

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Комп'ютеризована система керування поштовою скринькою

Виконав: студент IV курсу, групи СІ-42

спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

(підпис)

Цейко В.Л.

(прізвище та ініціали)

Керівник

(підпис)

Жаровський Р.О.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Тим Є. В.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Осухівська Г.М.

(прізвище та ініціали)

Рецензент

(підпис)

Бойко І.В.

(прізвище та ініціали)

Тернопіль
2025

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Осухівська Г.М.
(підпис) (прізвище та ініціали)
«27» січня 2025 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня бакалавр
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

студенту Цейко Володимир Леонідовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютеризована система керування поштовою скринькою

Керівник роботи Жаровський Руслан Олегович кандидат технічних наук, доцент кафедри КС
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 27 » січня 2025 року № 4/7-53

2. Термін подання студентом завершеної роботи _____

3. Вихідні дані до роботи Технічне завдання

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Аналіз технічного завдання

2. Проектна частина

3. Практична частина

4. Безпека життєдіяльності, основи охорони праці.

Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Структурна схема системи

2. Схема електрична принципова

3. Блок-схема алгоритму роботи програмного забезпечення

4. Схема тестового макета

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Безпека життєдіяльності, основи охорони праці</i>	<i>д.т.н., професор Пилипець М. І.</i>		

7. Дата видачі завдання 27.01.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	<i>Розробка технічного завдання</i>	<i>27.01 – 02.02</i>	<i>Викон.</i>
2.	<i>Аналіз технічного завдання</i>	<i>15.02 – 27.03</i>	<i>Викон.</i>
3.	<i>Проектна частина</i>	<i>06.04 – 02.05</i>	<i>Викон.</i>
4.	<i>Практична частина</i>	<i>04.05 – 25.05</i>	<i>Викон.</i>
5.	<i>Безпека життєдіяльності, основи охорони праці</i>	<i>26.05 – 02.06</i>	<i>Викон.</i>
6.	<i>Оформлення пояснювальної записки і графічного матеріалу</i>	<i>03.06 – 10.06</i>	<i>Викон.</i>
7.	<i>Перевірка на академічний плагіат, перевірка керівником та консультантами</i>	<i>9.06 – 15.06</i>	<i>Викон.</i>
8.	<i>Попередній захист кваліфікаційної роботи бакалавра</i>	<i>16.06 – 22.06</i>	<i>Викон.</i>
9.	<i>Захист кваліфікаційної роботи бакалавра</i>	<i>25.06</i>	<i>Викон.</i>

Студент

(підпис)*Цейко В. Л.*

(прізвище та ініціали)

Керівник роботи

(підпис)*Жаровський Р. О.*

(прізвище та ініціали)

АНОТАЦІЯ

Цейко В.Л. Комп'ютеризована система керування поштовою скринькою: робота на здобуття кваліфікаційного ступеня бакалавра: спец. 123 – комп'ютерна інженерія. – Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2025.

Ключові слова: комп'ютеризована система, поштова скринька, ESP8266, інтернет речей (IoT), сенсор, Telegram-бот, сповіщення, бездротова передача даних.

Кваліфікаційна робота присвячена розробці розумної поштової скриньки – пристрою, здатного автоматично реагувати на події (отримання посилки, відкриття кришки) та надсилати відповідні сповіщення користувачу в Telegram у режимі реального часу. Система побудована на базі мікроконтролера ESP8266 з підключенням сенсорів контролю та використанням інтернет-з'єднання через Wi-Fi.

Пояснювальна записка складається з чотирьох розділів.

У першому розділі виконано аналіз потреб користувачів у системах дистанційного моніторингу поштових скриньок, обґрунтовано доцільність використання бездротових технологій.

У другому розділі розроблено архітектуру системи, обґрунтовано вибір елементної бази, зокрема ESP8266, сенсорів та засобів зв'язку.

У третьому розділі описано програмну реалізацію, зокрема логіку обробки подій, взаємодію з Telegram-ботом та тестування пристрою в реальних умовах.

У четвертому розділі розглянуто питання безпеки життєдіяльності.

ANNOTATION

Tseyko V.L. Computerized system for managing a mailbox: Bachelor's Graduation Thesis: speciality 123 – computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2025.

Keywords: computerized system, mailbox, ESP8266, Internet of Things (IoT), sensor, Telegram bot, notifications, wireless data transmission.

This bachelor's qualification paper is dedicated to the development of a smart mailbox – a device capable of automatically reacting to events (parcel delivery, lid opening) and sending corresponding real-time notifications to the user via Telegram. The system is based on the ESP8266 microcontroller, with connected control sensors and Internet connectivity via Wi-Fi.

The explanatory note consists of four chapters.

The first chapter analyzes the needs of users for remote mailbox monitoring systems and substantiates the feasibility of using wireless technologies.

The second chapter presents the system architecture and justifies the choice of hardware components, including the ESP8266, sensors, and communication tools.

