компанії. Цей підрозділ займається інженерними розробками, дослідженнями нових методів вирощування рослин, тестуванням і впровадженням інновацій.

Виробничий відділ відповідає за виробництво та збірку автоматизованих систем, що виготовляє компанія. Цей підрозділ включає в себе робітників на виробництві, інженерів-технологів, які контролюють якість виробів.

Відділ продажів та маркетингу відповідає за просування продуктів компанії на ринку, залучення клієнтів, укладання угод і ведення комунікації з клієнтами. Включає в себе менеджерів з продажу, маркетологів, а також спеціалістів з реклами та PR.

Фінансовий відділ відповідає за фінансове планування, облік і контроль фінансових операцій компанії. Цей підрозділ включає в себе бухгалтерів, фінансових аналітиків та фінансових менеджерів.

Відділ кадрів та адміністрування відповідає за управління кадрами, розвиток персоналу, а також за загальне адміністрування офісу. Включає в себе менеджерів з кадрів, HR-спеціалістів, адміністраторів та офіс-менеджерів [2].

На рисунку 1.1 наведено організаційна структура компанії «Green Future».

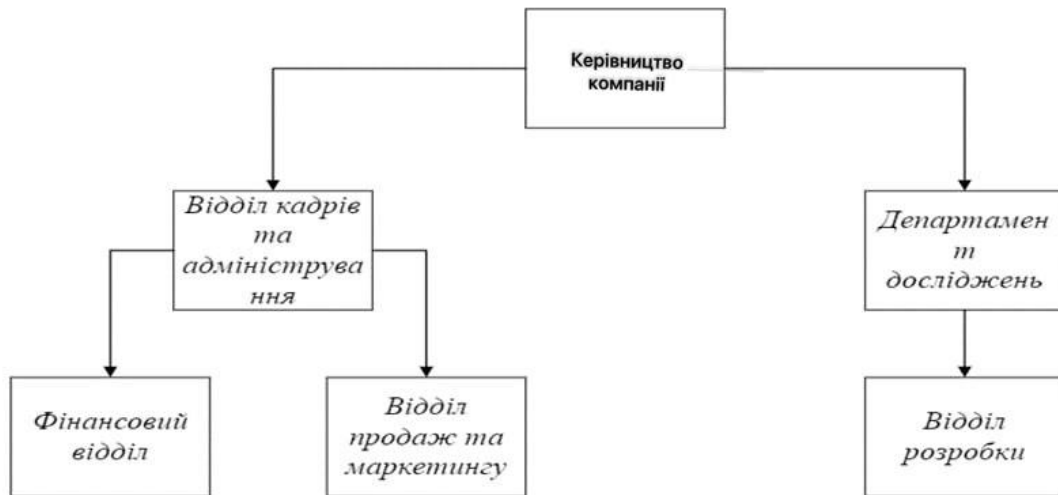


Схема організаційної структури компанії «Green Future»

Топографічне розміщення структурних підрозділів скається з однієї будівлі. Це офіс на 3-му поверсі у не житловій будівлі, має 5 кімнат. Знаходиться будівля за адресою: просп. Лесі Українки, 21, Дніпро, Дніпропетровська область, 49000 [1].

Топографічна схема розміщення структурних підрозділів показана на рисунку 1.2

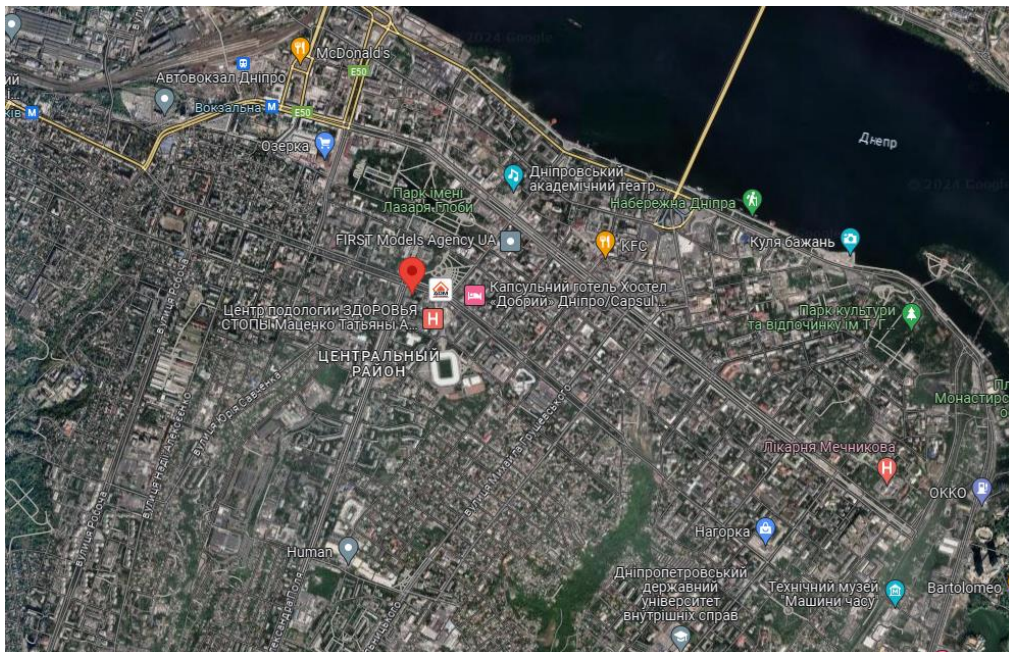
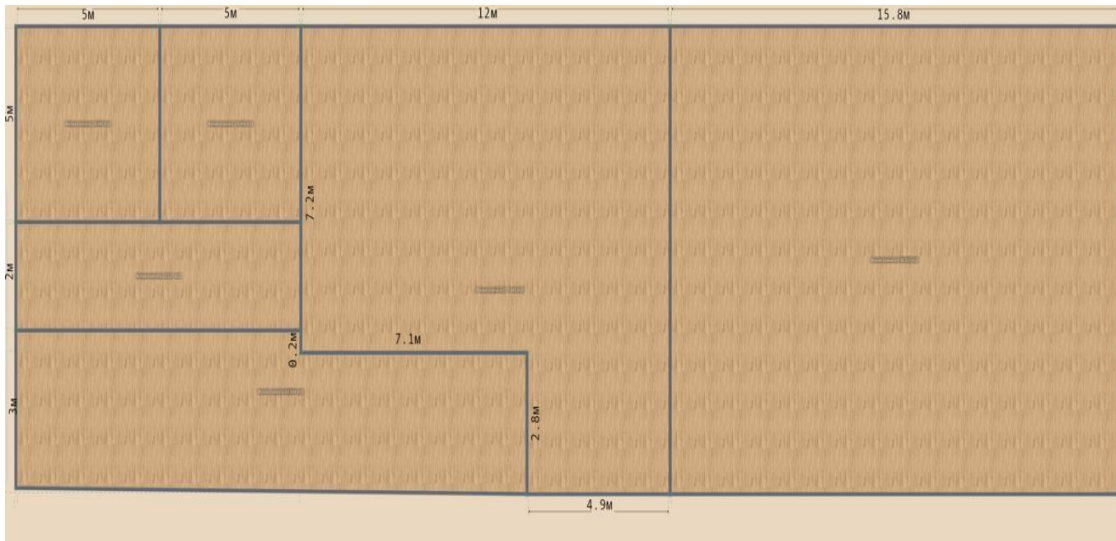


Схема розташування структурних підрозділів компанії «Green Future»

Структурна схема розміщення підрозділів у будівлі в якій знаходиться: Фінансовий відділ, відділ продажу та маркетингу, відділ адміністрування та

кадрів, відділ дослідження та розробки та зона тестування й досліджень (рисунок 1.3).



Структурна схема розміщення підрозділів у будівлі

1.6 Завдання і мета роботи

Метою роботи є розробка Комп'ютерної системи для ТОВ «ГРІН ФЬЮЧЕР ІНЖИНІРИНГ» з детальним опрацюванням побудови та налаштування корпоративної мережі та комплексу контролю процесів живлення в гідропонній теплиці [1].

Для реалізації задуманого проекту необхідно пройти кілька ключових етапів, включаючи:

- детальний аналіз об'єкта, де буде впроваджено систему;
- формулювання технічних вимог до об'єкта;
- обґрунтування вибору мережевої структури та обладнання для запланованих систем;
- створення специфікації апаратних засобів;
- аналіз мережевого трафіку найбільш навантаженої підмережі;
- розробка моделі комп'ютерної мережі та налаштування мережевого обладнання.

- розробити комплекс контролю процесів живлення в гідропонній теплиці;
- провести аналіз показників компонентів в гідропонній теплиці для виявлення кращих параметрів живильного розчину;
- розробити функцію відправки даних до локального серверу мережі;
- провести тестування систем в цілому та кожного компонента окремо.

Результатом роботи має бути корпоративна мережа, яка характеризується масштабованістю, надійністю та безпекою. Масштабованість забезпечує легке розширення мережі при збільшенні кількості користувачів або підключених пристроїв, що дозволяє компанії зростати без значних додаткових витрат. Надійність гарантується через високу ступінь відмовостійкості та забезпечення безперебійного доступу до ресурсів завдяки використанню резервування, відмовостійких протоколів та надійного обладнання. Безпека мережі забезпечується за рахунок сучасних засобів захисту від несанкціонованого доступу та кібератак, що є критично важливим для збереження конфіденційної інформації та безперебійної роботи компанії.

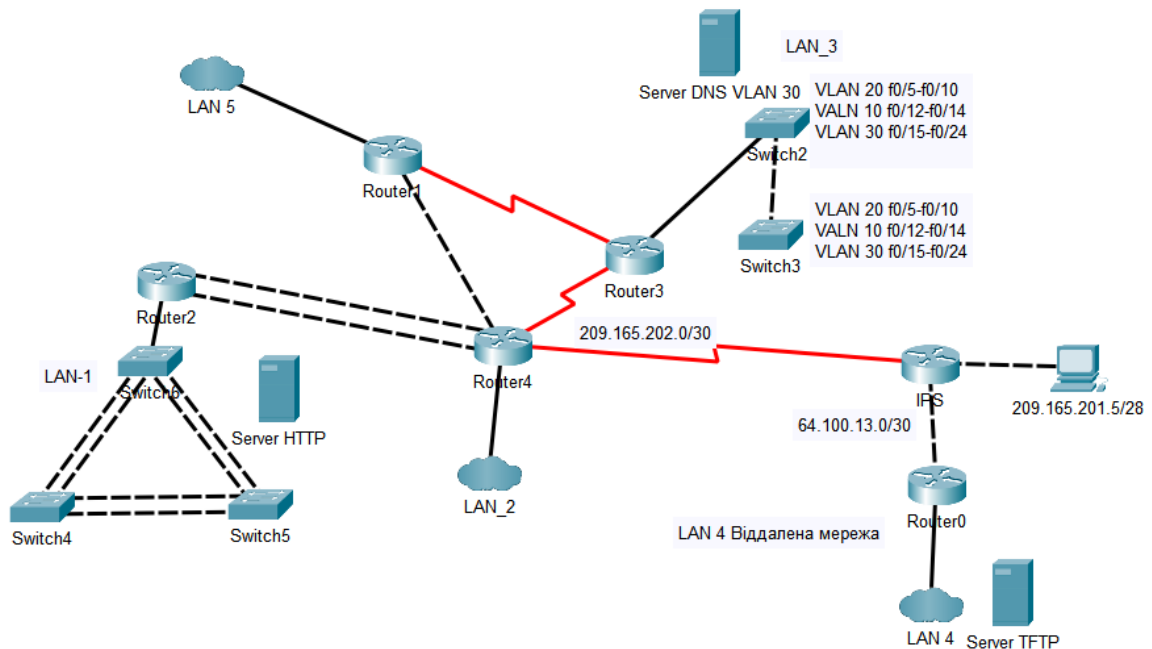
Крім того, розроблено кіберфізичну систему контролю процесів живлення в гідропонній теплиці. Ця система включає можливості автономної роботи, що дозволяє автоматично відстежувати стани датчиків та оперативно реагувати на зміни середовища. Завдяки цьому забезпечується оптимізація умов для вирощування рослин, що сприяє підвищенню ефективності гідропонного вирощування. Система також надає можливість проведення технічної та програмної модернізації, що дозволяє легко адаптувати її до нових вимог та умов експлуатації.

1.7 Визначення можливих напрямків рішення поставлених завдань

Розробка архітектури мережі:

- мережа повинна відповідати топології на (рисунок 1.4);

- проектування гнучкої і масштабованої мережевої архітектури, що може ефективно обробляти збільшення обсягу даних і кількості мережевих ресурсів;
- впровадження сучасних стандартів Ethernet та Wi-Fi для забезпечення швидкості та надійності передачі даних;
- використання віртуальних приватних мереж (VPN) для безпечного з'єднання з віддаленими ресурсами;
- інтеграція технологій для підвищення ефективності маршрутизації трафіку в корпоративній мережі.



Топологія мережі

Вибір та інтеграція обладнання:

- обрання мережевих комутаторів, маршрутизаторів, які підтримують необхідну пропускну спроможність і безпеку;
- розгляд опцій мережевого зберігання даних для ефективного архівування і швидкого доступу до інформації;
- впровадження резервних систем для забезпечення високої доступності мережевих ресурсів.

Безпека мережі:

- розробка комплексної стратегії безпеки, яка включає файєрволи, антивірусне програмне забезпечення, та інші заходи кібербезпеки;
- імплементація систем детекції та запобігання вторгненням для моніторингу та реагування на можливі загрози в реальному часі;
- використання шифрування даних для захисту чутливої інформації, передаваної через мережу.

Розробка та інтеграція системи контролю процесів живлення:

- інтеграція IoT рішень для збору даних з сенсорів і автоматичного регулювання умов вирощування;
- розробка програмного забезпечення для управління та аналізу зібраних даних, що дозволяє оптимізувати використання ресурсів і підвищити продуктивність.

Кожен із цих напрямків вимагає детального планування та реалізації, враховуючи специфічні потреби та умови компанії "ГРІН ФЬЮЧЕР ІНЖИНІРИНГ", щоб забезпечити успішне впровадження та ефективну експлуатацію розробленої системи.

1.8 Обґрунтування вибраного напрямку інженерного рішення

Вибір напрямку інженерного рішення для розробки та впровадження комп'ютерної системи компанії «ГРІН ФЬЮЧЕР ІНЖИРИНГ» ґрунтується на ретельному аналізі вимог до функціональності, надійності та ефективності системи, а також врахуванні специфіки діяльності компанії та умов експлуатації системи.

Перш за все, були визначені ключові вимоги до системи, які включають високу продуктивність, надійність, безпеку та масштабованість. Система повинна забезпечувати безперебійну роботу гідропонної теплиці, яка потребує постійного моніторингу та управління численними параметрами навколишнього середовища, такими як температура, вологість, освітленість та інші.

Компанія «ГРІН ФЬЮЧЕР ІНЖИРІНГ» спеціалізується на розробці інноваційних рішень для сільського господарства, що вимагає використання передових технологій і високого рівня автоматизації. Це обумовлює необхідність інтеграції кіберфізичних систем з комп'ютерними мережами, забезпечення зручного доступу до даних та можливість оперативного реагування на зміну умов.

Виходячи з аналізу вимог та специфіки діяльності компанії, було прийнято рішення використовувати мікроконтролери типу Arduino та Raspberry Pi Pico для збору даних та управління процесами у теплиці. Ці платформи дозволяють легко інтегрувати різноманітні датчики та виконавчі механізми, забезпечуючи гнучкість і можливість швидкої адаптації системи до нових вимог.

Для забезпечення ефективного управління та обміну даними між різними компонентами системи було обрано архітектуру, яка включає три рівні: ядро, доступ та хости. Це дозволяє забезпечити високу пропускну здатність, надійність та легкість у масштабуванні мережі. Використання маршрутизаторів і комутаторів з підтримкою VLAN та резервування забезпечує стабільність і безпеку мережевих з'єднань.

Особливу увагу було приділено вибору енергоефективних компонентів, що є важливим аспектом для зниження операційних витрат та забезпечення екологічної стійкості. Використання сучасних енергоефективних мікроконтролерів та мережевого обладнання дозволяє досягти оптимального балансу між продуктивністю та енергоспоживанням.

Обране інженерне рішення базується на інтеграції сучасних технологій у сфері комп'ютерної інженерії та кіберфізичних систем, що дозволяє створити ефективну, надійну та масштабовану систему для управління гідропонною теплицею. Такий підхід забезпечує високий рівень автоматизації, безпеку даних та можливість подальшого розвитку і модернізації системи відповідно до потреб компанії [1].

2 РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ КОМП'ЮТЕРНОЇ АБО КІБЕРФІЗИЧНОЇ СИСТЕМИ

2.1 Технічні вимоги до комп'ютерної системи компанії та кіберфізичної системи

2.1.1 Вимоги до систем в цілому

2.1.1.1 Вимоги до структури і функціонуванню системи

2.1.1.1.1 Перелік підсистем, їхнє призначення й основні характеристики, вимоги до числа рівнів ієрархії та ступені централізації Системи

Комп'ютерна система розроблена та призначена для забезпечення обміну інформацією з управлінням компанії «ГРІН ФЬЮЧЕР ІНЖИНІРИНГ». Системи мають працювати цілодобово для забезпечення функціоналу компанії для можливості проводити дослідження за допомогою кіберфізичної системи.

Для цього проекту потрібно створити п'ять локальних мереж. Рекомендується використовувати Ір-блок-адресу для призначення підмережі.

Корпоративна мережа має складатися з 5 підмереж:

- фінансовий відділ;
- відділ продажу та маркетингу;
- відділ адміністрування та кадрів;
- відділ дослідження та розробки;
- віддалена мережа.

Корпоративна мережа компанії поділяється на локальні мережі(системи) згідно до загальної архітектури.

Мережа поділяється на п'ять підмереж щоб забезпечити переваги, такі як:

- зручне управління: мережевим адміністраторам повинно бути забезпечено зручне керування мережею.;

- продуктивність: необхідно забезпечити високу швидкість передачі даних в локальних мережах;
- масштабованість: система повинна легко піддається заміні та додаванню пристроїв у локальній мережі;
- безпека: необхідно здійснити переважний захист мережі за допомогою точної системи безпеки, такої як брандмауери або VPN.

Потрібно розбити IP-адресу 172.24.192.0/21 на 5 підмереж враховуючи можливість розширення кількості вузлів дані замовником на подальше розширення компанії та забезпечити налаштування та взаємодіями з локальними серверами компанії у мережі для LAN з такими кількостями вузлів:

- підмережа LAN_1 – повинна забезпечувати 54 вузлів та до неї входить HTTP сервер для надання веб-сторінок;
- підмережа LAN_2 – повинна забезпечувати 68 вузлів;
- підмережа LAN_3 – повинна забезпечувати 179 вузлів та до неї входить DNS сервер для конвертації(перекладу) доменних імен в IP-адреси;
- підмережа LAN_4 – віддалена мережа яка повинна забезпечувати 119 вузлів;
- підмережа LAN_5 – повинна забезпечувати 53 вузлів.

Загалом підсистема офісу повинна налічувати 354 вузла, а віддалена мережа 119 вузлів [2].

Наступим етапом після мережі треба розробити кіберфізичну систему поливу та забезпечення поживних речовин для гідропонної теплиці. Система має забезпечувати роботу перистальтичних насосів для додавання поживних речовин у живильне середовище яке згодом має перекачуватись за допомогою занурюваної помпи з баку до рослин.

Система має забезпечувати відстеження таких показників як:

- температура живильного розчину;
- показник кислотності;

- показник електропровідності;
- кількості рідин у баках.

2.1.1.1.2 Вимоги до способів і засобів зв'язку між компонентами систем

Комунікаційні методи та засоби, які використовуються для обміну інформацією між компонентами системи, повинні відповідати критеріям, заданим технічним завданням. В мережі LAN_1 необхідно впровадити технологію EtherChannel, що дозволяє об'єднувати кілька фізичних з'єднань між комутаторами в єдине логічне з'єднання, збільшуючи тим самим пропускну здатність та надійність мережі.

Для LAN_3 рекомендовано використання технології VLAN, яка дозволяє логічно розділити одну фізичну мережу на декілька віртуальних мереж. З іншого боку, LAN_4 має здійснювати взаємодію з головною мережею через VPN для забезпечення безпечного з'єднання.

Всі мережі повинні використовувати протокол динамічної маршрутизації OSPF, що сприяє ефективному виявленню та визначенню маршрутів між різними підмережами. Підключення до Інтернету слід реалізувати через провайдера за допомогою технології NAT. Для внутрішнього з'єднання основної та віддаленої мереж варто використовувати виту пару, а для з'єднання між мережами — оптичне волокно.

Інформація з датчиків кіберфізичної системи до керуючого мікроконтролера має передається за допомогою шлейфів використовуючи електричні сигнали, той в свою чергу має передавати інформацію на сервер за допомогою бездротових технологій для подальшої обробки [2].

2.1.1.1.3 Вимоги до характеристик взаємозв'язків створюваної системи із суміжними системами

Обмін інформацією між вузлами відбувається технологією передачі даних Ethernet та протоколом OSPF. Для обміну між вузлами кіберфізичних систем виступаю налаштований IoT сервер та канали передачі даних по типу Ethernet або Wi-Fi.

2.1.1.1.4 Вимоги до режимів функціонування систем

Неперервність роботи: система повинна забезпечувати неперервний доступ до основних функцій таких як збереження даних з систем та керуванням кіберфізичними системами навіть у випадку виникнення тимчасових збоїв або відмов обладнання;

Ефективність відновлення: в разі виникнення неполадок система повинна мати можливість швидко і ефективно відновлюватися до нормального режиму роботи без значного впливу на продуктивність;

Гнучкість: система має бути гнучкою і здатною до додавання нових компонентів та модернізації обладнання;

Забезпечення безпеки: режими функціонування системи повинні забезпечувати високий рівень захисту від несанкціонованого доступу;

Масштабованість: система повинна бути здатною масштабуватися для врахування зростаючих потреб користувачів і обсягів даних без втрати продуктивності.

2.1.1.1.5 Вимоги до діагностування системи

Персонал повинен мати здатність ручно виявляти можливі проблеми та помилки в роботі системи шляхом аналізу доступної інформації та спостереження за роботою [2].

Працівники повинні періодично проводити моніторинг стану компонентів, ресурсів та процесів в системі за допомогою аналізу даних з

датчиків та наріканням працівників що до роботи мережі для вчасного виявлення проблем.

2.1.1.1.6 Перспективи розвитку, модернізації системи

Вимоги до комп'ютерної системи

Масштабованість:

– система повинна забезпечувати можливість масштабування за допомогою використання мережевих архітектур, використання віртуалізації та використання хмарних послуг для врахування зростаючих потреб компанії;

– впровадження новітніх технологій захисту інформації, обліку та протоколів передачі даних має бути здійснено з урахуванням потреби в адаптивності інфраструктури.

Інтеграція нових технологій:

– система повинна підтримувати впровадження технологій штучного інтелекту, машинного навчання та Інтернету речей (IoT).

Гнучкість та надійність:

– система повинна бути гнучкою, надійною та здатною до інтеграції з новітніми технологіями;

– забезпечення конкурентоспроможності компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ» на ринку має бути ключовою метою розвитку та модернізації системи.

Вимоги до кіберфізичної системи.

Модульність:

– кіберфізична система гідропонної теплиці повинна бути модульною, що забезпечить швидку заміну компонентів у разі їх пошкодження або модернізації.

– модульна структура має дозволяти додавання нових функціональних блоків без значних змін у загальній архітектурі системи.

Інтеграція нових технологій:

– використання IoT-датчиків для моніторингу стану рослин та умов навколишнього середовища має забезпечити своєчасне виявлення проблем та прийняття оптимальних рішень для їх усунення.

Програмне забезпечення:

– програмне забезпечення системи повинно бути розроблене на базі гнучкої та запитливої мови програмування, такої як `microPython`.

– використання `microPython` має забезпечити високу ефективність розробки, підтримки програмного забезпечення та спрощення процесу інтеграції нових функцій і технологій.

2.1.1.2 Вимоги до показників призначення

Система має забезпечити ефективне спілкування між співробітниками через текстові повідомлення, відеодзвінки та інші комунікаційні канали. Вона також повинна дозволяти працівникам доступ до інформації з кіберфізичних систем та обмін даними всередині організації.

Система має включати функції для централізованого зберігання даних на серверах з можливістю їх відновлення через резервне копіювання в хмарі. Вона повинна гарантувати захист даних від несанкціонованого доступу та шкідливого програмного забезпечення за допомогою різних захисних механізмів.

Крім того, система має забезпечувати окремим пристроям можливість підключення до Інтернету та підтримувати віддалене з'єднання співробітників через VPN для безпечного доступу до корпоративних ресурсів з будь-якого місця.

2.1.1.3 Вимоги до патентної чистоти

Обладнання та програмне забезпечення, що використовуються у комп'ютерній системі, мають відповідати встановленим патентним стандартам. Це передбачає застосування ліцензованого програмного забезпечення, детальний аналіз компонентів системи на предмет наявності

патентів, а також співпрацю з юридичними консультантами. Важливо також забезпечити належне документування процесу проектування компонентів системи, щоб у разі виникнення юридичних спорів мати обґрунтовану доказову базу про дотримання патентної чистоти.