The third chapter describes the software implementation, particularly the event-handling logic, interaction with the Telegram bot, and testing of the device in real conditions.

The fourth chapter addresses issues related to life safety.

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ	10
1.1 Основні технічні вимоги до комп'ютеризованої системи керування поштовою скринькою	10
1.2 Аналіз можливих рішень поставленого завдання	11
1.3 Огляд існуючих варіантів реалізації системи.....	13
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА.....	16
2.1 Вибір апаратного та програмного забезпечення проєктованої системи ..	16
2.1.1 Вибір мікроконтролера та його характеристики	16
2.1.2 Вибір клавіатури	21
2.1.3 Вибір сервоприводу	23
2.1.4 Вибір дисплею	24
2.1.5 Вибір датчику руху та принцип його роботи	25
2.2 Опис електричної принципової схеми модуля керування	27
2.3 Структура комп'ютеризованої системи керування поштовою скринькою	28
2.4 Вибір ПЗ для реалізації системи	30
2.4.1 Середовище розробки ПЗ для мікроконтролера.....	30
2.4.2 Telegram сервіс для дистанційного керування системою	32
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА.....	34

					КС КРБ 123.243.00.00 ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата								
Розроб.		Цейко В.Л.			Комп'ютеризована система керування поштовою скринькою				Літ.	Арк.	Аркушів	
Перевір.		Жаровський Р.О.									6	
Реценз.		Бойко І.В.							ТНТУ, каф. КС, гр. СІ-42			
Н. Контр.		Тиш Є.В.										
Затверд.		Осухівська Г.М.										

3.1	Розробка алгоритму роботи комп'ютеризованої системи керування поштовою скринькою	34
3.2	Створення макету системи	38
3.3	Опис програмних функцій та модулів	39
3.3.1	Ініціалізація бібліотек, пінів та констант.....	39
3.3.2	Опис програмних функцій	40
3.4	Налаштування середовища розробки програми для модуля NodeMCU ..	44
3.5	Тестування прототипу комп'ютеризованої системи керування поштовою скринькою	46
РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ ..		51
4.1	Ергономічні проблеми безпеки життєдіяльності.	51
4.2	Вплив електромагнітних полів (ЕМП) на людину та заходи щодо зменшення їх впливу на обслуговуючий персонал.	53
ВИСНОВКИ		55
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ		56
Додаток А Технічне завдання		
Додаток Б Перелік елементів		
Додаток В Лістинг коду		

СПИСОК СКОРОЧЕНЬ

ADC – Analog-to-Digital Converter

API – Application Programming Interface

GPIO – General-Purpose Input/Output

I2C – Inter-Integrated Circuit

IoT – Internet of Things

IARC – International Agency for Research on Cancer

ICNIRP – International Commission on Non-Ionizing Radiation Protection

LCD – Liquid Crystal Display

UART – Universal Asynchronous Receiver-Transmitter

					КС КРБ 123.243.00.00 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Комп'ютеризована система керування поштовою скринькою є сучасним технічним засобом, що забезпечує автоматизований контроль та взаємодію з поштовою скринькою за допомогою комп'ютерних або мобільних пристроїв. Така система реалізується на основі концепції Інтернету речей (IoT) і може функціонувати як окремий пристрій або бути частиною більшої інфраструктури розумного будинку. Її впровадження дозволяє значно підвищити зручність користувача, а також забезпечити інформованість і безпеку під час отримання кореспонденції.

Система дозволяє у режимі реального часу отримувати інформацію про стан скриньки, надходження нових листів або посилок, а також контролювати доступ до її вмісту. Це особливо корисно для користувачів, які часто перебувають поза домом і бажають своєчасно отримувати повідомлення про доставлену кореспонденцію. Крім того, за допомогою технологій сповіщення, таких як push-повідомлення, електронна пошта чи SMS, система дозволяє оперативно реагувати на зміну стану пристрою.

Високий рівень інтеграції з іншими компонентами автоматизованої інфраструктури, зокрема освітленням, камерами відеоспостереження чи сигналізацією, робить таку систему не лише зручною, але й надійною з точки зору безпеки. У разі відкриття скриньки можуть запускатися відповідні сценарії, що сприяють захисту особистої інформації та майна.

Таким чином, комп'ютеризована система керування поштовою скринькою є ефективним рішенням для забезпечення зручного, безпечного та технологічно просунутого способу взаємодії з традиційними засобами отримання кореспонденції.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Основні технічні вимоги до комп'ютеризованої системи керування поштовою скринькою

Для забезпечення ефективної роботи комп'ютеризованої системи керування поштовою скринькою необхідно сформулювати чіткі вимоги до її функціональності та загальних характеристик. Такі вимоги дозволяють створити надійний, зручний у користуванні та захищений пристрій, здатний виконувати автоматизований контроль доступу до поштової скриньки, здійснювати сповіщення та працювати в автономному режимі [1].