Крім того, необхідно проводити постійний моніторинг змін у патентному законодавстві та базах патентів для своєчасного виявлення можливих ризиків. Це включає регулярні перевірки нових патентних заявок, аналіз патентних спорів та відстеження діяльності конкурентів у цій сфері. Такий підхід дозволить мінімізувати ризики пов'язані з патентними порушеннями та уникнути фінансових та правових наслідків.

Не менш важливою є співпраця з розробниками та постачальниками обладнання і програмного забезпечення для забезпечення патентної чистоти. Всі контракти з постачальниками повинні містити положення щодо дотримання патентних прав та захисту від можливих претензій третіх сторін. Це допоможе не тільки захистити компанію від потенційних патентних суперечок, але й забезпечити безперебійне функціонування комп'ютерної системи та її компонентів у довгостроковій перспективі.

2.1.1.4 Додаткові вимоги

Система повинна бути здатна працювати у широкому діапазоні температур та вологості, що характерні для умов роботи в гідропонній теплиці. Обладнання повинно бути захищеним від пилу, вологи та механічних пошкоджень. Крім того, необхідно враховувати можливість впливу електромагнітних перешкод та забезпечити належний рівень захисту від них.

1) Вимоги до активного обладнання.

Активне обладнання повинно забезпечувати стабільну та безперебійну роботу системи цілодобово. Маршрутизатори та комутатори мають підтримувати необхідні протоколи та стандарти для забезпечення ефективного обміну даними.

Маршрутизатори повинні мати мінімум 3 порти GigabitEthernet, а комутатори – 24 порти FastEthernet та один порт GigabitEthernet. Порти повинні мати запас для можливого розширення системи у майбутньому.

Обладнання має встановлюватись у відповідні комутаційні шафи.

Активне обладнання повинно відповідати вимогам енергоефективності, мати низький рівень шуму та забезпечувати швидкий доступ для технічного обслуговування та ремонту;

2) Вимоги до кабель-каналів, інформаційних та електричних розеток.

Кабель-канали повинні бути виготовлені з негорючих матеріалів і мати достатній розмір для прокладання кабелів з урахуванням їхнього майбутнього розширення.

Кабель-канали та розетки повинні бути розміщені зручно для підключення обладнання та забезпечувати легкий доступ для технічного обслуговування при модернізації або заміні-додавання компонентів;

3) Вимоги до комунікаційного обладнання і його розташування.

Комунікаційне обладнання має розміщуватися у спеціально відведених приміщеннях з контрольованими умовами температури та вологості.

Використовувати серверні шафи стандарту 19 дюймів, які забезпечують належний рівень вентиляції та доступу для обслуговування.

Кабельні траси повинні прокладатися з урахуванням можливості їх розширення приблизно від 30 до 40 процентів, використовуючи стандартизовані кріплення та захисні канали.

Обладнання повинно розміщуватися з урахуванням забезпечення оптимального повітряного потоку для охолодження та легкого доступу до кабельних з'єднань;

4) Вимоги до однорідності.

Система повинна використовувати однорідні типи кабелів, роз'ємів та магістралей для забезпечення сумісності та зниження витрат на технічне обслуговування. Це включає використання стандартних категорій Ethernet-

кабелів (наприклад, Cat5e або Cat6), однакових типів роз'ємів (RJ45) та кабельних трас.

5) Вимоги до резервування.

Система повинна мати резервування критичних компонентів, включаючи резервні канали зв'язку, дублюючі маршрутизатори та комутатори, а також джерела безперебійного живлення (UPS) для забезпечення безперервної цілодобової роботи у разі відмови основних компонентів.

б) Спеціальні вимоги.

Повинні бути передбачені механізми для регулярного резервного копіювання даних та швидкого відновлення у разі втрати інформації.

Система повинна відповідати вимогам інформаційної безпеки, включаючи використання шифрування даних, авторизації та аутентифікації користувачів, а також засобів захисту від кіберзагроз.

Повинна бути забезпечена сумісність та можливість інтеграції з існуючими інформаційними системами компанії для забезпечення безперервності бізнес-процесів.

2.1.2 Вимоги функцій, виконуваним системою

1) Вимоги до функцій комп'ютерної системи.

Перелік функцій та задач:

Управління мережею: моніторинг і управління мережевим трафіком, забезпечення безпеки мережі, управління підключеннями та доступом.

Управління даними: зберігання, обробка і резервне копіювання даних, управління базами даних, забезпечення цілісності та доступності даних.

Підтримка користувачів: управління обліковими записами користувачів, надання технічної підтримки, забезпечення доступу до корпоративних ресурсів.

Аналітика та звітність: збір, обробка та аналіз даних, генерація звітів і аналітичних довідок для підтримки прийняття рішень.

Часовий регламент і вимоги до якості: Управління мережею: затримка передачі даних не повинна перевищувати 50 мс, доступність мережі – 99.9%.

Управління даними: резервне копіювання повинно виконуватися щоденно, час відновлення даних – не більше 2 годин, цілісність даних – 100%.

Підтримка користувачів: час реакції на запити користувачів – не більше 1 години, вирішення проблем – не більше 24 годин.

Аналітика та звітність: звіти повинні генеруватися щотижня, точність даних – 99%, час виконання аналітичних запитів – не більше 30 хвилин.

2) Вимоги до функцій кіберфізичної системи

Перелік функцій та задач: моніторинг середовища: збір даних з датчиків температури, вологості, освітленості та інших параметрів.

Управління процесами: автоматичне керування системами поливу, освітлення, вентиляції та живлення на основі зібраних даних.

Аналіз і прогнозування: обробка даних для виявлення трендів і аномалій, прогнозування потреб рослин у ресурсах.

Аварійне реагування: виявлення і реагування на критичні ситуації, наприклад, перевищення допустимих рівнів температури або вологості.

Часовий регламент і вимоги до якості:

Моніторинг середовища: збір даних кожні 5 хвилин, точність вимірювань – $\pm 2\%$ для температури, $\pm 5\%$ для вологості.

Управління процесами: реакція на зміни параметрів середовища не більше 1 хвилини, доступність системи управління – 99.9%.

Аналіз і прогнозування: оновлення прогнозів кожні 24 години, точність прогнозів – 95%.

Аварійне реагування: час реагування на аварійну ситуацію – не більше 30 секунд, точність виявлення критичних ситуацій – 99%.

Ці вимоги забезпечать ефективну роботу як комп'ютерної системи, так і кіберфізичної системи, дозволяючи виконувати всі необхідні функції з високою якістю та в необхідні терміни.

2.1.3 Вимоги до видів забезпечення комп'ютерної системи

Для забезпечення коректної роботи системи, необхідно врахувати кілька важливих аспектів математичного забезпечення. Перш за все, для датчиків температури (DS18B20 і DHT11) потрібно реалізувати точне перетворення показників температури з формату, який повертають датчики, у звичайні одиниці (градуси Цельсія). Це включає в себе правильне калібрування та врахування можливих похибок вимірювань. Для DS18B20 використовується метод `read_temp()`, який повертає температуру в градусах Цельсія. Для DHT11 потрібно врахувати похибку в показниках вологості та температури, що може вимагати додаткової обробки результатів.

По-друге, код включає роботу з аналоговими датчиками, такими як датчики рівня води та рН. Для кожного з них важливо здійснювати точне аналогово-цифрове перетворення (ADC).

Нарешті, для забезпечення стабільного мережевого підключення та передачі даних на сервер використовується модуль Wi-Fi. Тут необхідно забезпечити надійне підключення до мережі та коректну передачу даних, включаючи обробку можливих помилок підключення або втрат даних. Це включає побудову правильного протоколу передачі даних, який використовує структуровані пакети для відправки множинних вимірювань на сервер, а також обробку з'єднань та відновлення у випадку збоїв. Важливо враховувати, що підключення до мережі може бути нестабільним, тому потрібні механізми повторного підключення та перевірки успішності передачі даних.

2.1.3.1 Вимоги до інформаційного забезпечення

Вимоги до комп'ютерної системи.

а) Склад, структура і способи організації даних.

У комп'ютерній системі повинна бути забезпечена чітка структура даних, яка включає всі необхідні компоненти для ефективною роботи. Це

передбачає використання уніфікованих форматів даних та стандартів для забезпечення узгодженості та цілісності інформації.

б) Інформаційний обмін між компонентами системи

Обмін інформацією між різними компонентами комп'ютерної системи має бути організований таким чином, щоб забезпечити швидкий та надійний доступ до даних. Для цього необхідно впровадити ефективні протоколи передачі даних та механізми синхронізації інформації між різними модулями системи.

в) Інформаційна сумісність із суміжними системами.

Комп'ютерна система має бути проєктована таким чином, щоб забезпечувати сумісність з іншими існуючими системами, які використовуються в компанії. Це включає підтримку стандартів обміну даними, інтеграцію з існуючими програмними та апаратними засобами, а також забезпечення можливості безперешкодного обміну інформацією з зовнішніми системами.

г) Системи керування базами даних.

Для ефективного зберігання та обробки даних у комп'ютерній системі необхідно застосовувати надійні системи керування базами даних (СКБД). Вибір СКБД повинен відповідати вимогам до швидкості доступу, обсягу даних та безпеки інформації.

Вимоги до кіберфізичної системи.

а) Структура процесу збору, обробки та передачі даних:

У кіберфізичній системі процеси збору, обробки та передачі даних повинні бути чітко визначені та структуровані. Це включає використання датчиків для збору даних, алгоритмів для їх обробки та надійних каналів для передачі інформації до центральної системи управління.

б) Контроль, збереження та відновлення даних:

Важливим аспектом кіберфізичної системи є забезпечення контролю за збереженням даних та можливістю їх відновлення у разі пошкодження або

втрати. Для цього необхідно впровадити механізми резервного копіювання даних, моніторингу стану системи та протоколів відновлення інформації.

в) Інформаційний обмін між компонентами системи:

Компоненти кіберфізичної системи повинні забезпечувати безперервний та ефективний обмін інформацією між собою. Це включає використання стандартних протоколів передачі даних та забезпечення надійності та безпеки цих процесів.

г) Інформаційна сумісність з іншими системами:

Кіберфізична система повинна бути сумісною з іншими інформаційними та технічними системами, які використовуються в теплиці та компанії в цілому. Це забезпечить інтегроване управління всіма процесами та підвищить ефективність роботи системи.

2.1.3.2 Вимоги до лінгвістичного забезпечення

а) Вимоги до мов програмування.

У системі повинні використовуватися мови програмування високого рівня, які забезпечують ефективну розробку та підтримку програмного забезпечення. Рекомендованими мовами є Python, C++, та Java, які мають високу продуктивність, зрозумілий синтаксис та широку підтримку спільноти.

б) Вимоги до мов взаємодії користувачів і технічних засобів.

Мови взаємодії користувачів:

– інтерфейс користувача повинен бути реалізований на основі мов, що забезпечують інтуїтивно зрозумілу та зручну взаємодію з системою. Основними мовами для розробки інтерфейсів є HTML, CSS, JavaScript для веб-застосунків та відповідні мови для розробки десктопних застосунків, такі як C# для Windows Forms або WPF.

Мови взаємодії технічних засобів:

– для забезпечення ефективної взаємодії між технічними засобами системи (наприклад, між датчиками та контролерами) повинні

використовуватися стандартизовані протоколи та мови, такі як MQTT, HTTP, Modbus, які забезпечують надійний обмін даними.

в) Вимоги до кодування і декодування даних.

Кодування та декодування даних:

– у системі повинні використовуватися стандартизовані методи кодування та декодування даних для забезпечення сумісності між різними компонентами системи. Рекомендується використовувати формати JSON, XML для обміну даними, а також базові методи кодування, такі як Base64, для передачі бінарних даних через текстові протоколи.

г) Вимоги до мов маніпулювання даними.

Мови маніпулювання даними:

– для обробки та маніпулювання даними у системі повинні використовуватися мови, що забезпечують ефективну роботу з базами даних та іншими джерелами інформації. Основною мовою для роботи з реляційними базами даних є SQL, тоді як для роботи з NoSQL базами можуть використовуватися відповідні API та запити на базі JavaScript (наприклад, MongoDB).

д) Вимоги до засобів опису предметної області.

Засоби опису предметної області:

– для опису предметної області (об'єкта автоматизації) повинні використовуватися спеціалізовані мови та нотації, такі як UML (Unified Modeling Language), що забезпечують візуальне моделювання та зрозумілий опис структури та поведінки системи.

е) Вимоги до способів організації діалогу

Способи організації діалогу:

– для забезпечення ефективної взаємодії користувача з системою повинні використовуватися сучасні методи організації діалогу, такі як інтерактивні вікна, асистенти на базі штучного інтелекту, підказки та повідомлення про помилки. Інтерфейси повинні бути розроблені з

урахуванням принципів юзабіліті та доступності, щоб забезпечити максимально зручне користування системою для всіх категорій користувачів.

2.1.3.3 Вимоги до технічного забезпечення

Для забезпечення ефективної роботи кожне робоче місце повинно бути обладнане комп'ютером з відповідною конфігурацією. Процесор має бути чотириядерним з тактовою частотою не нижче 2 ГГц. Об'єм оперативної пам'яті повинен становити не менше 8 ГБ, а для графічних задач необхідний дискретний відеоадаптер. Об'єм жорсткого диска має бути не менше 256 ГБ. В якості операційної системи слід використовувати Windows 10 або Windows 11.

Сервер має відповідати наступним технічним вимогам. Процесор повинен мати тактову частоту не нижче 1,5 ГГц, а об'єм оперативної пам'яті повинен бути не менше 8 ГБ.

Комутатор повинен відповідати наступним вимогам. Він має мати 24 порти FastEthernet та один порт GigabitEthernet. Комутатор також повинен підтримувати технології EtherChannel та VLAN для оптимізації мережевого трафіку.

Маршрутизатор повинен відповідати наступним вимогам. Він повинен мати мінімум 3 порти GigabitEthernet та 4 слоти EHWIC. Маршрутизатор має підтримувати функції DHCP, NAT, VPN та AAA модель для забезпечення безпеки та керованості мережі.

Вимоги до кіберфізичної системи

Для забезпечення функціонування кіберфізичної системи, яка базується на мікроконтролерах Arduino або Raspberry Pi Pico, необхідно дотримуватись наступних вимог:

Мікроконтролери:

– Arduino: повинні використовуватися моделі з достатньою кількістю цифрових та аналогових входів/виходів для підключення необхідних датчиків та виконавчих механізмів. Рекомендовані моделі: Arduino Uno, Arduino Mega;

– Raspberry Pi Pico: мікроконтролер з двоядерним процесором ARM Cortex-M0+ з тактовою частотою 133 МГц, 264 КБ SRAM та 2 МБ флеш-пам'яті. Підтримка PIO для гнучкого управління периферійними пристроями.

Сумісні датчики:

– датчики температури та вологості які забезпечують точні вимірювання та мають низьке енергоспоживання;

– датчики освітленості які дозволяють моніторити рівень освітлення для оптимального росту рослин;

– пристрої відстеження рівня рН які використовуються для контролю рівня кислотності розчину.

Енергозабезпечення:

– джерела живлення повинні забезпечувати стабільну роботу всіх компонентів системи. Рекомендується використовувати блоки живлення з резервним джерелом енергії (акумулятори) для забезпечення безперебійної роботи у випадку відключення електроенергії.

Програмне забезпечення:

– мікроконтролери повинні бути програмовані на гнучких мовах програмування, таких як C++ (для Arduino) та MicroPython (для Raspberry Pi Pico), для забезпечення можливості швидкого налаштування та адаптації системи.

Підключення блоку живлення системи проводиться від мережі змінного струму напругою 220В +15/-15%, з частотою (50±1) Гц.

Забезпечення цих вимог дозволить створити ефективну та надійну кіберфізичну систему для контролю процесів у гідропонній теплиці, що сприятиме оптимальному росту рослин та підвищенню загальної продуктивності системи.

2.1.3.4 Вимоги до організаційного забезпечення

а) Визначення відповідальних осіб.

В системі повинні бути чітко визначені ролі та відповідальні особи за кожен аспект її функціонування. Це включає адміністраторів системи, технічних спеціалістів, відповідальних за обслуговування мережі, операторів кіберфізичної системи, а також менеджерів з інформаційної безпеки.

б) Планування та управління.

Повинен бути розроблений план управління системою, який включає графіки технічного обслуговування, оновлення програмного забезпечення та обладнання, а також процедури для швидкого реагування на інциденти. План управління повинен бути затверджений керівництвом компанії і регулярно оновлюватися.

в) Технічна документація.

Система повинна супроводжуватися повною технічною документацією, яка включає інструкції з монтажу, конфігурації, експлуатації та обслуговування всіх компонентів. Документація повинна бути доступна всім відповідальним особам і регулярно оновлюватися.

г) Користувацька документація.

Повинні бути створені керівництва для користувачів, які пояснюють основні функції системи, порядок користування та правила безпеки. Користувацька документація має бути зрозумілою та доступною для всіх співробітників, які взаємодіють з системою.

д) Первинне навчання персоналу.

Після впровадження системи необхідно провести первинне навчання персоналу. Це включає навчальні курси для адміністраторів, технічних спеціалістів та кінцевих користувачів, що дозволить їм ефективно працювати з новою системою та розуміти її основні функції.

е) Періодичне навчання персоналу.

Повинні бути передбачені періодичні курси підвищення кваліфікації для персоналу. Це допоможе співробітникам залишатися в курсі новітніх технологій, оновлень програмного забезпечення та змін в організаційних процесах, пов'язаних з експлуатацією системи.

ж) План аварійного відновлення.

Необхідно розробити та впровадити план аварійного відновлення, який включає процедури резервного копіювання даних, відновлення системи після збоїв та управління ризиками. План повинен бути регулярно тестований та оновлюваний відповідно до змін в інфраструктурі та вимогах бізнесу.

з) План забезпечення безперервності бізнесу.

Повинен бути створений план забезпечення безперервності бізнесу, який визначає критичні процеси, ресурси та процедури для підтримки діяльності компанії під час аварійних ситуацій. План має враховувати всі можливі ризики та мати чітко визначені дії для їх мінімізації.

и) Регулярні аудити.

Повинні проводитися регулярні аудити системи для оцінки її відповідності встановленим вимогам, виявлення можливих вразливостей та перевірки ефективності впроваджених заходів безпеки. Результати аудитів повинні використовуватися для постійного вдосконалення системи та підвищення її надійності.

Ці вимоги до організаційного забезпечення дозволять забезпечити ефективну експлуатацію, обслуговування та розвиток комп'ютерної та кіберфізичної системи, що сприятиме стабільній та безперебійній роботі компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ».

2.1.3.5 Вимоги до методичного забезпечення

- представлення структурної схеми комплексу технічних засобів;
- розробка топологічної схеми корпоративної мережі;
- підготовка таблиці адресації пристроїв;
- створення таблиці специфікації обладнання;
- створення таблиці специфікації структурованої кабельної мережі;
- підготовка схеми розміщення кабельних мереж.

2.1.3.6 Вимоги до задач (налаштувань), які виконує КС

При розробці адресації підмереж необхідно враховувати такі вимоги. Корпоративна мережа повинна включати п'ять підмереж LAN1-LAN5 (рисунок 1.4), з кількістю вузлів у кожній: 54, 68, 197, 119 і 53 відповідно. Для виділення підмереж слід використовувати адресний блок 172.24.192.0/21. У найбільшій мережі середня інтенсивність вихідного трафіку має бути $\mu = 98$ кадрів/с, а затримка передачі пакету повинна бути ≤ 6 мс.

Для базового налаштування конфігурації пристроїв необхідно дотримуватися наступних вимог. Назви пристроїв повинні формуватися за схемою: Прізвище студента_тип пристрою_номер пристрою. На всіх пристроях слід встановити пароль cisco для доступу до консолі та vty, а також пароль class для привілейованого режиму. Всі паролі, що зберігаються у відкритому вигляді, повинні бути зашифровані. Потрібно створити банер MOTD та налаштувати використання протоколу ssh на всіх лініях vty. Ім'я користувача та пароль на всіх пристроях мають відповідати формату: група_прізвище з паролем admincisco. Ім'я домена повинно відповідати імені пристрою. Для шифрування даних слід створювати RSA ключ довжиною 1024 біти. На DCE-інтерфейсах маршрутизаторів слід встановити тактову частоту – 128000. Потрібно налаштувати аудит та відправку повідомлень про початок і завершення процесу ехес з використанням локальної бази даних. Для підвищення пропускної здатності та надійності каналів у мережі LAN_1 на комутаторах слід об'єднати фізичні лінії.

На маршрутизаторах потрібно встановити протокол OSPF для динамічного управління маршрутами, що включає підтримку багатьох маршрутів, швидку адаптацію до змін у мережі та мінімізацію обслуговувального трафіку. При конфігурації маршрутизаторів важливо задекларувати прямо підключені мережі та вимкнути розсилання оновлень маршрутів до локальних інтерфейсів. У контексті VLAN необхідно створити агрегований маршрут та оголосити його сусіднім маршрутизаторам. На

маршрутизаторі, що з'єднується з інтернет-провайдером, слід встановити маршрут за замовчуванням і дисемінувати його через оновлення маршрутів.

Для впровадження сервісу AAA на маршрутизаторах потрібно врахувати кілька важливих аспектів. Автентифікацію до VTY ліній слід здійснювати за допомогою локальної бази даних користувачів, а для доступу до консолі використовувати аутентифікацію через RADIUS, або ж, якщо неможливо, локальну базу. Налаштування RADIUS-сервера має включати ключеве слово "radius123" і використання імені обладнання з паролем "admin123" для користувачів.

Підключення до інтернету в системі вимагає вибору єдиного інтернет-провайдера. Для забезпечення доступу робочих станцій в інтернет слід налаштувати граничний маршрутизатор з динамічним NAT, налаштувавши пул з іменем "Internet" та адресами від 209.165.200.5 до 209.165.200.30, а також номером доступу до списку "12". Встановлення HTTP-сервера дозволить користувачам доступатися до веб-сайту за адресою <http://123.dnipro.ua> або <http://209.165.200.4>, де вони знайдуть інформацію про тему та завдання кваліфікаційної роботи. Також слід налаштувати VPN на базі IPsec для забезпечення безпечного зв'язку між корпоративними підрозділами через інтернет.

Конфігурація VLAN та маршрутизації між ними вимагає налаштування транкових та доступних портів, вимкнення невикористовуваних фізичних портів комутаторів та налаштування SVI-інтерфейсів на комутаторах з IPv4-адресами з мережі Management VLAN, а також налаштування маршрутизації між VLAN-мережами.

2.2 Розробка апаратної частини комп'ютерної системи

2.2.1 Розробка загальної архітектури мережі підприємства

З'єднання між маршрутизаторами виконується кабелями Serial DTE або крос-кабелями, що забезпечує надійний і швидкий обмін даними між різними сегментами мережі. Використання таких кабелів дозволяє створити стабільні

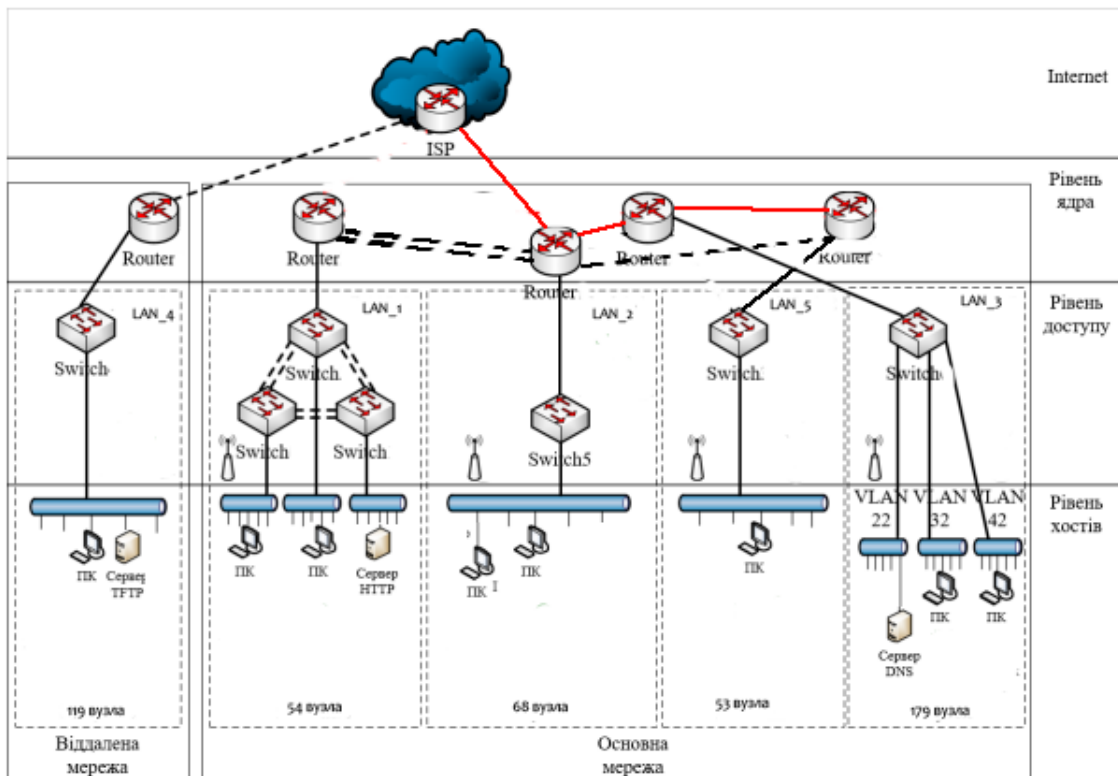
та ефективні з'єднання, необхідні для оптимального функціонування мережі. У цій мережевій архітектурі маршрутизатори відіграють ключову роль, забезпечуючи зв'язок між різними підмережами і дозволяючи їм обмінюватися інформацією.

Маршрутизатори з комутаторами поєднуються між собою за допомогою прямих кабелів, що забезпечує високу пропускну здатність та надійність з'єднань. Таке підключення дозволяє комутаторам ефективно передавати дані між різними пристроями в мережі, включаючи персональні комп'ютери, сервери та інші мережеві пристрої. Прямі кабелі є простими у встановленні та забезпечують стабільне з'єднання без втрат даних.

Комп'ютери підключаються до комутаторів також за допомогою прямих кабелів. Це забезпечує швидке та надійне підключення робочих станцій до мережі, що є критично важливим для забезпечення безперебійної роботи користувачів та доступу до необхідних ресурсів. Пряме підключення дозволяє уникнути затримок та забезпечити високу швидкість передачі даних.

Для з'єднання комутаторів між собою використовуються крос-кабелі, які дозволяють створювати високо ефективні і надійні з'єднання. Це особливо важливо у випадках, коли необхідно забезпечити резервування з'єднань або створити додаткові шляхи для передачі даних у великих мережах. Використання крос-кабелів гарантує стабільність і безперебійність роботи мережі навіть у випадку збоїв або підвищеного навантаження на окремі сегменти.

На основі загальної характеристики компанії, її архітектури, кількості підмереж та завдань, розроблено структурну схему комплексу технічних засобів комп'ютерної системи компанії, яка показана на рисунку 2.1. Ця схема відображає всі ключові компоненти мережі, їх взаємозв'язки та способи підключення, що дозволяє зрозуміти структуру та принципи роботи мережі. Вона є важливим інструментом для планування, розгортання та підтримки мережевої інфраструктури, забезпечуючи надійну основу для подальшого розвитку та модернізації системи [3].



Схематичне зображення компонентів комп'ютерної системи

2.2.2 Вибір і обґрунтування структурної схеми комплексу технічних засобів комп'ютерної системи

Вибір структурної схеми комплексу технічних засобів комп'ютерної системи ґрунтується на аналізі потреб компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ», що включає забезпечення надійного, ефективного та безпечного зв'язку між усіма компонентами системи. Основні вимоги до системи включають високу пропускну здатність, стійкість до відмов, легкість у масштабуванні та інтеграції нових технологій. Враховуючи ці вимоги, обрано схему, що включає три рівні: рівень ядра, рівень доступу та рівень хостів.

а) Рівень ядра.

На рівні ядра використовуються потужні маршрутизатори, які забезпечують централізоване управління трафіком та зв'язок між різними сегментами мережі. Ці маршрутизатори підключені один до одного за допомогою кабелів Serial DTE або крос-кабелів, що забезпечує високу швидкість та надійність передачі даних. Рівень ядра відповідає за основний

трафік та управління мережевими ресурсами, забезпечуючи стабільність та ефективність роботи всієї системи.

б) Рівень доступу.

Рівень доступу складається з комутаторів, які підключені до маршрутизаторів за допомогою прямих кабелів. Комутатори забезпечують розподіл трафіку між різними підмережами та користувачами, дозволяючи ефективно керувати потоками даних. Використання прямих кабелів для підключення комутаторів до маршрутизаторів та комп'ютерів забезпечує високу пропускну здатність та мінімальні затримки у передачі даних. На цьому рівні також реалізується підтримка VLAN для логічного розділення трафіку та забезпечення безпеки.

в) Рівень хостів.

На рівні хостів розміщуються робочі станції, сервери та інші кінцеві пристрої, які підключені до комутаторів за допомогою прямих кабелів. Це забезпечує стабільне та швидке підключення кожного пристрою до мережі. Використання VLAN на цьому рівні дозволяє організувати логічні групи пристроїв, що підвищує ефективність управління мережевими ресурсами та забезпечує додатковий рівень безпеки.

Обрана структурна схема забезпечує всі необхідні вимоги для ефективної роботи комп'ютерної системи компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ». Вона забезпечує високу пропускну здатність та надійність завдяки використанню сучасних маршрутизаторів та комутаторів, підтримує масштабованість завдяки модульній структурі, а також забезпечує безпеку даних за допомогою VLAN та інших механізмів захисту. Крім того, така схема є гнучкою та легко адаптується до змін у структурі компанії та впровадження нових технологій, що робить її оптимальним вибором для даної компанії.

2.2.3 Розробка специфікації апаратних засобів комп'ютерної системи

Вибір апаратної частини корпоративної мережі є важливим кроком для забезпечення надійності та продуктивності системи. У якості маршрутизатора було обрано Cisco ASR1001-X.

Ця модель має високу продуктивність та розширені можливості для обробки даних мережі. Технічні характеристики моделі наступні: має два WAN портів, дев'ять портів під підключення комутаторів швидкістю до 1 Gbps(9x10/1000) та мережевий екран швидкістю до 5Gbps. Маршрутизатор Cisco ASR1001-X відповідає потребам компанії для забезпечення стабільного та захищеного зв'язку між мережевими пристроями.

Для подальшої побудови локальної мережі, обрано комутатор Cisco Catalyst 2960X-24TS-L. Даний комутатор має: 24 порти Fast Ethernet (10/100 Mbps) та два порти зі швидкістю до 1 Gbps(9x10/1000). Комутатор підтримує такі функції як VLAN, Quality of Service (QoS), Spanning Tree Protocol (STP), Access Control Lists (ACLs), що дозволяють налаштовувати та керувати роботою комутатора, також він забезпечує швидку передачу даних, надає широкі можливості управління, безпеки та масштабованості, відповідає потребам компанії для забезпечення надійного з'єднання робочих станцій, серверів та інших мережевих пристроїв.

Для мережі буде використовуватись маршрутизатор MikroTik RB4011iGS+, який є технологічно новим пристроєм на ринку, спеціально розробленим для вимогливих корпоративних мереж. Основні характеристики маршрутизатора включають: Маршрутизатор підтримує стандарт Wi-Fi 6 (802.11ax) і забезпечує вражаючу швидкість до 6000 Мбіт/с, що дозволяє швидко і безперебійне підключення для багатьох користувачів одночасно. Підтримка технології Quality of Service (QoS) для пріоритетного виділення трафіку, що дозволяє гарантувати стабільну роботу важливих додатків та послуг. Можливість налаштовувати віртуальні приватні мережі (VPN) забезпечує безпеку та конфіденційність передачі даних у мережі. Підтримка можливості підключення додаткових мережевих пристроїв, що дозволяє легко розширювати мережу за необхідності.

Таблиця 2.1 – Специфікація обладнання

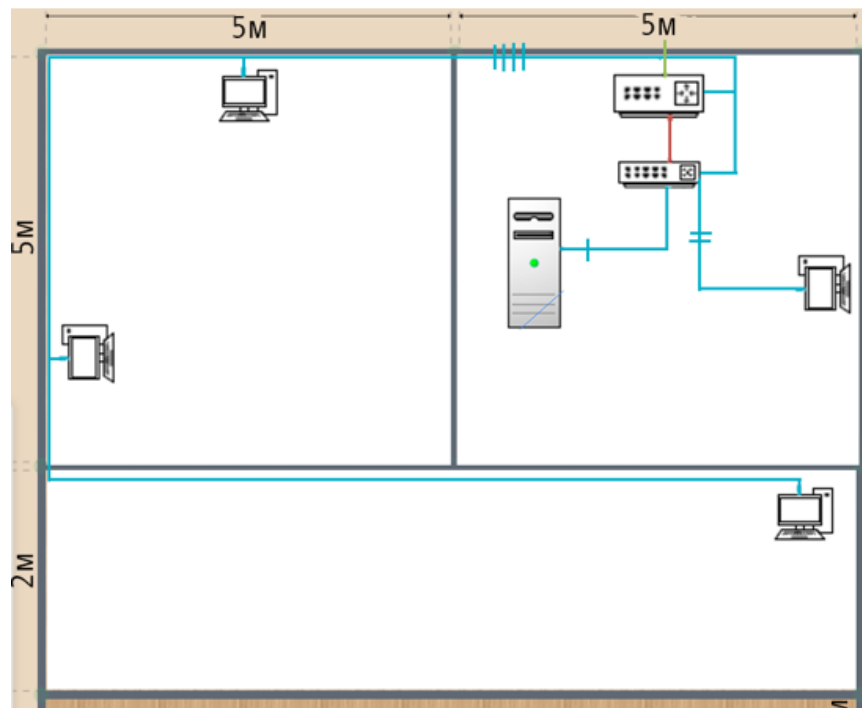
| Позиція | Найменування і технічна характеристика | Тип, марка, позначення документа | Одиниці виміру | Кількість |
|---------|---|--|----------------|-----------|
| 1 | 2 | 3 | 4 | 5 |
| 1 | Cisco ASR1001-X System: підтримка шифрування, 4 вбудовані порти GE, подвійне живлення, пропускна здатність 20 Gbit, 6 портів 1000Base-X (SFP), 2 порти 10G SFP+, інтегровані RP, SIP та ESP, 1 слот NIM, 1 слот SPA, 8 ГБ оперативної пам'яті, 2 блоки живлення AC. | Cisco ASR1001-X Cheredbychenko_R 1 Cheredbychenko_R2 Cheredbychenko_R3 Cheredbychenko_R4 Cheredbychenko_R0 | Од. | 5 |
| 2 | Комутатор: 24 порти Ethernet 10/100/1000 Мбіт/с, підтримка протоколів RIP v1, RIP v2, OSPF, наявність USB-порту, програмна платформа LAN Base, 4 слоти SFP. | Cisco Catalyst 2960X-24TS-L Cheredbychenko_SW 1-23 | Од. | 23 |

Продовження таблиці 2.1

| | | | | |
|---|--|---|-----|---------|
| 3 | AL21400 Cortex A15 CPU: 4 ядра, 1.4 ГГц на кожне ядро, 1 ГБ оперативної пам'яті, 10 портів Gigabit LAN, 1 порт SFP+, бездротовий модуль 5GHz 802.11ac з чотирма антенами, бездротовий модуль 2.4GHz 802.11b/g/n 2x2. | MikroTik RB4011iGS+ Cheredbychenko_WiFi1 Cheredbychenko_WiFi2 Cheredbychenko_WiFi3 Cheredbychenko_WiFi4 | Од. | 4 |
| 4 | Сервер: 2 процесори Intel Xeon E5-2650L v2 (1.70-2.10 ГГц), 8 ГБ оперативної пам'яті DDR3, 2 порти 1 Gb Ethernet, інтегрований контролер управління Cisco | Cisco UCS C220 M3 LFF | Од. | 3 |
| 5 | Комп'ютер: процесор AMD Ryzen 5 5600G (3.9 — 4.4 ГГц), 16 ГБ оперативної пам'яті DDR4, накопичувач SSD на 1 ТБ, відеокарта AMD Radeon Vega 7, операційна система Windows 11 Pro. | ARTLINE Business B38v08Win | Од. | 33 1 |

Всі компоненти мережі, окрім бездротових роутерів, вибрані від виробника Cisco, що забезпечує високий рівень сумісності усіх елементів системи. Подивимося на процес вибору структурованої кабельної мережі на прикладі LAN_3. Для цього розроблено план поверху та розташування кабелів, як це демонструється на ілюстрації 2.2. В структурованій кабельній системі передбачено необхідну кількість обладнання відповідно до технічного завдання замовника, що передбачає збільшену масштабованість, за винятком кількості кабелів і кабельних каналів.

Специфікація структурованої кабельної мережі наведена в таблиці 2.2.



Розташування кабельної інфраструктури для мережі LAN_3

Таблиця 2.2 – Специфікація структурованої кабельної мережі

| Позиція | Найменування | Тип, марка, позначення документа, опитувально го листа | Одиниці | Кількість |
|---------|---|--|---------|-----------|
| 1. | 2. | 3. | 4. | 5. |
| 1. | Кабельний канал 40×25 мм | «ЭЛЕКОР» | м. | 32 |
| 2. | Розетка мережева з інтерфейсом RJ-45 подвійна | Schneider Electric Asfora | од. | 91 |

Продовження таблиці 2.2

| | | | | |
|----|---------------------------------------|---------------------------|-----|-----|
| 3. | Lan кабель UTP КПВ-ВП cat.5E 4x2x0,51 | Одескабель | м. | 50 |
| 4. | Розетка настінна подвійна | Schneider Electric Asfora | од. | 182 |
| 5. | Кабель живлення ПВС 3*2,5 | Одескабель | м. | 50 |
| 6. | Кабельний канал 40×25 мм | «ЭЛЕКОР» | м. | 32 |
| 7. | Комутаційна коробка | Schneider Electric | од. | 1 |

2.2.4 Розрахунок інтенсивності трафіку вихідного трафіку найбільшої локальної мережі підприємства

Мережа налічує 179 ПК працівників. Вихідний трафік з центрального комутатора надсилається до роутера в лінію з пропускну здатністю, що становить 1000 Мбіт/с [3].

Для того, щоб комутатор не був перенасичений, швидкість надходження пакетів не повинна перевищувати швидкості їх відправлення. Вважаємо, що послугами одночасно користуються 100% користувачів. Середня інтенсивність трафіку $\mu=98$ (кадрів/с), а середня довжина повідомлення – 1150 байт.

Теоретично припустимо, що всі користувачі найбільшої підмережі одночасно використовують мережу. В такому разі, пропускна здатність на рівні доступу буде дорівнювати:

$$P_{p.p.} = \mu * L_{пов} * N * 8 = 98 * 1150 * 179 * 8 = 20.7 \text{ Мбіт/с} \quad (2.1)$$

де $L_{пов}$ – середня довжина повідомлення;

N – кількість вузлів в мережі.

Отриманий результат не перевищуватиме заданих параметрів мережі по вихідному каналу, отже перенавантажень не трапиться.

Комутатор також передає трафік до маршрутизатора зі швидкістю 1000 Мбіт/с. Отже, загальне навантаження на комутатор не повинно перевищувати:

$$\mu_{вих} = 10^9 / (1150 * 8) = 108\,696 \text{ пакетів/с} \quad (2.2)$$

Оскільки в середньому, кожне джерело виробляє 98 пакетів/с, то маршрутизатор обмежений кількістю приєднань, яку ми можемо дізнатись наступним чином:

$$N = \mu_{\text{вих}} / \mu = 108\,696 / 98 \approx 1109 \text{ джерела} \quad (2.3)$$

Ця кількість задовольняє кількості вузлів у найбільшій нашій локальній мережі, до якої входить 179 ПК.

Кожен з 179 ПК посилає потік заявок з інтенсивністю у 98 кадрів/с. Звідси, можемо розрахувати інтенсивність вихідного трафіку:

$$\lambda = N * \mu = 179 * 98 = 2254 \text{ пакетів/с.} \quad (2.4)$$

Коефіцієнт затримки:

$$\rho = \frac{\lambda}{\mu_{\text{вих}}} = \frac{2254}{108696} = 0,020 \quad (2.5)$$

Коефіцієнт зайнятості маршрутизатора:

$$\frac{\rho}{1-\rho} = \frac{0,020}{1-0,020} = 0,020 \quad (2.6)$$

Середня затримка кадру, пов'язана з чергою M/M/1, дорівнює:

$$T = \frac{1}{(\mu-\lambda)} = \frac{1}{(108696 - 2254)} = 9,3 \text{ мкс} \quad (2.7)$$

Середня довжина черги:

$$L_{\text{чер}} = \frac{\rho^2}{1-\rho} = \frac{0,020^2}{1-0,020} = 0,00040 \quad (2.8)$$

Середній час перебування пакета у черзі:

$$T_{\text{оч}} = \frac{L_{\text{чер}}}{\lambda} = \frac{0,0040}{2254} = 0,17 \text{ мкс} \quad (2.9)$$

Це значення задовольняє вимогам до затримки в ЛМ.

3 РОЗРОБКА КОРПОРАТИВНОЇ МЕРЕЖІ

3.1 Проектування логічної топології мережі

Відповідно до вимог в розділі 2.1.1, організаційної структури компанії та топології мережі (рисунок 1.4) в Packet Tracer була побудована модель мережі компанії (ри.3.1).

Підмережа LAN_1 – повинна забезпечувати 54 вузлів та до неї входить HTTP сервер для надання веб-сторінок.

Підмережа LAN_2 – повинна забезпечувати 68 вузлів.

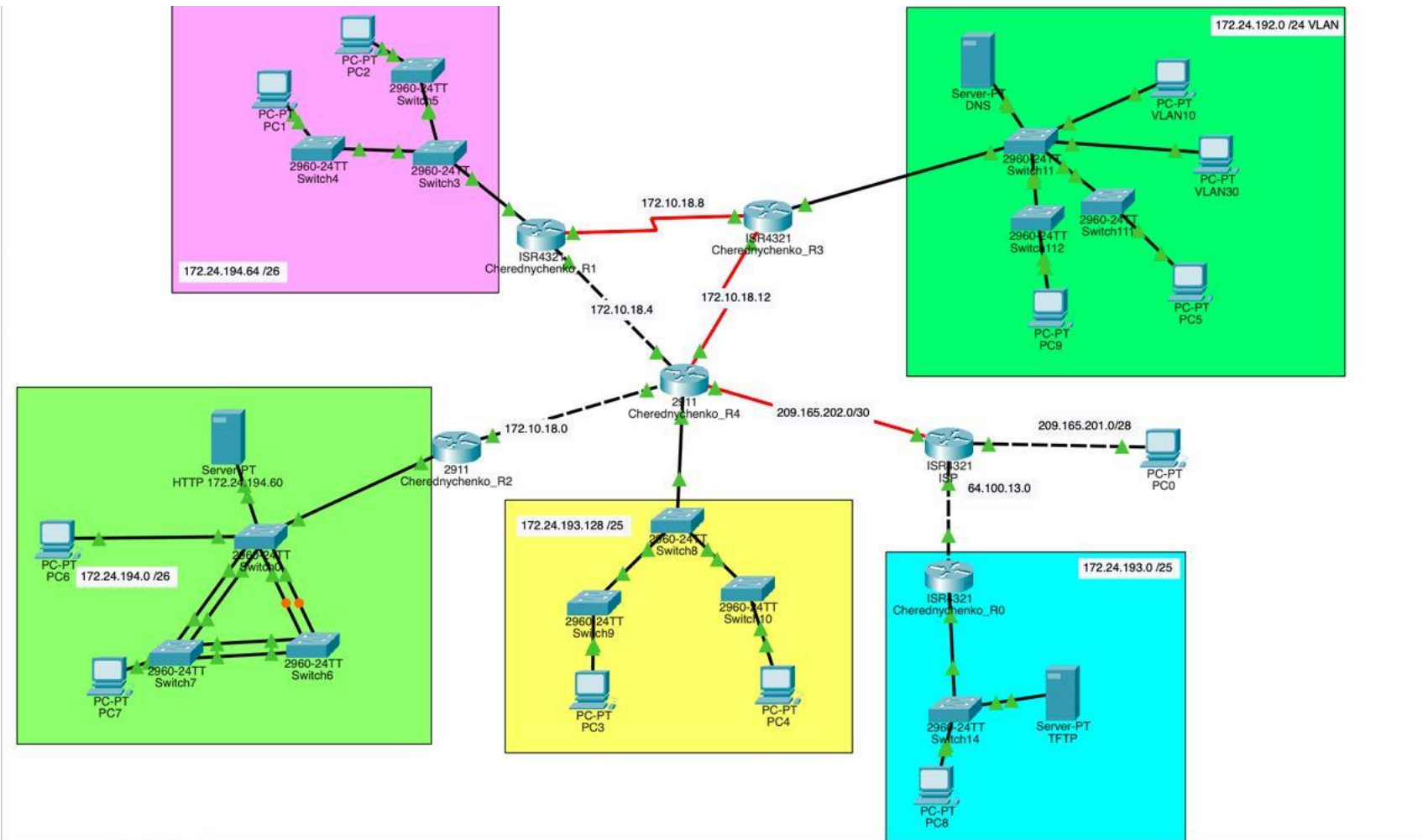
Підмережа LAN_3 – повинна забезпечувати 179 вузлів та до неї входить DNS сервер для конвертації(перекладу) доменних імен в IP-адреси.

Підмережа LAN_4 – віддалена мережа яка повинна забезпечувати 119 вузлів.

Підмережа LAN_5 – повинна забезпечувати 53 вузлів.

Загалом підсистема офісу повинна налічувати 354 вузла, а віддалена мережа 119 вузлів.

Логічна схема складається з трьох основних компонентів: основної мережі, віддаленої мережі та мережі провайдера. Основна мережа включає робочі станції, сервери та мережеві пристрої, розташовані на центральному об'єкті. Віддалена мережа з'єднується з основною для забезпечення доступу віддалених користувачів та філій до корпоративних ресурсів. Мережа провайдера забезпечує підключення до Інтернету та інших зовнішніх ресурсів. З'єднання між цими мережами здійснюється за допомогою кабелів SerialEthernet для передачі даних на великих відстанях і GigabitEthernet для високошвидкісного з'єднання в межах локальних сегментів мережі. [3].



Топологія мережі

3.2 Вибір та опис мережного обладнання

Для реалізації високопродуктивної корпоративної мережі, обрано низку мережевих рішень, що відповідають сучасним вимогам до надійності, швидкості та безпеки даних.

Маршрутизатор Cisco ASR1001-X оснащений широким набором функцій, включаючи криптографічні можливості, чотири вбудованих порти GE, подвійні блоки живлення, пропускну спроможність 20 Gbit та мультимедійні порти. Також він підтримує до шести портів 1000Base-X (SFP) і два порти 10G SFP+, забезпечуючи високу гнучкість підключення. Цей маршрутизатор був обраний за його здатність управляти великим обсягом даних та надійність у критичних застосуваннях, ідеально підходить для використання як основний маршрутизатор в структурі КТЗ.

Комутатор Cisco Catalyst 2960X-24TS-L має 24 порти Ethernet зі швидкістю до 1000 Мбіт/сек та підтримкою протоколів RIP v1, RIP v2 та OSPF. Також він оснащений чотирма SFP слотами, що дозволяє гнучко налаштовувати мережу під різні потреби. Обрано за його високу продуктивність та надійність, комутатор є ідеальним рішенням для організації точок агрегації в мережевій топології.

Сервер Cisco UCS C220 M3 оснащений двома процесорами Intel Xeon E5-2650L v2, 8 GB RAM та двома портами Ethernet 1 Gb.

Це рішення ідеально підходить для важливих мережевих служб, таких як HTTP, DNS та TFTP, завдяки його високій продуктивності та надійності.

Кожен компонент вибрано з метою забезпечення оптимальної продуктивності, надійності та ефективності для різних аспектів корпоративної мережі, від основних мережевих з'єднань до кінцевих користувацьких пристроїв.

3.3 Розрахунок схеми адресації корпоративної мережі

Розділення мережі 172.24.192.0/21 на підмережі передбачає створення окремих сегментів IP-адресного простору для кожної з підмереж. Метод

VLSM дозволяє ефективно управляти цим процесом, забезпечуючи точне відповідність кількості адрес кожної підмережі до її реальних потреб (табл.3.1).

Таблиця 3.1 – Блок адрес мережі та кількість вузлів в кожній підмережі

| Блок адрес | LAN1 | LAN2 | LAN3 | LAN4 | LAN5 |
|-----------------|------|------|------|------|------|
| 172.24.192.0/21 | 54 | 68 | 179 | 119 | 53 |

Тобто, необхідно створити 5 підмереж для 473 користувачів.

Для розділення блоку адрес 172.24.192.0/21 на п'ять підмереж використовується метод VLSM, що дозволяє ефективно розподілити адресний простір в залежності від потреб кожної підмережі у вузлах.

Для LAN1 з 51 вузлами: використовуємо маску /26, що забезпечує 62 адреси, досить для 54 вузлів.

Для LAN2 з 68 вузлами: обираємо маску /25, надаючи 126 адрес, що достатньо для 68 вузлів.

Для LAN3 з 179 вузлом: також використовуємо маску /24 що забезпечує 254 адреси, досить для 179 вузлів.

Для LAN4 з 119 вузлами: застосовуємо маску /25, що забезпечує 126 адреси, досить для 119 вузлів.

Для LAN5 з 53 вузлом: обираємо маску /26, що забезпечує 62 адреси, досить для 53 вузлів.

Для виділення підмережі переведемо адресу мережі в двійковий вид і відокремимо частину, в яку входить вже вибрана маска.

1) Підмережа LAN1 (51 вузлів)

Визначення маски: Для підтримки 51 вузлів з врахуванням мережевої та ширококомовної адрес, використовуємо маску /26, яка надає 62 адрес.

Двійковий розрахунок: LAN1 - 172.24.194.1, що у двійковому форматі: 10101100.00011000.11000010.00000001. Широкомовна адреса для цієї підмережі буде - 172.24.194.63, що в двійковому форматі виглядає як 10101100.00011000.11000010.00111111.

Діапазон IP-адрес: 172.24.194.1 - 172.24.194.62.

2) Підмережа LAN2 (68 вузлів)

Визначення маски: Для підтримки 67 вузлів з врахуванням мережевої та широкомовної адрес, використовуємо маску /25, яка надає 128 адрес.

Двійковий розрахунок: Виходимо з наступної адреси - 172.24.193.128, що у двійковому форматі: 10101100.00011000.11000001.10000000. Широкомовна адреса для цієї підмережі буде 172.24.193.255, що в двійковому форматі виглядає як 10101100.00011000.11000001.11111111.

Діапазон IP-адрес: 172.24.193.129 - 172.24.193.254.

3) Підмережа LAN3 (179 вузлів)

Визначення маски: Для підтримки 179 вузлів з врахуванням мережевої та широкомовної адрес, використовуємо маску /24, яка надає 254 адрес.

Двійковий розрахунок: Виходимо з наступної адреси - 172.24.192.0, що у двійковому форматі: 10101100.00011000.11000000.00000000. Широкомовна адреса для цієї підмережі буде 172.24.192.255, що в двійковому форматі виглядає як 10101100.00011000.11000000.11111111.

Діапазон IP-адрес: 172.24.192.1 - 172.24.192.254.

4) Підмережа LAN4 (119 вузлів)

Визначення маски: Для підтримки 119 вузлів з врахуванням мережевої та широкомовної адрес, використовуємо маску /25, яка надає 128 адрес.

Двійковий розрахунок: Виходимо з наступної адреси - 172.24.193.0, що у двійковому форматі: 10101100.00011000.11000001.00000000. Широкомовна адреса для цієї підмережі буде 172.24.193.127, що в двійковому форматі виглядає як 10101100.00011000.11000001.01111111.

Діапазон IP-адрес: 172.24.193.1 - 172.24.193.126.

5) Підмережа LAN5 (53 вузлів)

Визначення маски: Для підтримки 53 вузлів з врахуванням мережевої та широкомовної адрес, використовуємо маску /26, яка надає 64 адрес.

Двійковий розрахунок: Виходимо з наступної адреси - 172.24.194.64, що у двійковому форматі: 10101100.00011000.11000001.01000000. Широкомовна

адреса для цієї підмережі буде 172.24.194.127, що в двійковому форматі виглядає як 10101100.00011000.11000010.01111111.

Діапазон IP-адрес: 172.24.194.65 - 172.24.194.126.

Розроблена адресація підмереж узагальнена в таблиці 3.2

Таблиця 3.2 – Схема адресації мережі

| Назва мережі | Кількість вузлів | Номер мережі | Маска мережі | Початкове значення діапазону адрес вузлів у підмережі | Кінцеве значення діапазону адрес вузлів у підмережі |
|--------------|------------------|----------------|--------------|---|---|
| LAN1 | 51 | 172.24.194.0 | /26 | 172.24.194.1 | 172.24.194.62 |
| LAN2 | 68 | 172.24.193.128 | /25 | 172.24.193.129 | 172.24.193.254 |
| LAN3 | 179 | 172.24.192.0 | /24 | 172.24.192.1 | 172.24.192.254 |
| LAN4 | 119 | 172.24.193.0 | /25 | 172.24.193.1 | 172.24.193.126 |
| LAN5 | 53 | 172.24.194.64 | /26 | 172.24.194.65 | 172.24.194.126 |

Для з'єднання між маршрутизаторами буде використовуватися адресний блок 172.24.200.0/21. Використовуючи метод VLSM, цей блок буде розділений на п'ять підмереж, кожна з яких міститиме по два вузли. Таблиця 3.3 містить детальну схему адресації для цих з'єднань між маршрутизаторами.

Таблиця 3.3 – Схема адресації каналів між маршрутизаторами

| Назва мережі | Кількість вузлів | Номер мережі | Префікс мережі | Початкове значення діапазону адрес вузлів у підмережі | Кінцеве значення діапазону адрес вузлів у підмережі |
|--------------|------------------|---------------|----------------|---|---|
| WAN1 | 2 | 172.24.200.0 | /30 | 172.24.200.1 | 172.24.200.2 |
| WAN2 | 2 | 172.24.200.4 | /30 | 172.24.200.5 | 172.24.200.6 |
| WAN3 | 2 | 172.24.200.8 | /30 | 172.24.200.9 | 172.24.200.10 |
| WAN4 | 2 | 172.24.200.12 | /30 | 172.24.200.13 | 172.24.200.14 |
| WAN5 | 2 | 172.24.200.16 | /30 | 172.24.200.17 | 172.24.200.18 |

Маршрутизаторам IP-адреси відповідно до вимог призначається перша із допустимих адрес з діапазону розрахованих по табл. 3.2 адрес. Комутаторам наступні адреси.

Адреси SVI-інтерфейсів комутаторів наведені у таблиці 3.4.

Таблиця 3.4 – Схема адресації пристроїв

| Підмережа | Пристрій | IP-адреса SVI інтерфейсу | Маска підмережі | Адреса шлюзу |
|-----------|--------------------------|--------------------------|-----------------|----------------|
| LAN1 | Cherednychenko_Switch 0 | 172.24.194.1 | 255.255.255.192 | 172.24.194.1 |
| | Cherednychenko_Switch 1 | 172.24.194.2 | 255.255.255.192 | 172.24.194.1 |
| | Cherednychenko_Switch 2 | 172.24.194.3 | 255.255.255.192 | 172.24.194.1 |
| LAN2 | Cherednychenko_Switch 3 | 172.24.193.129 | 255.255.255.128 | 172.24.193.129 |
| | Cherednychenko_Switch 4 | 172.24.193.130 | 255.255.255.128 | 172.24.193.129 |
| | Cherednychenko_Switch 5 | 172.24.193.131 | 255.255.255.128 | 172.24.193.129 |
| LAN3 | Cherednychenko_Switch 6 | 172.24.192.1 | 255.255.255.0 | 172.24.192.1 |
| LAN3 | Cherednychenko_Switch 7 | 172.24.192.2 | 255.255.255.0 | 172.24.192.1 |
| | Cherednychenko_Switch 8 | 172.24.192.3 | 255.255.255.0 | 172.24.192.1 |
| | Cherednychenko_Switch 9 | 172.24.192.4 | 255.255.255.0 | 172.24.192.1 |
| | Cherednychenko_Switch 10 | 172.24.192.5 | 255.255.255.0 | 172.24.192.1 |
| | Cherednychenko_Switch 11 | 172.24.192.6 | 255.255.255.0 | 172.24.192.1 |
| | Cherednychenko_Switch 12 | 172.24.192.7 | 255.255.255.0 | 172.24.192.1 |
| | Cherednychenko_Switch 13 | 172.24.192.8 | 255.255.255.0 | 172.24.192.1 |
| LAN4 | Cherednychenko_Switch 14 | 172.24.193.1 | 255.255.255.128 | 172.24.193.1 |
| | Cherednychenko_Switch 15 | 172.24.193.2 | 255.255.255.128 | 172.24.193.1 |
| | Cherednychenko_Switch 16 | 172.24.193.3 | 255.255.255.128 | 172.24.193.1 |
| | Cherednychenko_Switch 17 | 172.24.193.4 | 255.255.255.128 | 172.24.193.1 |
| | Cherednychenko_Switch 18 | 172.24.193.5 | 255.255.255.128 | 172.24.193.1 |
| | Cherednychenko_Switch 19 | 172.24.193.6 | 255.255.255.128 | 172.24.193.1 |
| LAN5 | Cherednychenko_Switch 20 | 172.24.194.65 | 255.255.255.192 | 172.24.194.65 |
| | Cherednychenko_Switch 21 | 172.24.194.66 | 255.255.255.192 | 172.24.194.65 |
| | Cherednychenko_Switch 22 | 172.24.194.67 | 255.255.255.192 | 172.24.194.65 |

В таблиці 3.5 наведена IP-адресація інтерфейсів маршрутизаторів.

Таблиця 3.5 –Адресація інтерфейсів маршрутизаторів

| Пристрій | Інтерфейс | IP-адреса | Маска | Мережа |
|-------------------|-----------|----------------|-----------------|----------------|
| Cherednychenko_R2 | G0/0 | 172. 24.200.2 | 255.255.255.252 | 172. 24.200.0 |
| | G0/2 | 172.24.194.1 | 255.255.255.192 | 172.24.194.0 |
| Cherednychenko_R4 | G0/0 | 172. 24.200.1 | 255.255.255.252 | 172. 24.200.0 |
| | G0/1 | 172.24.193.129 | 255.255.255.128 | 172.24.193.128 |
| | G0/2 | 172. 24.200.5 | 255.255.255.252 | 172. 24.200.4 |
| | S0/3/0 | 172. 24.200.14 | 255.255.255.252 | 172. 24.200.12 |
| | S0/3/1 | 209.165.202.1 | 255.255.255.252 | 209.165.202.0 |
| Cherednychenko_R3 | S0/1/0 | 172. 24.200.10 | 255.255.255.252 | 172. 24.200.8 |
| | S0/1/1 | 172. 24.200.13 | 255.255.255.252 | 172. 24.200.12 |
| Cherednychenko_R1 | G0/0 | 172. 24.200.6 | 255.255.255.252 | 172. 24.200.4 |
| | G0/1 | 172.24.194.65 | 255.255.255.192 | 172.24.194.64 |
| | S0/1/0 | 172. 24.200.9 | 255.255.255.252 | 172. 24.200.8 |
| Cherednychenko_R0 | G0/0 | 64.100.13.2 | 255.255.255.252 | 64.100.13.0 |
| | G0/1 | 172.24.193.1 | 255.255.255.128 | 172.24.193.0 |

3.4 Базове налаштування конфігурації пристроїв

Базове налаштування конфігурації пристроїв включає в себе налаштування та шифрування паролів доступу до vty ліній, доступу до різних режимів конфігурації та привілейованого режиму, налаштування банеру, ім'я пристрою за стандартами та налаштування доменного імені пристрою.

Розглянемо налаштування на прикладі маршрутизатора Cherednychenko_R4:

```
hostname Cherednychenko_R4 // призначення назви пристрою
line console 0 // вхід в конфігураційний режим лінії консолі
password cisco // призначення паролю до консолі
login // вимикання анонімного доступу
line vty 0 15 // вхід в конфігураційний режим лінії VTY
password cisco // призначення паролю до лінії VTY
login // вимикання анонімного доступу
enable secret class // встановлення зашифрованого паролю для
привілейованого режиму
service password-encryption // шифрування паролів
banner motd #Cherednychenko_R4# // налаштування банера MOTD
line vty 0 15 // вхід в конфігураційний режим лінії VTY
transport input ssh // призначення використання протоколу SSH
```

```
login local // налаштування локальної аутентифікації
username CherednychenkoOIV_R4 password admincisco // призначення
імені користувача та паролю
ip domain-name Cherednychenko_R4 // налаштування імені домена
crypto key generate rsa // створення ключа шифрування
1024 // вибір довжини ключа шифрування
```

У мережі LAN_1 активно використовується технологія EtherChannel, яка дозволяє агрегувати кілька фізичних портів мережевого пристрою в один логічний канал. Цей підхід не тільки значно збільшує пропускну спроможність з'єднання, але й підвищує його надійність завдяки резервуванню та балансуванню навантаження між портами.

Технологія EtherChannel можна використовувати для агрегації від двох до восьми фізичних лінків в один логічний канал, який забезпечує сумарну пропускну спроможність всіх з'єднаних лінків. Це робиться за допомогою спеціалізованого протоколу, такого як Link Aggregation Control Protocol (LACP), що є частиною стандарту IEEE 802.3ad. LACP допомагає автоматично конфігурувати та управляти агрегованими лінками, забезпечуючи оптимальне розподілення трафіку.

Налаштування Etherchannel на прикладі комутатора з підмережі:

```
interface range fa0/1-2 // вибір інтерфейсів
channel-group 1 mode active // налаштування режиму портової групи
interface port-channel 1 // вибір інтерфейсу портової групи
switchport mode trunk // налаштування портової групи в режим транку
switchport trunk allowed vlan all // встановлення всіх VLAN як дозволених
для проходження даних через транковий порт
interface range fa0/3-4
channel-group 2 mode active
interface port-channel 2
switchport mode trunk
switchport trunk allowed vlan all
```

3.5 Налаштування DHCP

Для автоматичного призначення IP-адрес комп'ютерам у мережі буде застосовано протокол DHCP (Dynamic Host Configuration Protocol). Цей протокол значно спрощує процес управління мережею та конфігурацію кінцевих пристроїв, автоматично розподіляючи IP-адреси, маски підмереж, шлюзи та інші мережеві налаштування.

Переваги використання DHCP включають:

- автоматизація конфігурації: зменшує потребу в ручному призначенні IP-адрес, знижуючи ризик помилок та спрощуючи процес додавання нових пристроїв до мережі;

- економія часу: забезпечує швидке підключення нових пристроїв з мінімальним втручанням адміністратора;

- гнучкість: дозволяє легко змінювати мережеві налаштування з централізованої точки управління, що особливо корисно в динамічних мережевих середовищах.

Обрані маршрутизатори Cisco можуть виконувати роль DHCP-сервера. Це зручно і економічно, так як не потрібно додавати в мережу окремий виділений ПК з ПЗ DHCP-сервера.

Налаштування DHCP на прикладі маршрутизатора Cherednychenko_R4:

```
ip dhcp excluded-address 172.24.194.29 172.24.194.135 // виключення
вказаних адрес з dhcp пулів
ip dhcp excluded-address 172.24.194.254
ip dhcp excluded-address 172.24.194.255
ip dhcp pool LAN-2 // створення та вказання адреси dhcp пулу
network 172.24.193.128 255.255.255.128 // вказання IP-адреси мережі
default-router 172.24.193.129 // вказання IP-адреси шлюзу
dns-server 172.24.192.200 // вказання IP-адреси dns сервера
```

3.6 Налаштування маршрутизації

Для забезпечення взаємодії між користувачами, які знаходяться в різних підмережах, необхідно налагодити міжмережеву маршрутизацію. У цьому

контексті існують два основні методи налаштування маршрутів: статична маршрутизація та динамічна маршрутизація. Статична маршрутизація вимагає ручного введення маршрутів, що може бути часом неефективним у великих або динамічно змінюваних мережах. Динамічна маршрутизація, з іншого боку, автоматично адаптується до змін у мережі, оновлюючи маршрути для підтримки оптимальної роботи без постійного втручання адміністратора.

У цій роботі використовується протокол динамічної маршрутизації OSPF (Open Shortest Path First), який є одним з найбільш розповсюджених протоколів у сучасних мережах. OSPF використовує алгоритм SPF (Shortest Path First) для визначення найефективніших маршрутів в мережі, забезпечуючи швидкі та ефективні шляхи передачі даних.

Переваги використання OSPF включають:

- масштабованість: OSPF ефективно працює в малих, середніх та великих мережах, підтримуючи тисячі вузлів;
- безпека: протокол підтримує аутентифікацію між вузлами для запобігання несанкціонованому оновленню маршрутів;
- підтримка VLSM (Variable Length Subnet Masking): це дозволяє більш гнучке управління IP-адресацією;
- сумісність з обладнанням різних виробників, що робить OSPF відмінним вибором для гетерогенних мереж.

Налаштування протоколу OSPF на прикладі маршрутизатора Cherednychenko_R4:

```
router ospf 1 // увімкнення протоколу
log-adjacency-changes
passive-interface default // відключення поширення оновлень за
замовчуванням на всіх портах
no passive-interface GigabitEthernet0/0 // увімкнення поширення
оновлень на портах, через які будуть передаватись дані щодо підключених
мереж
no passive-interface GigabitEthernet0/1
no passive-interface GigabitEthernet0/2
```

```

no passive-interface Serial0/3/0
no passive-interface Serial0/3/1
network 172.24.200.0 0.0.0.3 area 0
network 172.24.200.4 0.0.0.3 area 0
network 172.24.200.12 0.0.0.3 area 0
network 172.24.200.16 0.0.0.3 area 0
network 172.24.193.128 0.0.0.127 area 0

```

На граничному маршрутизаторі Cherednychenko_R4 налаштуємо маршрут за замовчуванням до маршрутизатора інтернет-провайдера (ISP) і розповсюджуємо його.

```

ip route 0.0.0.0 0.0.0.0 209.165.202.2 // налаштуємо маршрут за
замовчуванням
router ospf 1 // увімкнення протоколу
redistribute static subnets // увімкнення розповсюдження статичних
маршрутів через протокол OSPF

```

Додаємо статичний маршрут до мережі постачальника інтернет-послуг (ISP).

```
ip route 209.165.201.0 255.255.255.240 209.165.202.2
```

3.7 Налаштування доступу в Інтернет

Так як для адресації в мережі компанії обрана приватна адреса, то для забезпечення доступу системи до Інтернету необхідно сконфігурувати NAT (Network Address Translation) – технологію, яка дозволяє трансформувати внутрішні IP-адреси у публічні та навпаки, а також змінювати номери портів для вхідного та вихідного трафіку [4].

Пул адрес NAT: 209.165.202.5 – 209.165.202.30.

Переглянемо налаштування NAT на прикладі прикордонного маршрутизатора Cherednychenko_R4:

```

ip access-list extended NAT12 // створення списку NAT12
deny ip 172.24.194.0 0.0.0.63 172.24.193.0 0.0.0.127 // заборона
находження пакетів з віддаленої мережі до основної мережі
deny ip 172.24.193.128 0.0.0.127 172.24.193.0 0.0.0.127
deny ip 172.24.192.0 0.0.0.255 172.24.193.0 0.0.0.127
deny ip 172.24.194.64 0.0.0.63 172.24.193.0 0.0.0.127
deny ip 172.10.18.0 0.0.0.255 172.24.193.0 0.0.0.127
permit ip 172.24.194.0 0.0.0.63 any // дозвіл на надходження будь-яких
пакетів з основної мережі
permit ip 172.24.193.128 0.0.0.127 any
permit ip 172.24.192.0 0.0.0.255 any
permit ip 172.24.194.64 0.0.0.63 any
permit ip 172.10.18.0 0.0.0.255 any
ip nat pool Internet 209.165.200.5 209.165.200.30 netmask 255.255.255.224
// створення пулу адрес
ip nat inside source list NAT12 pool Internet
ip nat inside source static 172.24.192.200 209.165.200.4 // призначення
HTTP серверу IP-адреси NAT
ip nat inside source static 172.24.194.60 209.165.200.3
interface Serial0/3/0
ip nat inside
interface GigabitEthernet0/0 // вибір інтерфейсу
ip nat inside // вказання інтерфейсу як внутрішнього
interface GigabitEthernet0/1
ip nat inside
interface GigabitEthernet0/2
ip nat inside
interface Serial0/3/1
ip nat outside // вказання інтерфейсу як зовнішнього

```

3.8 Налаштування мереж VLAN, маршрутизації між VLAN

Технологія VLAN є інструментом для логічного розділення однієї фізичної мережі на кілька віртуальних мереж, кожна з яких діє як окрема підмережа. Застосування VLAN дозволяє обмежити необхідність фізичної

конфігурації пристроїв і кабельних мереж, підвищує безпеку шляхом ізоляції мережевих сегментів та сприяє ефективнішому управлінню трафіком [5].

Основні переваги використання VLAN включають:

- поліпшене управління мережею: VLAN спрощують управління мережею, дозволяючи групувати користувачів або служби незалежно від їхнього фізичного розташування в мережі;

- зменшення заторів мережі: відділення трафіку в межах окремих VLAN може допомогти знизити загальне навантаження на мережу, оскільки мережеві взаємодії відбуваються лише в межах кожного VLAN;

- забезпечення безпеки: ізоляція користувацьких груп і мережевих служб у різних VLAN може запобігти несанкціонованому доступу до критичних систем або конфіденційних даних;

- гнучкість і масштабованість: VLAN дозволяють легко змінювати мережеві налаштування без потреби в перекладанні кабелів чи переміщенні обладнання;

- кращий контроль якості обслуговування (QoS): У мережах з використанням VoIP або відеоконференцій, VLAN можуть допомогти пріоритизувати важливий трафік і забезпечити більш високий рівень якості послуг.

Підмережа LAN_5 була поділена на три підмережі VLAN. Схема адресації для підмереж VLAN наведена в таблиці 3.

Таблиця 3.6 – Адресація мереж VLAN

| Name | Network Address | Slash | Mask | Usable Range |
|--------|-----------------|-------|-----------------|---------------------------------|
| VLAN10 | 172.24.192.128 | /26 | 255.255.255.192 | 172.24.192.129 - 172.24.192.190 |
| VLAN20 | 172.24.192.64 | /26 | 255.255.255.192 | 172.24.192.65 - 172.24.192.127 |
| VLAN30 | 172.24.192.192 | /26 | 255.255.255.192 | 172.24.192.193 - 172.24.192.254 |
| VLAN99 | 172.24.192.1 | /28 | 255.255.255.240 | 172.24.192.2 - 172.24.192.15 |

Таблиця розподілу портів для окремих мереж VLAN представлена в таблиці 3.7.

Таблиця 3.7 – Розподіл портів для окремих мереж VLAN

| VLAN | Розподіл портів |
|------|-----------------|
| 10 | F0/6-11 |
| 20 | F0/12-14 |
| 30 | F0/15-24 |

Налаштування VLAN на прикладі комутатора з мережі:

```

int range fa0/6-11 // вибір портів
switchport mode access // налаштування портів
switchport access vlan 10 // присвоювання портам влану
int range fa0/12-14
switchport mode access
switchport access vlan 20
int range fa0/15-24
switchport mode access
switchport access vlan 30
int range fa0/1-5
switchport mode trunk // налаштування портів в режим транку
switchport trunk native vlan 99 // налаштування власної мережі на
транковому
switchport trunk allowed vlan 10,20,30,99-100 //налаштування списку
дозволених VLAN на транковому порті

```

Налаштування портів на комутаторах, привласнивши їм адреси з мережі VLAN, на прикладі комутатора:

```

int vlan 99 // вибір VLAN
ip address 172.24.192.2 255.255.255.240 // призначення IP-адреси
ip default-gateway 172.24.192.1 // вказання IP-адреси шлюзу за
замовчуванням

```

Налаштовуємо підінтерфейси на маршрутизаторі Cherednychenko_R3, що будуть виступати в ролі шлюзу для вказаних VLAN:

```

int g0/0/0.20 // вибір підінтерфейсу
encapsulation dot1Q 20 // встановлення мітки для вибраного порту

```



```

        ip address 172.24.192.33 255.255.255.128 // вказання IP-адреси
підінтерфейсу
        int g0/0/0.10
        encapsulation dot1Q 10
        ip address 172.24.192.129 255.255.255.192
        int g0/0/0.30
        encapsulation dot1Q 30
        ip address 172.24.192.193 255.255.255.192
        int g0/0/0.99
        encapsulation dot1Q 99
        ip address 172.24.192.3 255.255.255.240

```

Для автоматичного призначення IP-адрес вузлам у різних VLAN буде використовуватися протокол DHCP. Налаштування DHCP на маршрутизаторі Cherednychenko_R3, який виконуватиме роль DHCP-сервера:

```

        ip dhcp excluded-address 172.24.192.1 172.24.192.15 // виключення
вказаних адрес з dhcp пулів
        ip dhcp excluded-address 172.24.192.17 172.24.192.31
        ip dhcp excluded-address 172.24.192.33 172.24.192.35
        ip dhcp excluded-address 172.24.192.129 172.24.192.131
        ip dhcp excluded-address 172.24.192.192 172.24.192.205
        ip dhcp excluded-address 172.24.192.253 172.24.192.255
        ip dhcp pool LAN5-VLAN10 // створення та вказання адреси dhcp пулу
network 172.24.192.128 255.255.255.192 // вказання IP-адреси мережі
default-router 172.24.192.129 // вказання IP-адреси шлюзу
dns-server 172.24.192.200 // вказання IP-адреси dns сервера
        ip dhcp pool LAN5-VLAN30
network 172.24.192.192 255.255.255.192
default-router 172.24.192.193
dns-server 172.24.192.200
        ip dhcp pool LAN5-VLAN20
network 172.24.192.64 255.255.255.192
default-router 172.24.192.65
dns-server 172.24.192.200

```

3.9 Захист інформації в комп'ютерній системі від несанкціонованого доступу

3.9.1 Налаштування маршрутизаторів на підтримку служби AAA

AAA (Аутентифікація, Авторизація, Облік) є важливою моделлю в мережевій безпеці, яка допомагає управляти доступом до мережевих ресурсів. Ця модель забезпечує не тільки авторизацію і ідентифікацію користувачів, але й веде облік їхньої діяльності, що дозволяє слідкувати за доступом до мережі та використанням ресурсів.

Один з ключових компонентів системи AAA є RADIUS-сервер (Remote Authentication Dial-In User Service), який виконує функції централізованої аутентифікації, авторизації та обліку для користувачів, що намагаються здійснити доступ до мережі через різні точки входу, такі як комутатори і маршрутизатори. RADIUS-сервер спрощує адміністрування безпеки мережі, оскільки всі дані про користувачів та їхні повноваження зберігаються централізовано, що дозволяє легко управляти політиками доступу і реагувати на зміни в безпекових вимогах.

Крім того, використання RADIUS-сервера надає додаткові переваги:

- гнучкість управління політиками: можливість встановлення детальних правил для груп користувачів або окремих користувачів;
- зменшення навантаження на адміністраторів: автоматизація процесу аутентифікації та авторизації зменшує потребу в ручному втручанні;
- підвищення безпеки: за допомогою RADIUS можна імплементувати багаторівневу верифікацію, яка забезпечує додатковий захист від несанкціонованого доступу.

Налаштовуємо всі маршрутизаторів на підтримку служби AAA на прикладі Cherednychenko_R3:

```
aaa new-model // увімкнення служби AAA
radius-server host 172.24.194.60 auth-port 1645 key radius123 // вказання IPадреси RADIUS-серверу, порту підключення та ключа аутентифікації
```

```

aaa authentication login console group radius local // налаштування
аутентифікації для консольного доступу до мережевого пристрою з
використанням RADIUS-сервера
line console 0 // вхід в режим конфігурації лінії консолі
login authentication console // встановлення методу аутентифікації для
доступу до консольного порту
aaa authentication login default local // створення локальної бази даних
користувачів
username CherednychenkoOIV password admin123 // налаштування логіну
та паролю у локальній базі
line vty 0 15 // вхід в режим конфігурації ліній віртуального терміналу
login authentication default // встановлення за замовчуванням методу
аутентифікації для доступу через VTY-порти

```

3.9.2 Налаштування віртуальної приватної мережі VPN

VPN є технологією, яка дозволяє створювати захищене з'єднання через відкриті, незахищені мережі, такі як Інтернет. Це забезпечує конфіденційність та безпеку даних, коли користувачі підключаються до основної мережі з віддалених локацій [6].

Основні характеристики та переваги VPN:

- захищене шифрування: всі дані, які передаються через VPN, зашифровані, що запобігає їх перехопленню або читанню третіми особами;
- анонімність: VPN дозволяє приховати реальну IP-адресу користувача, надаючи йому можливість переглядати веб-ресурси без розкриття особистої інформації;
- обхід географічних обмежень: VPN може використовуватися для доступу до інтернет-ресурсів, які зазвичай обмежені в певних регіонах або країнах;
- безпечний віддалений доступ: використання VPN є ідеальним рішенням для віддалених працівників або організацій з кількома філіалами, які потребують безпечного доступу до корпоративної мережі з різних частин світу.

Застосування VPN у нашому випадку полягає в наданні віддаленим співробітникам або філіалам можливості підключення до центральної корпоративної мережі. Це не тільки забезпечує безпеку передачі конфіденційних даних через публічні мережі, але й сприяє підтримці продуктивності та ефективності діяльності організації.

Налаштування VPN розглянемо на прикладі Cherednychenko_R0:

```
license boot module c2900 technology-package securityk9 // активація
модуля securityk9
ip access-list extended VPN18 // створення ACL-списку VPN18, щоб
визначити трафік з основної мережі до віддаленої
permit ip 172.24.194.0 0.0.0.63 172.24.193.0 0.0.0.127 // надання доступу
на проходження пакетів з основної на віддалену мережу
permit ip 172.24.193.128 0.0.0.127 172.24.193.0 0.0.0.127
permit ip 172.24.192.0 0.0.0.255 172.24.193.0 0.0.0.127
permit ip 172.24.194.64 0.0.0.63 172.24.193.0 0.0.0.127
permit ip 172.10.18.0 0.0.0.255 172.24.193.0 0.0.0.127
crypto isakmp policy 10 // створення криптографічної політики
encr 3des // вибір алгоритму шифрування
hash md5 // вибір алгоритму створення геш-суми
authentication pre-share // вибір методу аутентифікації пірів
group 2
crypto isakmp key cisco address 209.165.202.1 // створення ключа для
взаємодії з обраним партнером
crypto ipsec transform-set TS esp-3des esp-md5-hmac // створення набору
перетворень
crypto map MAP 10 ipsec-isakmp // створення криптографічного
з'явлення
set peer 209.165.202.1 // створення піра
set transform-set TS // вибір набору перетворень
match address VPN18 // прив'язка до списку VPN18
int GigabitEthernet0/1 // вибір інтерфейсу
crypto map MAP // прив'язка криптографічного з'явлення MAP до
вихідного інтерфейсу
```

3.10 Перевірка компонентів корпоративної мережі компанії

Перевірено базові налаштування мережевого обладнання на прикладі маршрутизатора Cherednychenko_R4. Використовуючи команду show running-

config з привілейованого режиму, детально перевірено ряд важливих параметрів конфігурації.

Це включає в себе ім'я пристрою, що відображено на рисунку 3.2, налаштування пароля для доступу через консоль (рисунок 3.3), налаштування та забезпечення безпеки ліній vty з використанням протоколу SSH (рисунок 3.4), встановлення пароля для доступу до привілейованого режиму (рисунок 3.5), конфігурація банера MOTD, який відображається при спробі доступу до системи (рисунок 3.6), а також перевірка імені користувача та пароля (рисунок 3.7) та налаштування імені домену (рисунок 3.8).

```
!
hostname Cherednychenko_R4
!
```

Назва маршрутизатора

```
:
!
line con 0
password 7 0822455D0A16
login authentication console
!
```

Доступ за допомогою пароля на консольному режимі

```
:
line vty 0 4
password 7 0822455D0A16
login authentication default
transport input ssh
line vty 5 15
password 7 0822455D0A16
login authentication default
transport input ssh
,
```

Доступ до vty ліній за допомогою паролю

```
:
!
enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCi1
!
```

Доступ до привілейованого режиму за допомогою пароля

```
!
banner motd ^CCherednychenko_R4^C
!
```

Банер MOTD

```
username Cherednychenko01V password 7 082048430017544541
```

Ім'я користувача та пароль

```
!
ip domain-name Cherednychenko_R4
!
```

Ім'я домена на комутаторі

Як бачимо, усі паролі захищені шифруванням. Перевірка тактової частоти на DCE-інтерфейсах маршрутизаторів (рисунок 3.9) на прикладі маршрутизатора Cherednychenko_R4 виконується за допомогою команди `do show controllers`, попередньо увійшовши до необхідного інтерфейсу за командою `int serial0/3/1`.

```
Interface Serial0/3/1
Hardware is PowerQUICC MPC860
DCE V.35, clock rate 128000
idb at 0x81081AC4, driver data structure at 0x81084AC0
SCC Registers:
General [GSMR]=0x2:0x00000000, Protocol-specific [PSMR]=0x8
Events [SCCE]=0x0000, Mask [SCCM]=0x0000, Status [SCCS]=0x0
Transmit on Demand [TODR]=0x0, Data Sync [DSR]=0x7E7E
```

Тактова частота комутаторів мережі

Перевіряємо технологію EtherChannel в LAN_1 (рисунок 3.10).

```
-----
Switch#show etherchannel summary
Flags: D - down          P - in port-channel
       I - stand-alone  s - suspended
       H - Hot-standby (LACP only)
       R - Layer3       S - Layer2
       U - in use       f - failed to allocate aggregator
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port

Number of channel-groups in use: 2
Number of aggregators:          2

Group  Port-channel  Protocol    Ports
-----+-----+-----+-----
1      Po1(SU)          LACP       Fa0/1(P) Fa0/2(P)
2      Po2(SU)          LACP       Fa0/3(P) Fa0/4(P)
```

Технологія EtherChannel

Перевіряємо налаштування маршрутів на прикладі маршрутизатора Cherednychenko_R4. За допомогою команди `show ip protocols` яка виводить маршрути протоколу `ospf` (рисунок 3.11).





```

Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 209.165.202.1
  It is an autonomous system boundary router
  Redistributing External Routes from,
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    172.24.200.0 0.0.0.3 area 0
    172.24.200.4 0.0.0.3 area 0
    172.24.200.12 0.0.0.3 area 0
    172.24.200.16 0.0.0.3 area 0
    172.24.193.128 0.0.0.127 area 0
  Passive Interface(s):
    Vlan1
  Routing Information Sources:
    Gateway Distance Last Update
    172.24.192.193 110 00:02:30
    172.24.194.1 110 00:01:56
    172.24.194.65 110 00:01:56
    209.165.202.1 110 00:01:56

```

Налаштований OSPF

Як бачимо з рисунок 3.12, зв'язок між різними підмережами на прикладі підмереж LAN_5 та LAN_1 є успішним.

| ire | Last Status | Source | Destination | Type | Color | Time(sec) | Periodic | Num | Edit | Delete |
|-----|-------------|---------|-------------|-------|---|-----------|----------|-----|------|----------|
| | -- | Cher... | HTTP 17... | IC... |  | 0.000 | N | 0 | (... | (delete) |
| | Failed | VLA... | PC6 | IC... |  | 0.000 | N | 1 | (... | (delete) |
| | Successful | VLA... | PC6 | IC... |  | 0.000 | N | 2 | (... | (delete) |
| | Successful | PC6 | VLAN10 | IC... |  | 0.000 | N | 3 | (... | (delete) |

Зв'язок між LAN_5 та LAN_1

На прикладі маршрутизатора Cherednychenko_R4 перевіряємо призначену пропускну здатність серійних інтерфейсів за допомогою команди `show interfaces serial0/3/1` (рисунок 3.13). Налаштування маршруту за замовчуванням на маршрутизаторі Cherednychenko_R4, який має пряме підключення до інтернет-провайдера, перевіряємо за допомогою команди `show ip route static` (рисунок 3.14).

```

-----,-----_-----
Cherednychenko_R4#show interfaces serial0/3/1
Serial0/3/1 is up, line protocol is up (connected)
Hardware is HD64570
Internet address is 209.165.202.1/30
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
Conversations 0/0/256 (active/max active/max total)
Reserved Conversations 0/0 (allocated/max allocated)
Available Bandwidth 1158 kilobits/sec
5 minute input rate 51 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  57 packets input, 3648 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
DCD=up DSR=up DTR=up RTS=up CTS=up

```

Пропускна спроможність на serial-інтерфейсі

```

Cherednychenko_R4#
Cherednychenko_R4#show ip route static
S* 0.0.0.0/0 is directly connected, Serial0/3/1

```

Конфігурація маршруту за умовчанням у маршрутизаторі

За допомогою маршрутизатора Cherednychenko_R4 ми перевіряємо налаштування служби AAA на всіх маршрутизаторах. Як показано, при спробі входу система вимагає ввести логін та пароль (рисунок 3.15).

```

Press RETURN to get started!

Cherednychenko_R4

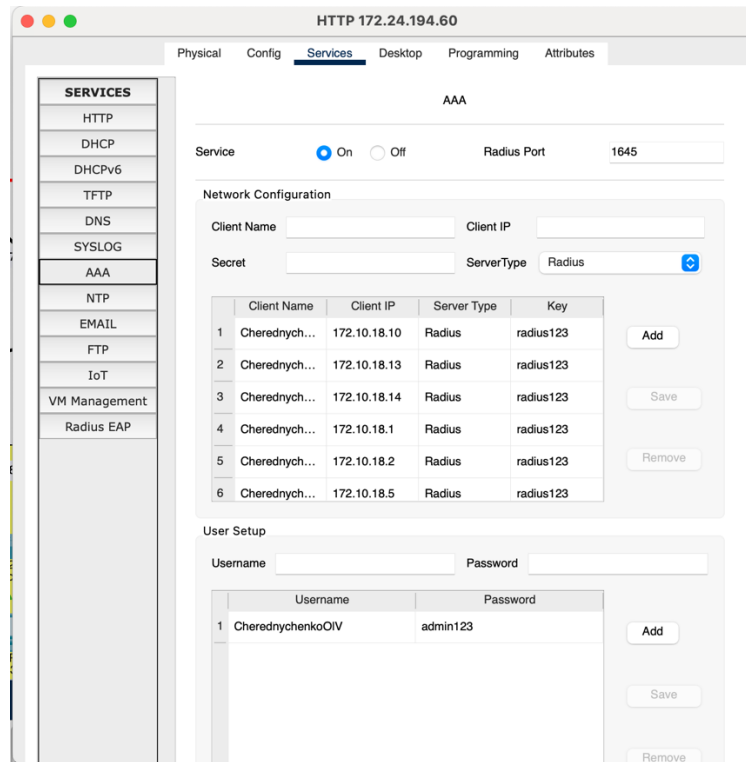
User Access Verification

Username: Cherednychenko01V
Password:
Cherednychenko_R4>

```

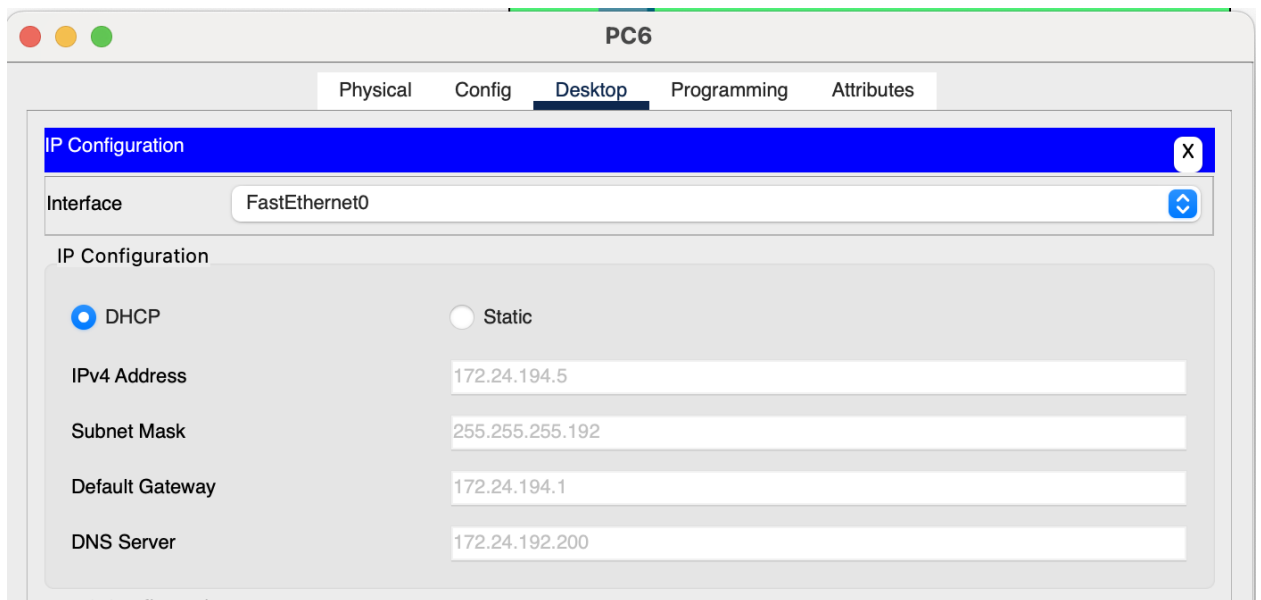
Налаштований маршрутизатор на підтримку служби AAA

Налаштування захисту за допомогою RADIUS-сервера наведено на рисунку 3.16.



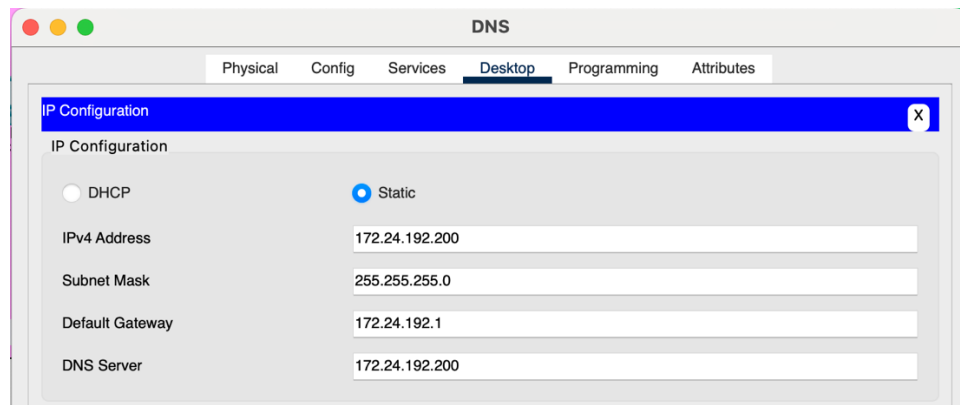
Конфігурація сервера RADIUS

Конфігурацію DHCP перевіряємо, використовуючи комп'ютер PC6 у мережі LAN_1 (рисунок 3.17).

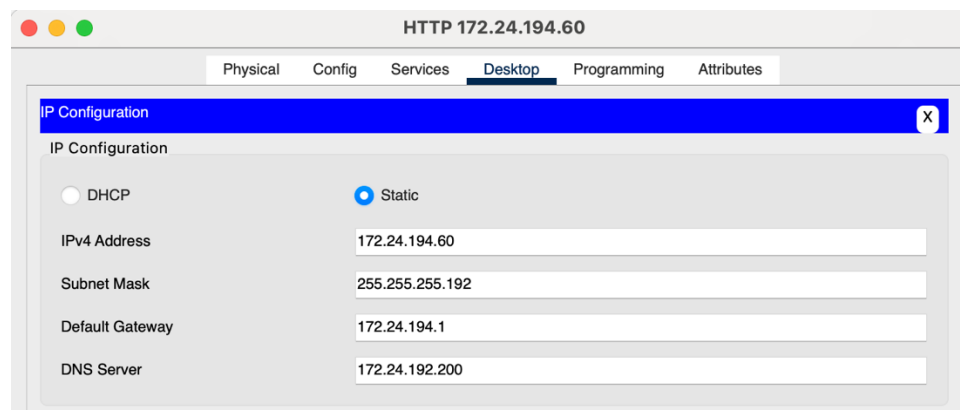


IP-адреса PC6

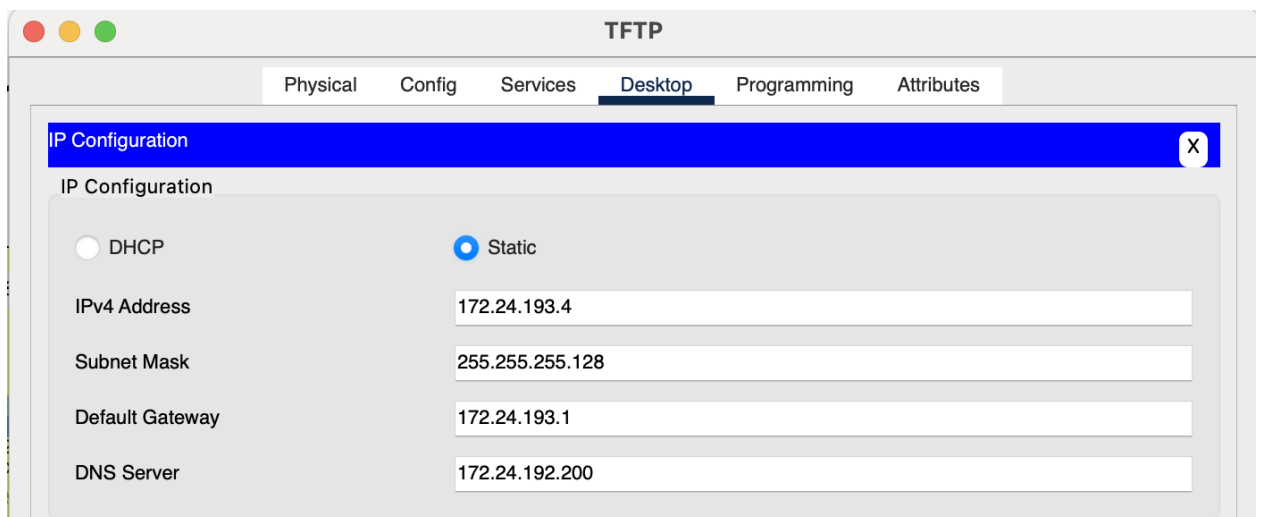
Перевіряємо призначені статичні IP-адреси серверам DNS (рисунок 3.18), HTTP(рисунок 3.19) та TFTP (рисунок 3.20).



IP-адреса DNS-сервера



IP-адреса HTTP-сервера



IP-адреса TFTP-сервера

На прикладі маршрутизатора Cherednychenko_R4 перевіряємо призначені IP-адреси за допомогою команди `show ip interface brief` з привілейованого режиму (рисунок 3.21).

```
Cherednychenko_R4#show ip interface brief
Interface          IP-Address      OK? Method Status        Protocol
GigabitEthernet0/0 172.10.18.1    YES manual up            up
GigabitEthernet0/1 172.24.193.129 YES manual up            up
GigabitEthernet0/2 172.10.18.5    YES manual up            up
Serial0/3/0        172.10.18.14   YES manual up            up
Serial0/3/1        209.165.202.1 YES manual up            up
Vlan1              unassigned     YES unset  administratively down down
Cherednychenko_R4#
```

IP-адреси маршрутизатора

Перевіряємо призначені імена VLAN та порти, що належать кожному VLAN, за допомогою команди `show vlan` (рисунок 3.22).

```
Switch#show vlan

VLAN Name                Status    Ports
-----
1    default                 active    Fa0/1, Fa0/3, Fa0/4, Fa0/11
    Gig0/2
10   Human_Resources_Department active    Fa0/12, Fa0/13, Fa0/14
20   IT_Department           active    Fa0/5, Fa0/6, Fa0/7, Fa0/8
    Fa0/9, Fa0/10
30   Service_Department     active    Fa0/15, Fa0/16, Fa0/17, Fa0/18
    Fa0/19, Fa0/20, Fa0/21, Fa0/22
    Fa0/23, Fa0/24
99   Management              active    Gig0/1
100  Native                  active
1002 fddi-default            active
1003 token-ring-default     active
1004 fddinet-default       active
1005 trnet-default        active

VLAN Type  SAID      MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
-----
1    enet  100001   1500  -     -     -     -   -         0      0
10   enet  100010   1500  -     -     -     -   -         0      0
20   enet  100020   1500  -     -     -     -   -         0      0
30   enet  100030   1500  -     -     -     -   -         0      0
99   enet  100099   1500  -     -     -     -   -         0      0
100  enet  100100   1500  -     -     -     -   -         0      0
1002 fddi  101002   1500  -     -     -     -   -         0      0
1003 tr    101003   1500  -     -     -     -   -         0      0
1004 fdnet 101004   1500  -     -     -     -   ieee      0      0
1005 trnet 101005   1500  -     -     -     -   ibm       0      0

VLAN Type  SAID      MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
-----

Remote SPAN VLANs
-----

Primary Secondary Type          Ports
-----
```

Імена та порти VLAN

```
Switch#
Switch#show int trunk
Port      Mode          Encapsulation  Status        Native vlan
Fa0/2     on            802.1q         trunking     100

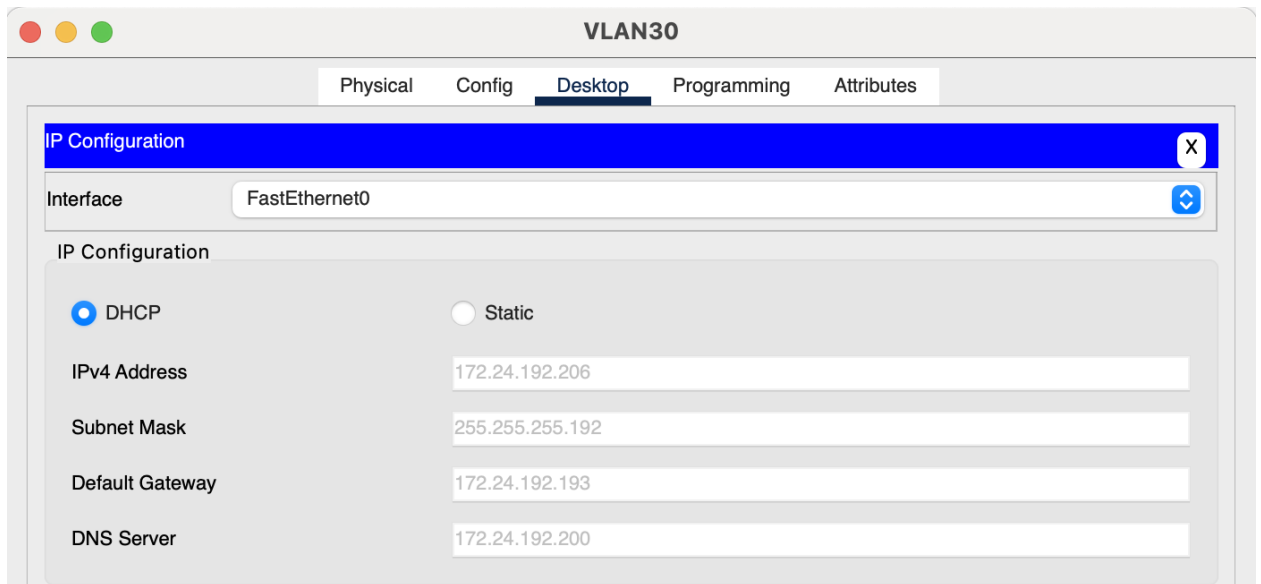
Port      Vlans allowed on trunk
Fa0/2     10,20,30-100

Port      Vlans allowed and active in management domain
Fa0/2     10,20,30,99,100

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/2     10,20,30,99,100
```

Гранкові порти

На прикладі ПК VLAN30, який знаходиться у VLAN30, перевіряємо налаштування DHCP для VLAN. Використовуємо відповідні команди для перевірки коректності автоматичного призначення IP-адрес та інших параметрів мережі (рисунок 3.24).



IP-адреса PC VLAN30

| Fire | Last Status | Source | Destination | Type | Color | Time(sec) | Periodic | Num | Edit | Delete |
|------|-------------|--------|-------------|-------|-------|-----------|----------|-----|-------|----------|
| | Failed | VLA... | VLAN30 | IC... | | 0.000 | N | 0 | (...) | (delete) |
| | Successful | VLA... | VLAN30 | IC... | | 0.000 | N | 1 | (...) | (delete) |
| | Successful | VLA... | VLAN10 | IC... | | 0.000 | N | 2 | (...) | (delete) |

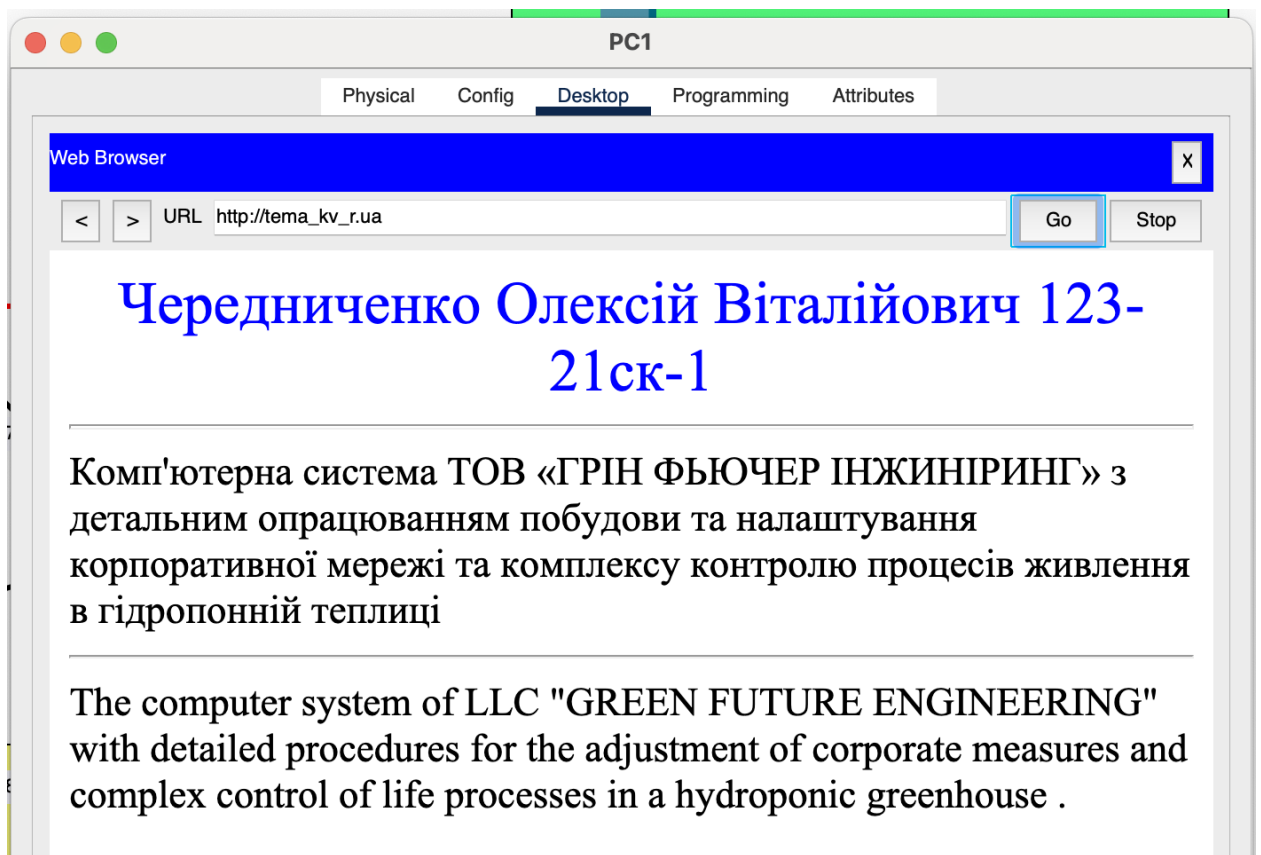
Зв'язок між VLAN42 та VLAN22

На маршрутизаторі Cherednyuchenko_R4 перевіряємо роботу NAT за допомогою команди `show ip nat translations`. Для цього відправляємо пакет від ПК з мережі LAN_1 до ПК у віддаленій мережі, а потім застосовуємо зазначену команду для перегляду трансляцій NAT (рисунок 3.26). Це дозволяє переконатися, що NAT працює правильно і виконує трансляцію адрес.

```
Cherednychenko_R4#show ip nat translations
Pro  Inside global      Inside local      Outside local     Outside global
icmp 209.165.200.5:5     172.24.194.5:5   172.24.193.6:5   172.24.193.6:5
icmp 209.165.200.6:1   172.24.192.133:1 172.24.193.6:1   172.24.193.6:1
icmp 209.165.200.6:2   172.24.192.133:2 172.24.193.6:2   172.24.193.6:2
---  209.165.200.3       172.24.194.60    ---              ---
---  209.165.200.4       172.24.192.200   ---              ---
```

Робота NAT

Перевірку доступності веб-сайту з інформацією про тему та завдання кваліфікаційної роботи студента здійснюємо на прикладі комп'ютера PC0 (рисунок 3.27).



Веб-сайт з інформацією про кваліфікаційну роботу студента доступний для перегляду

4 РОЗРОБКА КОМПОНЕНТА СИСТЕМИ

4.1 Інженерне рішення розробки гідропонної системи теплиці

Розробка гідропонної системи для теплиці вимагає інтеграції різних технологічних рішень для забезпечення ефективного контролю та управління процесами вирощування рослин. Компанія «ГРІН ФЬЮЧЕР ІНЖИРІНГ» ставить за мету створення високопродуктивної системи, яка забезпечить оптимальні умови для росту рослин.

За допомогою аналізу продуктів компанії розроблено концепт гідропонної системи теплиці. Гідропонна система складається з кількох ключових аспектів:

- система подачі поживних речовин та води до рослин;
- контроль наявності рідин у резервуарах;
- моніторинг системи за допомогою датчиків температури рідини, датчиків наявності рідин, датчики контролю кислотності, датчик контролю електропровідності питної речовини;
- надсилання даних до локального або віддаленого серверу;
- виведення поточної інформації на дисплей;
- програмне забезпечення для автоматичного управління всіма процесами в теплиці [7].

4.2 Застосовані рішення, програмне забезпечення та інженерні рішення

У якості концепції взаємодій між елементами обрано концепцію IoT.

Інтернет речей (Internet of Things, IoT) – це концепція взаємозв'язаних об'єктів, які можуть збирати, обробляти та обмінюватися даними через Інтернет. Ця технологія відкриває нові можливості для автоматизації, моніторингу та управління різноманітними системами в різних сферах, включаючи сільське господарство, промисловість, транспорт, охорону здоров'я та багато інших.

Для контролю всіх процесів, комунікації та сценаріїв обрано мову високого рівня для мікроконтролерів – MicroPython.

Мова програмування MicroPython є версією мови Python, оптимізованою для мікроконтролерів та вбудованих систем. Її основна мета полягає в наданні інструменту для розробки програмного забезпечення для малих пристроїв з обмеженими ресурсами. MicroPython включає більшість стандартних бібліотек Python, що робить його потужним інструментом для створення різноманітних проектів, від простих датчиків до складних систем автоматизації.

Однією з головних переваг MicroPython є його компактність та ефективність. Він розроблений для роботи на пристроях з обмеженою пам'яттю і процесорними ресурсами, таких як мікроконтролери серій ESP та інші. Це дозволяє використовувати MicroPython в ситуаціях, де повноцінна версія Python була б занадто ресурсомісткою.

Іншою суттєвою перевагою є зручність і простота використання. Python є відомою мовою програмування завдяки своїй читабельності та легкості у навчанні, і ці якості перенесені до MicroPython.

MicroPython також підтримує широкий спектр апаратного забезпечення та периферійних пристроїв, включаючи датчики, дисплеї, двигуни та інші компоненти. Це робить його універсальним рішенням для розробки IoT проектів.

Для передачі даних з мікроконтролера на сервер використовується python сокет-сервер.

Спочатку мікроконтролер збирає дані з підключених сенсорів або інших джерел. Потім ці дані передаються через мережевий сокет, який створює безпосереднє з'єднання між мікроконтролером та сервером. На сервері прийняті дані збережені у базі даних для подальшого використання або аналізу. Використання сокетів забезпечує ефективне і надійне з'єднання для передачі даних в реальному часі.

4.3 Розробка компонентів системи

4.3.1 Вибір мікроконтролера, датчиків та пристроїв для системи

Основним компонентом розробки системи обрано мікроконтролер Raspberry Pi Pico W за його компактність, низьке енергоспоживання та наявність вбудованого Wi-Fi модуля, що дозволяє реалізувати бездротовий моніторинг та управління. Нижче наведено опис обраних датчиків та інших компонентів системи, що використовуватимуться для побудови корпоративної мережі та контролю процесів живлення в системі гідропонної теплиці [8].

Далі обрано такі датчики та компоненти як:

1. Датчик температури DS18B20 обраний для вимірювання температури розчину завдяки його високій точності та широкому діапазону вимірювань (-55°C до +125°C). Він підтримує інтерфейс 1-Wire, що дозволяє підключати декілька датчиків до одного порту мікроконтролера, що зменшує кількість необхідних портів введення/виведення.

2. Модуль мультиплектора використовується для розширення кількості доступних входів/виходів Raspberry Pi Pico W. Це дозволяє підключити більше датчиків та інших пристроїв, ніж дозволяє обмежена кількість GPIO на мікроконтролері.

5. OLED-дисплей використовується для виведення інформації про стан системи, включаючи показники датчиків та статус системи. Дисплей забезпечує високу контрастність та чіткість зображення, що дозволяє легко зчитувати дані навіть при яскравому освітленні.

6. Годинник реального часу DS3231 обрано для забезпечення точного часу та дати. Це необхідно для синхронізації процесів у системі та ведення логів. DS3231 має високу точність завдяки вбудованому кварцовому генератору.

7. Датчик наявності води використовується для визначення наявності або відсутності води у резервуарах системи. Це важливо для запобігання пересиханню та забезпечення якісного живильного розчину.

8. ЕС-метр DFR0300 використовується для вимірювання електропровідності (ЕС) розчину в гідропонній системі. Це дозволяє контролювати концентрацію поживних речовин у воді, що є критичним параметром для забезпечення оптимальних умов росту рослин.

9. Датчик рН SEN0161-V2 використовується для вимірювання рівня рН розчину в гідропонній системі. Це важливий параметр для підтримки оптимальних умов росту рослин. Датчик забезпечує точні вимірювання та має тривалий термін служби.

10. 4-х каналне реле використовується для керування кількома пристроями одночасно. Це дозволяє централізовано керувати різними аспектами системи, такими як полив, освітлення та вентиляція.

У якості виконавчих пристроїв обрано такі компоненти як:

1. Перистальтичний насос 5В який призначений для перекачування рідин з високою точністю і контролем. Завдяки своїй конструкції, такий насос забезпечує рівномірну подачу живильного розчину без забруднення, що є важливим для гідропонних систем. Він працює від напруги 5В і має продуктивність 160 мл/хв. Перистальтичні насоси відомі своєю надійністю і здатністю перекачувати різні типи рідин, включаючи агресивні хімічні розчини, оскільки рідина не контактує з механічними частинами насоса.

Занурювальна помпа 5В пристрій, що використовуються для переміщення рідин у системах з низьким енергоспоживанням. Такі помпи можуть бути використані для циркуляції води в невеликих гідропонних системах або для подачі води в інші частини системи. Вони працюють від напруги 5В, що дозволяє їх легко інтегрувати з мікроконтролерами та іншими низьковольтними системами. Помпи 5В забезпечують надійність і довговічність, а також можуть мати різну продуктивність залежно від конкретної моделі та виробника, що дозволяє вибрати оптимальний варіант для конкретних завдань у системі гідропоніки [9].

Таблиця 4.1 – Специфікація компонентів

| № | Назва компоненту | Кількість | Входи/Виходи | Живлення |
|----|--------------------------------|-----------|-----------------|----------|
| 1 | Raspberry Pi Pico W | 1 | 26 GPIO | 5 DC |
| 2 | Датчик температури DS18B20 | 1 | 1-Wire | 3.3 DC |
| 3 | Модуль мультиплектора | 1 | I2C | 3.3 DC |
| 4 | Перистальтичний насос | 3 | постійний струм | 5 DC |
| 5 | Занурювальна помпа | 1 | постійний струм | 5 DC |
| 6 | OLED-дисплей | 1 | I2C | 3.3/5 DC |
| 7 | Годинник реального часу DS3231 | 1 | I2C | 5 DC |
| 8 | Датчик наявності рідин | 3 | Аналоговий | 3.3/5 DC |
| 9 | ЕС-метр DFR0300 | 1 | Аналоговий | 5 DC |
| 10 | Датчик рН SEN0161-V2 | 1 | Аналоговий | 5 DC |
| 11 | 4-х канальне реле | 1 | Цифровий | 3.3/ DC |

На рисунку 4.1 представлена схематична підключень елементів системи.

4.3.2 Розробка компонента керуванням станів наповнення резервуарів

Для забезпечення стабільної та ефективної роботи системи гідропонної теплиці, необхідно стабільно підтримувати наявність рідин у системі, даний компонент розроблено для сповіщення персоналу або користувача, що треба долити той чи інший компонент до системи.

Для моніторингу рівня рідини використовуються ємнісні датчики, в вагу своєї простоти та низьку вартість у порівнянні зі своїми аналогами (наприклад, ультразвукові датчики).

При виявленні недостачі рідини, виводиться оповіщення на OLED екран з інформацією та відправляються відповідні стани на сервер. У подальшому масштабуванні проекту планується виводити оповіщення push-up повідомленнями через мобільний додаток [11].

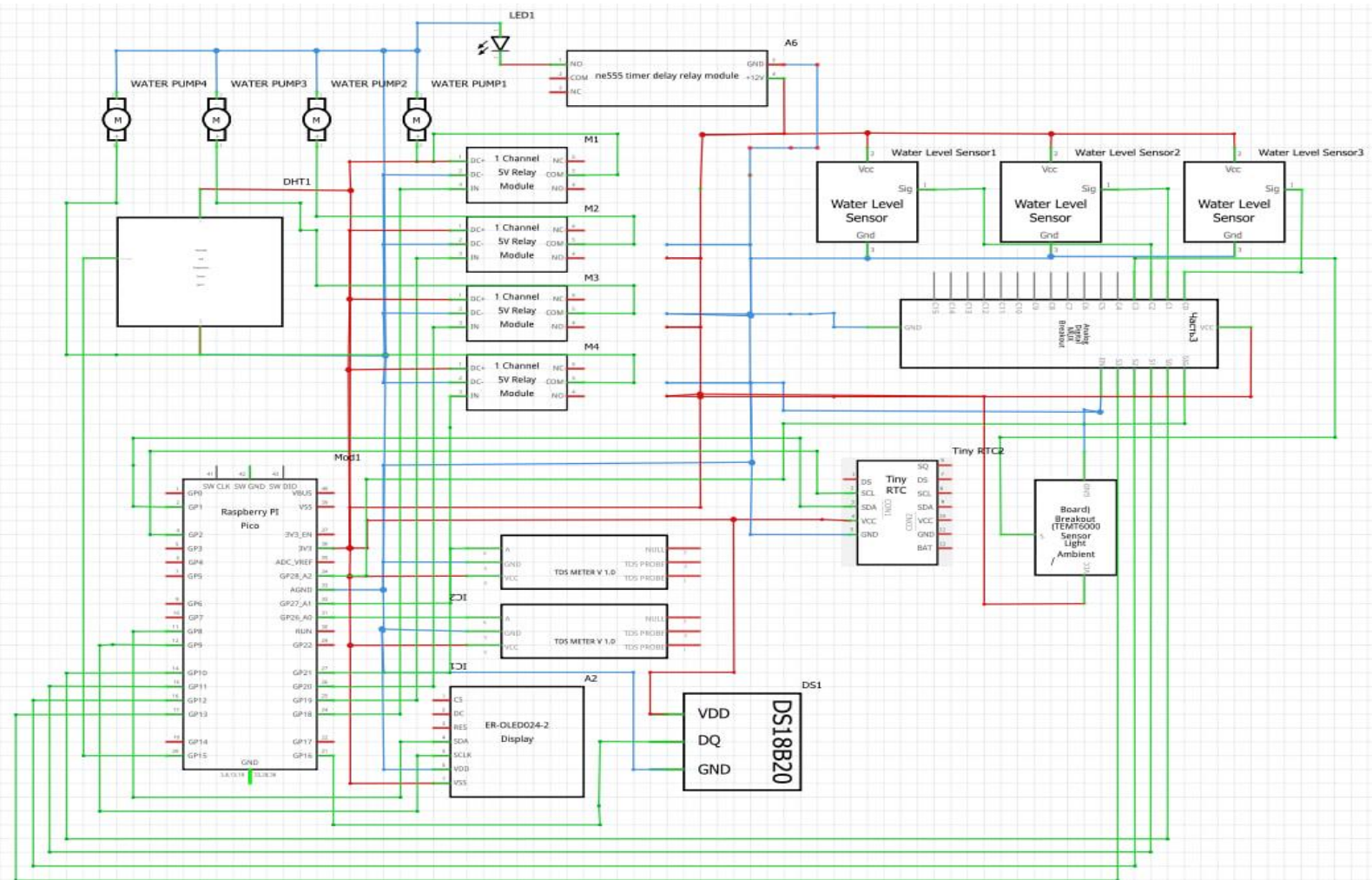


Схема підключень

Алгоритм роботи коду починається з перевірки наявності води за допомогою першого сенсора. Функція `check_water_presence1()` викликається для зчитування даних з сенсора. Якщо функція повертає `True`, це означає, що вода виявлена, і змінній `water_level1` присвоюється значення 1, а також виводиться повідомлення "Water detected!". Якщо ж функція повертає `False`, то змінній `water_level1` присвоюється значення 0, і виводиться повідомлення "No water detected."

Після цього аналогічний процес повторюється для другого та третього сенсорів. Для кожного з них викликається функція `check_water_presence1()`, яка перевіряє наявність води. Залежно від результату, відповідно встановлюються значення змінних `water_level2` та `water_level3`, і виводяться відповідні повідомлення про наявність або відсутність води.

Після зчитування даних з усіх трьох сенсорів, OLED дисплей очищається за допомогою методу `oled.fill(0)`. Потім на дисплеї відображаються повідомлення, які залежать від рівнів води, зчитаних з сенсорів. Якщо `water_level1` дорівнює 0, `water_level2` дорівнює 1, а `water_level3` дорівнює 1, на дисплеї з'являється повідомлення "Add water to the tank A". Якщо ж `water_level1` дорівнює 1, `water_level2` дорівнює 0, а `water_level3` дорівнює 1, виводиться повідомлення "Add water to the tank B". Якщо `water_level1` дорівнює 1, `water_level2` дорівнює 1, а `water_level3` дорівнює 0, з'являється повідомлення "Add water to the tank Water". Нарешті, якщо всі три змінні дорівнюють 0, відображається повідомлення "Add water all to the tanks".

Код алгоритму наведено на рисунку 4.2.

```

if check_water_presence1():
    print("Water detected!")
    water_level_1=1
else:
    print("No water detected.")
    water_level_1=0
if check_water_presence2():
    print("Water detected!")
    water_level_2=1
else:
    print("No water detected.")
    water_level_2=0
if check_water_presence3():
    print("Water detected!")
    water_level_3=1
else:
    print("No water detected.")
    water_level_3=0

oled.fill(0)
if water_level_1 == 0 and water_level_2 == 1 and water_level_3 == 1:
    oled.text('Add water to', 0, 10)
    oled.text('the tank A ', 0, 20)
elif water_level_1 == 1 and water_level_2 == 0 and water_level_3 == 1 :
    oled.text('Add water to', 0, 10)
    oled.text('the tank B', 0, 20)
elif water_level_1 == 1 and water_level_2 == 1 and water_level_3 == 0 :
    oled.text('Add water to', 0, 10)
    oled.text('the tank Water', 0, 20)
elif water_level_1 == 0 and water_level_2 == 0 and water_level_3 == 0:
    oled.text('Add water all', 0, 10)
    oled.text('to the tanks', 0, 20)

```

Приклад коду виведення інформації на екран

4.3.3 Розробка компонента підтримки живильного розчину та його процесу циркуляції

При дослідженні різних розчинів та серед живлення для рослин команда «Грін фьючер» дійшла висновку що показники для живлячого розчину є значення РН(кислотності середовища) та ЕС(електропровідності речовини).

Після аналізу їх досліджень для розробки компонента підтримки живильного розчину до проекту додано датчик РН, ЕС, датчик температури закритого типу для розрахунків компенсацій при роботі з речовиною, реле для керування системою циркуляції та 3 реле керуванням перистальтичними насосами для додавання компонентів речовини та підтримки показників розчину. Для даного проекту в якості рослини розглядається базилік, його

значення кислотності речовини мають бути у межах 6 одиниць, електропровідність між 1.6 та 2.3 mS/cm.

Алгоритм для вимірювання рН та електропровідності розчину складається з кількох етапів, починаючи з калібрування та закінчуючи обчисленням необхідних значень. Спочатку функція `read_ph` зчитує аналогове значення напруги з рН сенсора за допомогою функції `read_voltage_ph()`. Отримане значення напруги використовується у формулі $ph_value = slope * voltage_ph + intercept$ для обчислення рН. Коефіцієнти `slope` та `intercept`, необхідні для цієї формули, визначаються заздалегідь шляхом калібрування сенсора. Значення напруги при рН 7 та рН 4 встановлюються вручну (`voltage_at_ph7` та `voltage_at_ph4`), а функція `calibrate` використовує ці значення для розрахунку калібрувальних коефіцієнтів. Результат обчислення рН повертається функцією `read_ph`.

Наступний крок алгоритму включає налаштування аналогового входу для зчитування даних з сенсора електропровідності. Пін GP26 використовується як аналоговий вхід для цього сенсора. Функція `read_voltage_es` зчитує сире значення з сенсора та перетворює його в напругу за формулою $voltage_es = (val / 65535) * 3.3$. Це значення напруги необхідне для подальших обчислень електропровідності.

Перетворення напруги в електропровідність виконується функцією `voltage_to_mS_cm`, яка використовує коефіцієнт конверсії `conversion_factor = 8292` для обчислення електропровідності в мікросіменсах на сантиметр. Далі значення переводиться в мілісімени на сантиметр, ділячи на 1000.

На рисунку 4.3 зображено код програми роботи з цими компонентами [11].

```

def read_ph(slope, intercept):
    voltage_ph = read_voltage_ph()
    ph_value = slope * voltage_ph + intercept
    return ph_value

voltage_at_ph7 = 1 # Среднее значение из ваших измерений для pH 7
voltage_at_ph4 = 1.25 # Среднее значение из ваших измерений для pH 4

slope, intercept = calibrate(voltage_at_ph7, voltage_at_ph4)

sensor_ec_pin = ADC(Pin(27))

def read_voltage_ec():
    val = sensor_ec_pin.read_u16() # Чтение сырого значения
    voltage_ec = (val / 65535) * 3.3 # Преобразование в напряжение
    return voltage_ec

def voltage_to_mS_cm(voltage_ec):
    conversion_factor = 8292 #  $\mu\text{S}/\text{cm}$  per volt, полученный ранее
    conductivity_uS_cm = voltage_ec * conversion_factor
    conductivity_mS_cm = conductivity_uS_cm / 1000 # Перевод из  $\mu\text{S}/\text{cm}$  в  $\text{mS}/\text{cm}$ 
    return conductivity_mS_cm

```

Функції обробки значень

Алгоритм починається з вимірювання напруги сенсора електропровідності за допомогою функції `read_voltage_ec()`. Отримане значення напруги перетворюється в електропровідність, використовуючи функцію `voltage_to_mS_cm(voltage_ec)`, після чого результат виводиться на екран. Далі алгоритм перевіряє, чи значення електропровідності менше 1 мС/см, і в такому випадку вмикає реле на каналі 2 (`turn_relay_on(2)`). Якщо значення електропровідності більше 1.65 мС/см, алгоритм вмикає реле на каналі 3 (`turn_relay_on(3)`). В іншому випадку реле на обох каналах вимикаються (`turn_relay_off(2)` і `turn_relay_off(3)`). Після цього алгоритм зчитує значення рН, використовуючи функцію `read_ph(slope, intercept)`, і виводить його на екран разом із значенням напруги, зчитаним з рН сенсора (`read_voltage_ph()`). Потім перевіряється, чи значення рН менше 5, і в такому випадку вмикається реле на каналі 1 (`turn_relay_on(1)`). Якщо значення рН більше 7, вмикається реле на каналі 3 (`turn_relay_on(3)`). В іншому випадку обидва реле вимикаються (`turn_relay_off(1)` і `turn_relay_off(3)`).

На рисунку 4.4 Зображено фрагмент коду для тримання значень у заявлених діапазонах та ввімкнення насосів для керування рівнів значень.

```

voltage_ec = read_voltage_ec()
conductivity = voltage_to_mS_cm(voltage_ec)
print("Conductivity:", conductivity, "mS/cm")
if conductivity < 1:
    turn_relay_on(2)
elif conductivity > 1.65:
    turn_relay_on(3)
else:
    turn_relay_off(2)
    turn_relay_off(3)

ph = read_ph(slope, intercept)
print("pH value:", ph)
print("Voltage:", read_voltage_ph())

if ph < 5:
    turn_relay_on(1)
elif ph > 7:
    turn_relay_on(3)
else:
    turn_relay_off(1)
    turn_relay_off(3)

```

Керування рівнями значень

4.3.4 Розробка компонента сповіщення користувача та збереження станів

Для сповіщення користувача про стани пристроїв та аналітичні дані з датчиків, на цьому етапі задіяно Oled дисплей роздільною здатністю 128x64 пікселя. На дисплей виводяться такі дані як:

- температура живильного розчину;
- температура в системі в цілому;
- вологість у системі;
- показник значення кислотності розчину;
- показник електропровідності розчину;
- показник яскравості освітлення.

```

#
oled.text('Time: {}:{}:{}'.format(current_time[4], current_time[5], current_time[6]), 0, 0)
if water_level_1 != 0:
    oled.text('Water Temp: {:.1f}C'.format(wtemp), 0, 8)
    oled.text('Air Temp: {:.1f}C'.format(temp if temp is not None else 0), 0, 16)
    oled.text('Humidity: {:.1f}%'.format(hum if hum is not None else 0), 0, 24)
    oled.text('pH: {:.2f}'.format(ph), 0, 32)
    oled.text('EC: {:.2f} mS/cm'.format(conductivity), 0, 40)
    oled.text('light: {:.2f} lm'.format(light_level), 0, 48)
oled.show()

```

Код виведення даних на дисплей

Також розроблено функціонал виведення повідомлень при недостатці розчинів або води у додаткових контейнерах.

```

if water_level_1 == 0 and water_level_2 == 1 and water_level_3 == 1:
    oled.text('Add water to', 0, 10)
    oled.text('the tank A ', 0, 20)
elif water_level_1 == 1 and water_level_2 == 0 and water_level_3 == 1 :
    oled.text('Add water to', 0, 10)
    oled.text('the tank B', 0, 20)
elif water_level_1 == 1 and water_level_2 == 1 and water_level_3 == 0 :
    oled.text('Add water to', 0, 10)
    oled.text('the tank water', 0, 20)
elif water_level_1 == 0 and water_level_2 == 0 and water_level_3 == 0:
    oled.text('Add water all', 0, 10)
    oled.text('to the tanks', 0, 20)

```

Код виведення застережень на дисплей

Для збереження даних з датчиків та станів, розроблено елемент коду який відправляє масив даних до пайтон серверу, написаного за допомогою бібліотеки socket, він приймає масив конвертує його з бінарного вигляду та зберігаю у базу даних.

Він працює так: створюється сокет, який прив'язується до всіх доступних інтерфейсів на порту 12345 і починає прослуховування вхідних підключень. Потім здійснюється підключення до бази даних SQLite та створюється таблиця data, якщо вона ще не існує. Таблиця містить стовпці для різних параметрів, таких як рівень води, температура, вологість, рН, електропровідність тощо. Основний цикл сервера чекає на підключення клієнтів. Після встановлення підключення з клієнтом дані приймаються в буфер розміром 1024 байти. Прийняті дані розпаковуються відповідно до визначеного формату, що включає вісім цілих чисел та шість чисел з плаваючою точкою, а також п'ять цілих чисел. Розпаковані дані вставляються в таблицю data в базі даних. Якщо прийом даних припиняється або клієнт відключається, сервер закриває з'єднання з клієнтом і повертається до очікування нових підключень. При натисканні комбінації клавіш для завершення роботи (наприклад, Ctrl+C), сервер зупиняється, закривається

сокет і з'єднання з базою даних, що забезпечує коректне завершення роботи програми.

При написанні серверу, враховано перевірка на наявність .db файлів, при сценарії де файл не знайдено у локальній папці сервера, створюється новий файл з таблицею за критеріями відправляємих даних.

Щоб мікроконтролер мав змогу відправити дані на локальний сервер, до коду додано функцію підключення до бездротової мережі так як на борту мікроконтролера є Wi-Fi модуль.

Функція `connect_to_wifi` встановлює з'єднання з Wi-Fi мережею за допомогою вказаних SSID та паролю. Спочатку активується інтерфейс Wi-Fi у режимі станції. Якщо пристрій не підключений до мережі, виводиться повідомлення "Connecting to network..." і виконується спроба підключення до вказаної мережі. Функція очікує, поки з'єднання не буде встановлено, у циклі, що безперервно перевіряє стан підключення. Після успішного підключення виводиться конфігурація мережі. Нарешті, виклик функції `connect_to_wifi` з конкретними SSID та паролем встановлює з'єднання з мережею (рисунок 4.9).

```
def connect_to_wifi(ssid, password): #network connecting
    wlan = network.WLAN(network.STA_IF)
    wlan.active(True)
    if not wlan.isconnected():
        print('Connecting to network...')
        wlan.connect(ssid, password)
        while not wlan.isconnected():
            pass
    print('Network config:', wlan.ifconfig())

# change ssid and password, after
connect_to_wifi('Kak_ugodnoGG', '0636381045')
```

■ Функція підключення до бездротової мережі

Далі для відправлення масиву до серверу перевіряємо поточний час, і якщо хвилини кратні 5, а секунди не кратні 10, відбувається спроба підключення до сервера для передачі даних. Якщо значення температури та вологості не є None, створюється сокет для з'єднання з сервером за вказаними IP та портом. Після успішного підключення формується масив даних, що

включає поточний час та виміряні значення (рівень води, температуру води, температуру, вологість, рН, електропровідність, рівень освітленості та стан реле). Масив даних конвертується в байтовий формат за допомогою `struct.pack` і відправляється на сервер через сокет. Якщо значення температури або вологості відсутні, виводиться повідомлення про невдачу відправки даних. У випадку виникнення виключення, виводиться відповідне повідомлення про помилку. Після завершення операцій сокет закривається. Приклад відправлення наведено на рисунку 4.8.

```
if current_time[5] % 5 == 0 and current_time[6] % 10:
    try:
        if temp is not None and hum is not None:
            socket = usocket.socket(usocket.AF_INET, usocket.SOCK_STREAM)
            socket.connect((server_ip, server_port))
            print('Успешное подключение к серверу')

            # Пример массива данных
            data = [current_time[0], current_time[1], current_time[2], current_time[4], current_time[5], water_level_1, 1, 1, wtemp, t

            # Конвертация данных в байты
            bytes_data = struct.pack('8i6f5i', *data)

            # Отправка данных
            socket.send(bytes_data)
            print('Данные отправлены')
        else:
            print("send data faild")
    except Exception as e:
        print('Ошибка подключения или отправки данных:', e)

finally:
    # Закрытие сокета
    socket.close()
    print('Соединение закрыто')
```

Відправка масиву до серверу за допомогою бібліотеки `usocket`

4.3.5 Розробка допоміжних компонентів

Для керування системою у визначений проміжок часу для дотримання всіх сценаріїв які зав'язані на часі використовується шилд плата годинника реального часу DS3231 яка має автономне живлення від батарейки 3.3 DC. За допомогою автономного живлення маємо завжди доступ до точної дати та часу.

Час у системі впливає на такі функції як:

- погодинне освітлення рослин;
- перегін живильного розчину у системи у якийсь заданий проміжок часу;
- вимірювання датчиків через кожні 2-10 хвилин;

– відправлення даних до серверу кожні 5 хвилин.

Клас DS1307 використовується для взаємодії з годинником реального часу (RTC) через інтерфейс I2C. При ініціалізації об'єкта класу, метод `__init__` приймає об'єкт I2C та адресу пристрою (за замовчуванням 0x68). Метод `set_time` встановлює час на RTC, перетворюючи десятичні значення року, місяця, дня, дня тижня, години, хвилини та секунди в формат BCD (Binary-Coded Decimal) і записуючи їх до відповідних регістрів пристрою. Перетворення здійснюється за допомогою лямбда-функції `bcd`, яка додає до числа 6 разів частку від його ділення на 10. Метод `read_time` читає 7 байт даних з RTC, починаючи з регістра 0x00, і перетворює отримані значення з формату BCD у десяткові числа за допомогою лямбда-функції `bcd_to_dec`, яка виконує зворотнє перетворення. Результатом роботи методу є кортеж, що містить рік, місяць, день, день тижня, годину, хвилину та секунду.

Приклад коду наведено на рисунку 4.9.

```
def __init__(self, i2c_time, addr=0x68):
    self.i2c = i2c_time
    self.addr = addr

def set_time(self, year, month, day, weekday, hour, minute, second):
    # Преобразуем десятичные числа в BCD
    bcd = lambda n: n + 6 * (n // 10)
    self.i2c.writeto_mem(self.addr, 0x00, bytes([bcd(second), bcd(minute), bcd(hour), bcd(weekday), bcd(day), bcd(month), bcd(year - 2000)]))

def read_time(self):
    # Читаем 7 байт времени начиная с регистра 0x00
    data = self.i2c.readfrom_mem(self.addr, 0x00, 7)
    # Преобразуем BCD в десятичные числа
    bcd_to_dec = lambda n: n - 6 * (n >> 4)
    return (
        bcd_to_dec(data[6]) + 2000, # Год
        bcd_to_dec(data[5]),      # Месяц
        bcd_to_dec(data[4]),      # День
        bcd_to_dec(data[3]),      # День недели
        bcd_to_dec(data[2]),      # Час
        bcd_to_dec(data[1]),      # Минута
        bcd_to_dec(data[0])       # Секунда
    )
```

Функції роботи з годинником

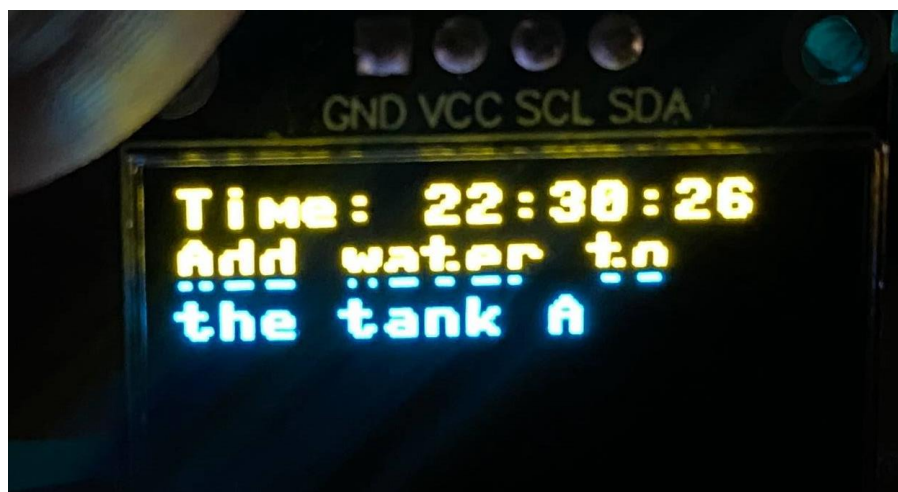
4.4 Тестування системи

Першим етапом тестуємо виведення коректної інформацію з датчикі на дисплей(рисунок 4.12).



Виведення поточних станів датчиків системи та дати

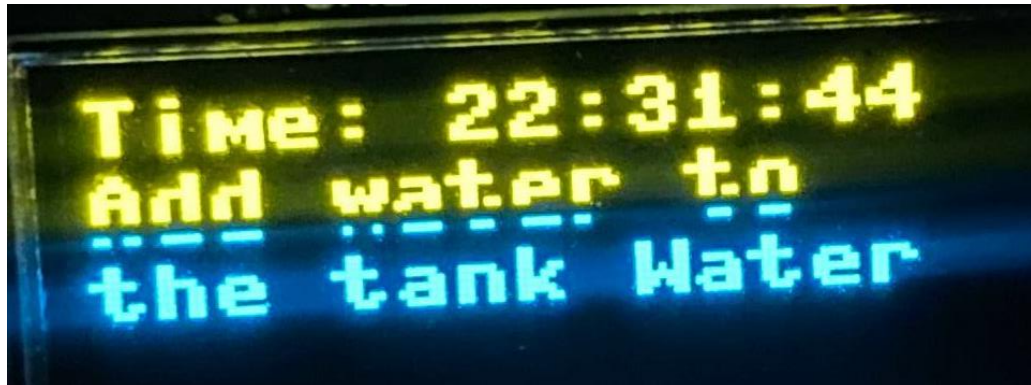
Для перевірки системи при відсутності рідин у резервуарах переміщено сенсори з води до сухої середи та отримано повідомлення на дисплей(рисунок 4.13-16).



Виведення повідомлення про закінчення компоненту А



Виведення повідомлення про закінчення компоненти В



Виведення повідомлення про закінчення води у додатковому баку



Виведення повідомлення про закінчення рідин у всіх баках

```

Текущее время: 2023-6-4 2, 22:7:17
Conductivity: 0.3607561 mS/cm
pH value: 12.78219
Voltage: 0.5117052
Temperature: 26.6
Humidity: 34.0
Текущее время: 2023-6-4 2, 22:7:18
Conductivity: 0.4008401 mS/cm
pH value: 12.87887
Voltage: 0.4980087
Temperature: 26.6
Humidity: 35.0
  
```

Виведення станів та інформації у com порт при роботі з мікроконтролером

Для перевірки роботи сервера та збереження даних використано фрагмент для роботи з сервером який наведено на рисунку 4.10. Після вдалої відправки перевіряємо дані у базі даних(рисунок 4.18).

| | year | month | day | hours | minutes | Waterlevel_A | Waterlevel_B | Waterlevel | Water_temp | temp | hum | Ph | Ec | Light_level | relay_1 | relay_2 | relay_3 |
|----|---------|---------|---------|---------|---------|--------------|--------------|------------|------------|------------------|---------|-------------------|-------------------|------------------|---------|---------|---------|
| | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... | Филь... |
| 1 | 2023 | 6 | 4 | 21 | 44 | 896 | 1 | 1 | 27.75 | 27.6000003814697 | 40.0 | 18.5649337768555 | 0.427562773227692 | 7.81261920928955 | 1 | 1 | 1 |
| 2 | 2023 | 6 | 4 | 21 | 44 | 848 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.4779224395752 | 0.387478768825531 | 6.10360860824585 | 1 | 1 | 1 |
| 3 | 2023 | 6 | 4 | 21 | 44 | 864 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.4392490386963 | 0.187058702111244 | 6.83604192733765 | 1 | 1 | 1 |
| 4 | 2023 | 6 | 4 | 21 | 44 | 960 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.5069255828857 | 0.447604775428772 | 6.59189748764038 | 1 | 1 | 1 |
| 5 | 2023 | 6 | 4 | 21 | 44 | 880 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.6616153717041 | 0.400840073823929 | 5.12703132629395 | 1 | 1 | 1 |
| 6 | 2023 | 6 | 4 | 21 | 44 | 864 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.57460212270752 | 0.354075402021408 | 4.88288686859668 | 1 | 1 | 1 |
| 7 | 2023 | 6 | 4 | 21 | 44 | 1056 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.6326122283936 | 0.374117404222488 | 8.78919696807861 | 1 | 1 | 1 |
| 8 | 2023 | 6 | 4 | 21 | 44 | 896 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.5359306335449 | 0.587898790836334 | 7.81261920928955 | 1 | 1 | 1 |
| 9 | 2023 | 6 | 4 | 21 | 44 | 944 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.6036071777344 | 0.260546088218689 | 7.81261920928955 | 1 | 1 | 1 |
| 10 | 2023 | 6 | 4 | 21 | 44 | 1024 | 1 | 1 | 27.75 | 27.5 | 39.0 | 18.4875907897949 | 0.414201408624649 | 8.05676364898682 | 1 | 1 | 1 |
| 11 | 2023 | 6 | 4 | 21 | 44 | 832 | 1 | 1 | 27.75 | 27.5 | 39.0 | 18.552673339844 | 0.420882105827332 | 7.08018636703491 | 1 | 1 | 1 |
| 12 | 2023 | 6 | 4 | 21 | 44 | 896 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.5165939331055 | 0.374117404222488 | 9.7657737319336 | 1 | 1 | 1 |
| 13 | 2023 | 6 | 4 | 21 | 44 | 912 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.2265510559082 | 0.287268728017807 | 7.56847476959229 | 1 | 1 | 1 |
| 14 | 2023 | 6 | 4 | 21 | 44 | 848 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.4875907897949 | 0.32067209482193 | 7.81261920928955 | 1 | 1 | 1 |
| 15 | 2023 | 6 | 4 | 21 | 44 | 848 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.552673339844 | 0.340714067220688 | 11.4747848510742 | 1 | 1 | 1 |
| 16 | 2023 | 6 | 4 | 21 | 44 | 896 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.7389602661133 | 0.327352762222229 | 8.78919696807861 | 1 | 1 | 1 |
| 17 | 2023 | 6 | 4 | 21 | 44 | 864 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.6132755279541 | 0.481008112430573 | 7.56847476959229 | 1 | 1 | 1 |
| 18 | 2023 | 6 | 4 | 21 | 44 | 848 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.7099571228027 | 0.394159436225891 | 7.32433032989502 | 1 | 1 | 1 |
| 19 | 2023 | 6 | 4 | 21 | 44 | 672 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.7099571228027 | 0.514411449432373 | 8.30090808868408 | 1 | 1 | 1 |
| 20 | 2023 | 6 | 4 | 21 | 44 | 832 | 1 | 1 | 27.75 | 27.6000003814697 | 39.0 | 18.6132755279541 | 0.394159436225891 | 10.9864959716797 | 1 | 1 | 1 |
| 21 | 2023 | 6 | 4 | 21 | 44 | 912 | 1 | 1 | 27.75 | 27.5 | 39.0 | 18.5649337768555 | 0.387478768825531 | 7.32433032989502 | 1 | 1 | 1 |
| 22 | 2023 | 6 | 4 | 21 | 44 | 848 | 1 | 1 | 27.75 | 27.5 | 39.0 | 18.4585857391357 | 0.193739384412766 | 8.05676364898682 | 1 | 1 | 1 |

Вміст бази даних після кількох спрацювань відправки

ВИСНОВКИ

У даній роботі було проведено ґрунтовний аналіз компанії «Грін Фьючер Інжирінг» та розглянуто побудову, налаштування та побудовано макет майбутньої корпоративної мережі компанії. Також були розглянуті вимоги до мережі та специфікації для системи контролю процесів живлення в гідропонній теплиці. Проведений детальний огляд об'єкта розробки включав визначення вимог до комп'ютерної системи та системи гідропонної теплиці. Відповідно до цих вимог було підібрано необхідне обладнання. Розроблено систему контролю процесів живлення в гідропонній теплиці яка включає в себе модулі для вимірювання та регулювання рівня води, температури, вологості, рН та електропровідності розчину, що дозволяє забезпечити оптимальні умови які враховує особливості росту такої рослини як базилік.. Автоматизація цих процесів дозволяє знизити людський фактор і забезпечити безперервний моніторинг та управління параметрами в реальному часі.

Робота передбачає впровадження інноваційних рішень для систем. У процесі аналізу корпоративної були враховані потреби компанії у масштабуванні та подальшому розвитку системи, що дозволить ефективно адаптуватися до швидких змін у технологічному середовищі. Застосування передових технологій та врахування найкращих практик в галузі інформаційної безпеки забезпечує стійкість та надійність системи.

В ході виконання роботи були розглянуті можливості масштабування та розвитку системи, що дозволяє врахувати потреби компанії у майбутньому. Використання сучасних технологій та найкращих практик інформаційної безпеки гарантує надійність та стійкість розробленої системи.

Та розроблена система контролю процесів живлення в гідропонній теплиці повністю відповідає сучасним вимогам та забезпечує високу ефективність і надійність управління виробничими процесами.

Робота була виконана відповідно до визначеної тематики та вимог, заданих завдань, при цьому оформлення документації відповідає усім стандартам та рекомендаціям, що передбачені методичними вказівками.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Сайт компанії «ГРІН ФЬЮЧЕР ІНЖИРІНГ» – Education [Електронний ресурс] – Режим доступу до ресурсу: <https://greenfuture.com.ua/> (дата звернення 01.05.2024р.)
2. Організаційний менеджмент. Лекція з курсу "Менеджмент організацій" для студентів спеціальності 073 "Менеджмент". / Павленчик А. О. – Львівський державний університет фізичної культури імені Івана Боберського, 2020. – 16 с
3. Методичні рекомендації до виконання кваліфікаційної роботи бакалавра студентами галузі знань 12 Інформаційні технології спеціальності 123 Комп'ютерна інженерія / Л.І. Цвіркун, С.М. Ткаченко, Я.В. Панферова, Д.О. Бешта, Л.В. Бешта. – Д.: НТУ «ДП», 2024. – 62 с.
4. NAT для корпоративних мереж – Education [Електронний ресурс] – Режим доступу до ресурсу: <https://www.coursehero.com/> (дата звернення 10.05.2024р.)
5. VLAN для корпоративних мереж – Education [Електронний ресурс] – Режим доступу до ресурсу: <https://www.coursehero.com/> (дата звернення 12.05.2024р.)
6. ACL списки для корпоративних мереж – Education [Електронний ресурс] – Режим доступу до ресурсу: <https://www.coursehero.com/> (дата звернення 14.05.2024р.)
7. VPN для корпоративних мереж – Education [Електронний ресурс] – Режим доступу до ресурсу: <https://www.coursehero.com/> (дата звернення 25.05.2024р.)
8. Інтернет речей – Education [Електронний ресурс] – Режим доступу до ресурсу: <http://surl.li/dcsqu> (дата звернення 17.05.2024р.)
9. Офіційна сторінка raspberrypi – [Електронний ресурс] – Режим доступу до ресурсу: www.raspberrypi.com/documentation (дата звернення 20.05.2024р.)

10. Датчики та їх компоненти – hwlibre [Електронний ресурс] – Режим доступу до ресурсу: <https://www.hwlibre.com/> (дата звернення 23.05.2024р.)

11. Компоненти кіберфізичних систем – dfrobot [Електронний ресурс] – Режим доступу до ресурсу: <https://edu.dfrobot.com/> (дата звернення 25.05.2024р.)

12. Мова програмування microPython – Education [Електронний ресурс] – Режим доступу до ресурсу: <https://micropython.org/> (дата звернення 26.05.2024р.)

Додаток А

Текст main блоку обробки з мікроконтролера Raspberry Pi

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“ДНІПРОВСЬКА ПОЛІТЕХНІКА”

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
НАЛАШТУВАННЯ МІКРОКОНТРОЛЛЕРА RASPBERRY PI PICO W

Текст програми контролера Raspberry Pi pico W

804.02070743.24018-01 12 01

Листів 9

АНОТАЦІЯ

Дана програма розроблена для автоматизації контролю процесів живлення в гідропонній теплиці за допомогою різних сенсорів і компонентів.

Вона підключається до Wi-Fi мережі, зчитує дані з датчиків температури, вологості, рівня води, рН, електропровідності та рівня освітленості, і відображає ці дані на OLED дисплеї. Крім того, програма відправляє зібрані дані на сервер через сокетне з'єднання для подальшого аналізу і зберігання.

Система також керує реле для включення та виключення різних пристроїв в залежності від отриманих значень з датчиків, забезпечуючи автоматизацію і оптимізацію умов для росту рослин у гідропонній теплиці.

ЗМІСТ

| | |
|--------------------|---|
| 1. Блок main | 4 |
|--------------------|---|

main.py

```

from machine import Pin, I2C, ADC, RTC
import utime
from dht import DHT11
import time
import ssl306
from ssl306 import SSD1306_I2C
import framebuffer
import urequests
import network
import rp2
import sys
import socket
import struct
import onewire
import ds18x20
import usocket

server_ip = '192.168.31.33'
server_port = 12345

dat_pin = machine.Pin(15)
ds_sensor = ds18x20.DS18X20(onewire.OneWire(dat_pin))

roms = ds_sensor.scan()
print('Found DS18B20 sensors:', roms)
multi = ADC(Pin(28))
control_pins = [Pin(i, Pin.OUT) for i in (10, 11, 12, 13)]

i2c_time = I2C(1, scl = Pin(3), sda = Pin(2))
class DS1307:
    def __init__(self, i2c_time, addr=0x68):
        self.i2c = i2c_time
        self.addr = addr

    def set_time(self, year, month, day, weekday, hour, minute, second):
        bcd = lambda n: n + 6 * (n // 10)
        self.i2c.writeto_mem(self.addr, 0x00, bytes([bcd(second),
bcd(minute), bcd(hour), bcd(weekday), bcd(day), bcd(month), bcd(year -
2000)]))

    def read_time(self):
        data = self.i2c.readfrom_mem(self.addr, 0x00, 7)

        bcd_to_dec = lambda n: n - 6 * (n >> 4)
        return (
            bcd_to_dec(data[6]) + 2000,
            bcd_to_dec(data[5]),
            bcd_to_dec(data[4]),
            bcd_to_dec(data[3]),
            bcd_to_dec(data[2]),
            bcd_to_dec(data[1]),
            bcd_to_dec(data[0])
        )

#pin for connection DHT11 sensor
pin = Pin(14, Pin.OUT, Pin.PULL_DOWN)
sensor = DHT11(pin)

#setings sda and scl for I2C interface
#i2c = I2C(0, scl=Pin(9), sda=Pin(8), freq=400000)

```

```

#settings for relay pins
relay_pins = [
    Pin(16, Pin.OUT), #like pin number 1
    Pin(17, Pin.OUT), # Like pin number 2
    Pin(18, Pin.OUT), # Like pin number 3
    Pin(19, Pin.OUT) # Like pin number 4
]

for i in range(0, 4):
    print("Humidity: {}".format(i))
    relay_pins[i].value(1)
rtc = DS1307(i2c_time)
#display settings
#WIDTH = 128
#HEIGHT = 32
i2c = I2C(0, scl=Pin(9), sda=Pin(8), freq=400000)
#display = SSD1306_I2C(WIDTH, HEIGHT, i2c)
#Create an object for work with Oled display
oled_width = 128
oled_height = 64
oled = ssd1306.SSD1306_I2C(128, 64, i2c, addr=0x3C)

#water sensor settings like water constant
water_level_sensor = ADC(Pin(26))
water=0
compA_level_sensor = ADC(Pin(27))
compA=0
compB_level_sensor = ADC(Pin(28))
compB=0
# array for animations
images = []
def connect_to_wifi(ssid, password): #network connecting
    wlan = network.WLAN(network.STA_IF)
    wlan.active(True)
    if not wlan.isconnected():
        print('Connecting to network...')
        wlan.connect(ssid, password)
        while not wlan.isconnected():
            pass
    print('Network config:', wlan.ifconfig())

# change ssid and password, after
connect_to_wifi('Kak_ugodnoGG', '0636381045')

for n in range(1, 28):
    with open('/folder_anima/image%s.pbm' % n, 'rb') as f: #open folder
and image
        f.readline() # Magic number
        f.readline() # Creator comment
        f.readline() # Dimensions
        data = bytearray(f.read())
        fbuf = framebuf.FrameBuffer(data, 30, 30, framebuf.MONO_HLSB)
#adjust accordingly the width and height
        images.append(fbuf)

count=0
while count<6:
    for i in images:
        oled.blit(i, 0, 0)
        oled.blit(i, 32, 0)
        oled.blit(i, 64, 0)
        oled.blit(i, 96, 0)

```



```

        oled.show()
        time.sleep(0.01)
        count=count+1
        print(count)

#for pin_num in relay_pins:
#    pin = Pin(pin_num, Pin.OUT)

#
def display_text(text):
    oled.fill(0) # Очистка дисплея
    oled.text(text, 0, 0) # Вывод текста в левом верхнем углу
    oled.show() # Отображение на дисплее

def turn_relay_on(channel):
    relay_pins[channel].value(0)

def turn_relay_off(channel):
    relay_pins[channel].value(1)

def check_water_presence(threshold=30000):
    sensor_value = water_level_sensor.read_u16()
    if sensor_value > threshold:
        return False # Вода обнаружена
    else:
        return True

def measure_temperature():
    try:
        temperature = sensor.temperature
        humidity = sensor.humidity
        print("Temperature: {}".format(temperature))
        print("Humidity: {}".format(humidity))
        #display_text("Temperature: {:.1f}C\nHumidity:
{:.1f}%".format(temperature, humidity))
        return temperature, humidity
    except Exception as e:
        print("Error:", e)
        return None, None

sensor_ph_pin = ADC(Pin(26))

def read_voltage_ph():
    analog_value = sensor_ph_pin.read_u16()
    voltage_ph = analog_value * 3.3 / 65535
    return voltage_ph

def calibrate(voltage_at_ph7, voltage_at_ph4):
    slope = (7 - 4) / (voltage_at_ph7 - voltage_at_ph4)
    intercept = 7 - slope * voltage_at_ph7
    return slope, intercept

def read_ph(slope, intercept):
    voltage_ph = read_voltage_ph()
    ph_value = slope * voltage_ph + intercept
    return ph_value

voltage_at_ph7 = 1
voltage_at_ph4 = 1.25

slope, intercept = calibrate(voltage_at_ph7, voltage_at_ph4)

sensor_ec_pin = ADC(Pin(27))

```

```

def read_voltage_ec():
    val = sensor_ec_pin.read_u16() # Чтение сырого значения
    voltage_ec = (val / 65535) * 3.3 # Преобразование в напряжение
    return voltage_ec

def voltage_to_mS_cm(voltage_ec):
    conversion_factor = 8292 # µS/cm per volt, полученный ранее
    conductivity_uS_cm = voltage_ec * conversion_factor
    conductivity_mS_cm = conductivity_uS_cm / 1000 # Перевод из µS/cm в
mS/cm
    return conductivity_mS_cm
def read_wtemperature():
    ds_sensor.convert_temp()
    time.sleep_ms(750) # DS18B20 требует некоторое время для
преобразования температуры
    for rom in roms:
        wtemp = ds_sensor.read_temp(rom)
        print('Temperature:', ds_sensor.read_temp(rom), '°C')
    return wtemp

def select_channel(channel):
    for i in range(4):
        control_pins[i].value((channel >> i) & 1)

def read_sensor(channel):
    select_channel(channel)
    time.sleep(0.1) # Пауза для стабилизации сигнала
    return multi.read_u16()

def convert_to_lumens(adc_value):
    калибровки для точности
    max_adc = 65535
    max_lumens = 1000
    lumens = (adc_value / max_adc) * max_lumens
    return lumens

def check_water_presence1(threshold=15000):
    sensor_value = read_sensor(4)
    if sensor_value < threshold:
        return False
    else:
        return True

wtemp = read_wtemperature()
measure_temperature()

while True:
    temp,hum = measure_temperature()

    current_time = rtc.read_time()
    print(": {}-{}-{} {}, {}:{}".format(current_time[0],
current_time[1], current_time[2], current_time[3], current_time[4],
current_time[5], current_time[6]))
    utime.sleep(1)

```

```

if 6 <= current_time[4] < 18:
    turn_relay_on(0)
else:
    turn_relay_off(0)
#if check_water_presence():
#    print("Water detected!")
#    water=1
#else:
#    print("No water detected.")
#    water=0
voltage_ec = read_voltage_ec()
conductivity = voltage_to_mS_cm(voltage_ec)
print("Conductivity:", conductivity, "mS/cm")
if conductivity < 1:
    turn_relay_on(2)
elif conductivity > 1.65:
    turn_relay_on(3)
else:
    turn_relay_off(2)
    turn_relay_off(3)

ph = read_ph(slope, intercept)
print("pH value:", ph)
print("Voltage:", read_voltage_ph())

if ph < 5:
    turn_relay_on(1)
elif ph > 7:
    turn_relay_on(3)
else:
    turn_relay_off(1)
    turn_relay_off(3)

adc_value = read_sensor(0)
light_level = convert_to_lumens(adc_value)
if check_water_presencel():
    print("Water detected!")
    water=1
else:
    print("No water detected.")
    water=0
water_level_1 = read_sensor(4)
#water_level_2 = read_sensor(3)
#water_level_3 = read_sensor(2)
if current_time[5] % 1 == 0 and current_time[6] % 40 == 0:
    wtemp = read_wtemperature()

oled.fill(0)
if water_level_1 == 0 and water_level_2 == 1 and water_level_3 ==
1:
    oled.text('Add water to', 0, 10)
    oled.text('the tank A ', 0, 20)
elif water_level_1 == 1 and water_level_2 == 0 and water_level_3 ==
1 :
    oled.text('Add water to', 0, 10)
    oled.text('the tank B', 0, 20)
elif water_level_1 == 1 and water_level_2 == 1 and water_level_3 ==
0 :
    oled.text('Add water to', 0, 10)
    oled.text('the tank Water', 0, 20)
elif water_level_1 == 0 and water_level_2 == 0 and water_level_3 ==
0:

```

```

oled.text('Add water all', 0, 10)
oled.text('to the tanks', 0, 20)

oled.text('Time: {}:{}:{}'.format(current_time[4], current_time[5],
current_time[6]), 0, 0)
if water_level_1 != 0:
    oled.text('Water Temp: {:.1f}C'.format(wtemp), 0, 8)
    oled.text('Air Temp: {:.1f}C'.format(temp if temp is not None
else 0), 0, 16)
    oled.text('Humidity: {:.1f}%'.format(hum if hum is not None else
0), 0, 24)
    oled.text('pH: {:.2f}'.format(ph), 0, 32)
    oled.text('EC: {:.2f} mS/cm'.format(conductivity), 0, 40)
    oled.text('light: {:.2f} lm'.format(light_level), 0, 48)
oled.show()
if current_time[5] % 5 == 0 and current_time[6] % 10:
    try:
        if temp is not None and hum is not None:
            socket = usocket.socket(usocket.AF_INET,
usocket.SOCK_STREAM)
            socket.connect((server_ip, server_port))

            data = [current_time[0], current_time[1],
current_time[2], current_time[4], current_time[5], water_level_1, 1, 1,
wtemp, temp, hum, ph, conductivity, light_level, 1, 1, 1, 1, 1]

            bytes_data = struct.pack('8i6f5i', *data)

            socket.send(bytes_data)
            print('Данные отправлены')
        else:
            print("send data faild")
    except Exception as e:

finally:
    socket.close()

```

Додаток Б

Текст програми налаштування мережного обладнання

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“ДНІПРОВСЬКА ПОЛІТЕХНІКА”

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
НАЛАШТУВАННЯ МЕРЕЖІ КОМП'ЮТЕРНОЇ СИСТЕМИ

Текст програми налаштування мережного обладнання

804.02070743.24018-01 12 01

Листів 13

АНОТАЦІЯ

Дана програма включає набір команд, розроблених спеціально для конфігурації маршрутизаторів та комутаторів, які використовуються в структурі корпоративної мережі. Вона охоплює широкий спектр налаштувань, включаючи присвоєння IP-адрес, базову конфігурацію мережевих пристроїв, а також деталізоване налаштування ключових мережевих сервісів та протоколів.

В рамках програми передбачено налаштування DHCP для автоматичного розподілу IP-адрес, NAT для трансляції адрес в мережі з метою забезпечення безпечного доступу до інтернету, а також VPN для створення захищених віртуальних приватних мереж. Також в програмі враховано налаштування системи AAA для забезпечення аутентифікації, авторизації та обліку діяльності користувачів, а протокол OSPF використовується для динамічної маршрутизації в масштабованих мережах.

Крім того, програма включає в себе налаштування VLAN для ефективної сегментації мережі, налаштування статичних маршрутів для контролю маршрутизації трафіку, EtherChannel для збільшення пропускну здатності та забезпечення відмовостійкості з'єднань, а також заходи безпеки портів для захисту від несанкціонованого доступу.

ЗМІСТ

| | |
|---|----|
| 1. Конфігурація Cherednychenko_R4 | 4 |
| 2. Конфігурація Cherednychenko_R3 | 7 |
| 3. Конфігурація Cherednychenko_R2 | 10 |
| 4. Конфігурація Cherednychenko_SW1 etherchannel | 12 |
| 4. Конфігурація Cherednychenko_SW4 VLAN | 15 |

1. Конфігурація Cherednychenko_R4

```
Current configuration : 2957 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname Cherednychenko_R4
enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCil
!ip dhcp excluded-address 172.24.194.29 172.24.194.135
ip dhcp excluded-address 172.24.194.254
ip dhcp excluded-address 172.24.194.255
!
ip dhcp pool LAN-2
network 172.24.193.128 255.255.255.128
default-router 172.24.193.129
dns-server 172.24.192.200
aaa new-model
!
aaa authentication login console group radius local
aaa authentication login default local
no ip cef
no ipv6 cef
username CherednychenkoOlV password 7 082048430017544541
username Cherednychenko_R1 password 7 082048430017544541
license udi pid CISCO2911/K9 sn FTX1524HI7L-
ip domain-name Cherednychenko_R4
spanning-tree mode pvst
interface GigabitEthernet0/0
ip address 172.10.18.1 255.255.255.252
ip nat inside
duplex auto
speed auto
!
interface GigabitEthernet0/1
ip address 172.24.193.129 255.255.255.128
ip nat inside
duplex auto
speed auto
!
interface GigabitEthernet0/2
ip address 172.10.18.5 255.255.255.252
ip nat inside
duplex auto
speed auto
!
interface Serial0/3/0
ip address 172.10.18.14 255.255.255.252
ip nat inside
!
interface Serial0/3/1
ip address 209.165.202.1 255.255.255.252
ip nat outside
clock rate 128000
!
interface Vlan1
no ip address
shutdown
!
router ospf 1
log-adjacency-changes
passive-interface default
```

```

no passive-interface GigabitEthernet0/0
no passive-interface GigabitEthernet0/1
no passive-interface GigabitEthernet0/2
no passive-interface Serial0/3/0
no passive-interface Serial0/3/1
auto-cost reference-bandwidth 1000
network 172.24.200.0 0.0.0.3 area 0
network 172.24.200.4 0.0.0.3 area 0
network 172.24.200.12 0.0.0.3 area 0
network 172.24.200.16 0.0.0.3 area 0
network 172.24.193.128 0.0.0.127 area 0
default-information originate
!
ip nat pool Internet 209.165.200.5 209.165.200.30 netmask
255.255.255.224
ip nat inside source list NAT12 pool Internet
ip nat inside source static 172.24.192.200 209.165.200.4
ip nat inside source static 172.24.194.60 209.165.200.3
ip classless
ip route 0.0.0.0 0.0.0.0 Serial0/3/1
!
ip flow-export version 9
ip access-list extended NAT12
permit ip 172.24.0.0 0.0.255.255 any
!
banner motd ^CCherednychenko_R4^C
!
radius server 10.23.61.150
address ipv4 10.23.61.150 auth-port 1645
key radius123
radius server host
address ipv4 172.24.194.60 auth-port 1645
key radius123
radius server 172.24.194.60
address ipv4 172.24.194.60 auth-port 1645
key radius123
line con 0
password 7 0822455D0A16
login authentication console
!
line aux 0
line vty 0 4
password 7 0822455D0A16
login authentication default
transport input ssh
line vty 5 15
password 7 0822455D0A16
login authentication default
transport input ssh
end

```

2. Конфігурація Cherednychenko_R3

Current configuration : 2920 bytes

```

!
version 15.4
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname Cherednychenko_R3
enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCi1
ip dhcp excluded-address 172.24.192.1 172.24.192.15
ip dhcp excluded-address 172.24.192.17 172.24.192.31
ip dhcp excluded-address 172.24.192.33 172.24.192.35

```

```
ip dhcp excluded-address 172.24.192.129 172.24.192.131
ip dhcp excluded-address 172.24.192.192 172.24.192.205
ip dhcp excluded-address 172.24.192.253 172.24.192.255
!
ip dhcp pool LAN5-VLAN10
network 172.24.192.128 255.255.255.192
default-router 172.24.192.129
dns-server 172.24.192.200
ip dhcp pool LAN5-VLAN30
network 172.24.192.192 255.255.255.192
default-router 172.24.192.193
dns-server 172.24.192.200
ip dhcp pool LAN5-VLAN20
network 172.24.192.64 255.255.255.192
default-router 172.24.192.65
dns-server 172.24.192.200
aaa new-model
!
aaa authentication login console group radius local
aaa authentication login default local
no ip cef
no ipv6 cef
username Cherednychenko01V password 7 082048430017544541
username Cherednychenko_R3 password 7 082048430017544541
ip domain-name Cherednychenko_3
spanning-tree mode pvst
interface GigabitEthernet0/0/0
no ip address
duplex auto
speed auto
interface GigabitEthernet0/0/0.10
encapsulation dot1Q 10
ip address 172.24.192.129 255.255.255.192
interface GigabitEthernet0/0/0.20
encapsulation dot1Q 20
ip address 172.24.192.65 255.255.255.192
!
interface GigabitEthernet0/0/0.30
encapsulation dot1Q 30
ip address 172.24.192.193 255.255.255.192
!
interface GigabitEthernet0/0/0.99
encapsulation dot1Q 99
no ip address
!
interface GigabitEthernet0/0/1
no ip address
duplex auto
speed auto
shutdown
!
interface Serial0/1/0
ip address 172.10.18.10 255.255.255.252
clock rate 128000
!
interface Serial0/1/1
ip address 172.10.18.13 255.255.255.252
clock rate 128000
!
interface Vlan1
no ip address
shutdown
!
router ospf 1
```

```

log-adjacency-changes
passive-interface default
no passive-interface GigabitEthernet0/0/0
no passive-interface Serial0/1/0
no passive-interface Serial0/1/1
auto-cost reference-bandwidth 1000
network 172.24.200.8 0.0.0.3 area 0
network 172.24.200.12 0.0.0.3 area 0
network 172.24.192.0 0.0.0.255 area 0
banner motd ^CCherednychenko_R3^C
!
radius server host
address ipv4 172.24.194.60 auth-port 1645
key radius123
radius server 172.24.194.60
address ipv4 172.24.194.60 auth-port 1645
key radius123
!
line con 0
password 7 0822455D0A16
login authentication console
!
line aux 0
!
line vty 0 4
password 7 0822455D0A16
login authentication default
transport input ssh
line vty 5 15
password 7 0822455D0A16
login authentication default
transport input ssh
end

```

3. Конфігурація Cherednychenko_R2

```

Current configuration : 2017 bytes
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname Cherednychenko_R2
enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCi1
ip dhcp excluded-address 172.24.194.1 172.24.194.4
ip dhcp excluded-address 172.24.194.62
ip dhcp excluded-address 172.24.194.63
!
ip dhcp pool LAN-1
network 172.24.194.0 255.255.255.192
default-router 172.24.194.1
dns-server 172.24.192.200
aaa new-model
!
aaa authentication login console group radius local
aaa authentication login default local
no ip cef
no ipv6 cef
username CherednychenkoOlV password 7 082048430017544541
username Cherednychenko_R2 password 7 082048430017544541
license udi pid CISCO2911/K9 sn FTX15248D8N-
ip domain-name Cherednychenko_R2
spanning-tree mode pvst
interface GigabitEthernet0/0
ip address 172.10.18.2 255.255.255.252

```

```

duplex auto
speed auto
!
interface GigabitEthernet0/1
no ip address
duplex auto
speed auto
!
interface GigabitEthernet0/2
ip address 172.24.194.1 255.255.255.192
duplex auto
speed auto
!
interface Vlan1
no ip address
shutdown
!
router ospf 1
log-adjacency-changes
passive-interface default
no passive-interface GigabitEthernet0/0
no passive-interface GigabitEthernet0/2
auto-cost reference-bandwidth 1000
network 172.24.200.0 0.0.0.3 area 0
network 172.24.194.0 0.0.0.63 area 0
!
ip classless
!
ip flow-export version 9
banner motd ^CCherednychenko_R2^C
!
radius server host
address ipv4 172.24.194.60 auth-port 1645
key radius123
radius server 172.24.194.60
address ipv4 172.24.194.60 auth-port 1645
key radius123
line con 0
password 7 0822455D0A16
login authentication console
!
line aux 0
!
line vty 0 4
password 7 0822455D0A16
login authentication default
transport input ssh
line vty 5 15
password 7 0822455D0A16
login authentication default
transport input ssh
end

```

4. Конфігурація Cherednychenko_SW1 etherchannel

```

Current configuration : 1722 bytes
!
version 15.0
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname Cherednychenko_SW1
!

```

```
enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCil
ip domain-name Cherednychenko_SW1
!
username Cherednychenko01V privilege 1 password 7 082048430017061E010803
spanning-tree mode pvst
spanning-tree extend system-id
!
interface Port-channel1
switchport mode trunk
!
interface Port-channel2
switchport mode trunk
!
interface FastEthernet0/1
switchport mode trunk
channel-group 1 mode active
!
interface FastEthernet0/2
switchport mode trunk
channel-group 1 mode active
!
interface FastEthernet0/3
switchport mode trunk
channel-group 2 mode active
!
interface FastEthernet0/4
switchport mode trunk
channel-group 2 mode active
!
interface FastEthernet0/5
!
interface FastEthernet0/6
!
interface FastEthernet0/7
!
interface FastEthernet0/8
!
interface FastEthernet0/9
!
interface FastEthernet0/10
!
interface FastEthernet0/11
!
interface FastEthernet0/12
!
interface FastEthernet0/13
!
interface FastEthernet0/14
!
interface FastEthernet0/15
!
interface FastEthernet0/16
!
interface FastEthernet0/17
!
interface FastEthernet0/18
!
interface FastEthernet0/19
!
interface FastEthernet0/20
!
interface FastEthernet0/21
!
interface FastEthernet0/22
```

```

!
interface FastEthernet0/23
!
interface FastEthernet0/24
!
interface GigabitEthernet0/1
!
interface GigabitEthernet0/2
!
interface Vlan1
no ip address
shutdown
!
banner motd ^CCherednychenko_SW1^C
line con 0
password 7 0822455D0A16
login
!
line vty 0 4
password 7 0822455D0A16
login local
transport input ssh
line vty 5 15
password 7 0822455D0A16
login local
transport input ssh
end

```

5. Конфігурація Cherednychenko_SW4 VLAN
Cherednychenko_SW4#show run
Building configuration...

Current configuration : 2933 bytes

```

!
version 15.0
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname Cherednychenko_SW4
!
enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCil
ip domain-name Cherednychenko_SW4
!
username Cherednychenko01V privilege 1 password 7 082048430017061E010803
spanning-tree mode pvst
spanning-tree extend system-id
!
interface FastEthernet0/1
switchport trunk native vlan 100
switchport trunk allowed vlan 10,20,30-100
switchport mode trunk
!
interface FastEthernet0/2
switchport trunk native vlan 100
switchport trunk allowed vlan 10,20,30-100
switchport mode trunk
!
interface FastEthernet0/3
switchport trunk native vlan 100
switchport trunk allowed vlan 10,20,30-100
switchport mode trunk
!
interface FastEthernet0/4

```

```
switchport trunk native vlan 100
switchport trunk allowed vlan 10,20,30-100
switchport mode trunk
!
interface FastEthernet0/5
switchport access vlan 20
switchport mode access
!
interface FastEthernet0/6
switchport access vlan 20
switchport mode access
!
interface FastEthernet0/7
switchport access vlan 20
switchport mode access
!
interface FastEthernet0/8
switchport access vlan 20
switchport mode access
!
interface FastEthernet0/9
switchport access vlan 20
switchport mode access
!
interface FastEthernet0/10
switchport access vlan 20
switchport mode access
!
interface FastEthernet0/11
!
interface FastEthernet0/12
switchport access vlan 10
switchport mode access
!
interface FastEthernet0/13
switchport access vlan 10
switchport mode access
!
interface FastEthernet0/14
switchport access vlan 10
switchport mode access
!
interface FastEthernet0/15
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/16
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/17
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/18
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/19
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/20
switchport access vlan 30
```



```
switchport mode access
!
interface FastEthernet0/21
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/22
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/23
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/24
switchport access vlan 30
switchport mode access
!
interface GigabitEthernet0/1
switchport access vlan 99
switchport mode trunk
!
interface GigabitEthernet0/2
!
interface Vlan1
no ip address
shutdown
!
interface Vlan99
ip address 172.24.192.2 255.255.255.240
!
ip default-gateway 172.24.192.1
!
banner motd ^CCherednychenko_SW4^C
line con 0
password 7 0822455D0A16
login
!
line vty 0 4
password 7 0822455D0A16
login local
transport input ssh
line vty 5 15
password 7 0822455D0A16
login local
transport input ssh
end
```