Функціональні вимоги:

- Виявлення подій. Система повинна фіксувати відкриття скриньки, надходження нової кореспонденції, а також взаємодію користувача з клавіатурою або іншими елементами введення. Для цього використовуються сенсори (наприклад, інфрачервоні або контактні), правильно налаштовані та розміщені в конструкції.
- Керування доступом. Фізичне блокування або відкриття поштової скриньки реалізується за допомогою сервоприводу, яким керує мікроконтролер після перевірки введеного користувачем PIN-коду. У разі правильного коду замок відкривається, у разі неправильного — залишається заблокованим.
- Авторизація користувача. Ідентифікація проводиться через цифрову клавіатуру. Система обробляє введений пароль і приймає рішення про доступ. У разі неправильного вводу активується захист: сповіщення про спробу несанкціонованого доступу.

					<i>КС КРБ 123.243.00.00 ПЗ</i>		
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>			
<i>Розроб.</i>		<i>Цейко В.Л.</i>			<i>Аналіз технічного завдання</i>		
<i>Перевір.</i>		<i>Жаровський</i>					
<i>Реценз.</i>		<i>Бойко І.В.</i>					
<i>Н. Контр.</i>		<i>Тиш Є.В.</i>					
<i>Затверд.</i>		<i>Осухівська Г.М.</i>					
					<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
						<i>10</i>	
					<i>ТНТУ, каф. КС, гр. СІ-42</i>		

– Віддалене сповіщення. Події, пов’язані з надходженням кореспонденції, відкриттям скриньки або спробами введення неправильного PIN-коду, надсилаються користувачеві через Telegram-бота у вигляді миттєвих повідомлень.

– Інформаційний інтерфейс. На OLED-дисплеї відображаються повідомлення про статус скриньки, результат авторизації, підказки щодо дій або помилки.

1.2 Аналіз можливих рішень поставленого завдання

Для реалізації комп’ютеризованої системи керування поштовою скринькою необхідно продумати архітектуру, яка дозволить виявляти події, забезпечувати керування доступом, надавати інформацію користувачу та підтримувати дистанційний моніторинг. З огляду на поставлену задачу, можливими є різні варіанти реалізації, які залежать від вимог до функціональності, енергоспоживання, надійності та зручності користування [2].

У першу чергу система повинна мати можливість визначення факту отримання кореспонденції або відкриття скриньки. Для цього можуть бути використані різноманітні сенсори, наприклад, інфрачервоні датчики, датчики руху, геркони, ємнісні або оптичні сенсори. Кожен з них має свої переваги: одні краще підходять для виявлення руху поруч із пристроєм, інші – для фіксації зміни положення дверцят або наявності об’єкта у скриньці.

Для обробки сигналів від сенсорів та реалізації логіки роботи системи доцільно використати мікроконтролер із підтримкою бездротового зв’язку. Це може бути модуль з вбудованим Wi-Fi або Bluetooth, або ж контролер із можливістю підключення до мережі через зовнішні засоби зв’язку. Прикладом можуть бути як енергоефективні 8- або 32-бітні мікроконтролери, так і

					КС КРБ 123.243.00.00 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

одноплатні комп'ютери, у випадку потреби у більш складній обробці даних [3].

З метою зручного керування доступом до скриньки, система може бути оснащена елементами введення інформації. Це можуть бути цифрові клавіатури, сенсорні панелі, RFID-зчитувачі або інші пристрої ідентифікації. Для підтвердження введення або сповіщення про стан скриньки доцільно використати світлодіоди, зумери або дисплеї – зокрема, текстові або графічні модулі, які дозволяють виводити повідомлення чи інструкції [4].

Для підвищення безпеки система може мати механізм блокування скриньки, наприклад, за допомогою електромеханічного замка чи сервомеханізму. Контроль за відкриванням може здійснюватися лише після успішної авторизації користувача. Також важливо врахувати реалізацію захисту від несанкціонованого доступу, включно з лімітами на кількість спроб авторизації та можливістю фіксації підозрілої активності.

Інформування користувача про події можливе через локальні сигнали, але значно ефективнішим є впровадження віддалених повідомлень. Зокрема, система може підтримувати надсилання push-повідомлень, повідомлень у месенджери, електронну пошту або інтеграцію з мобільними застосунками. Для цього потрібно забезпечити підтримку сучасних протоколів передачі даних, таких як MQTT, HTTP, HTTPS або інші.

Оскільки скринька може бути встановлена поза приміщенням або в місці без постійного доступу до мережі, необхідно передбачити автономне живлення. Для цього можуть використовуватися акумулятори, батареї або сонячні панелі.

Також варто враховувати можливість розширення функціоналу системи у майбутньому, що дозволить підключати додаткові сенсори, модулі зв'язку або інтегрувати систему в більшу платформу «розумного будинку». Інтеграція з іншими пристроями – камерами спостереження, системами сигналізації,

					КС КРБ 123.243.00.00 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

інтелектуальними голосовими асистентами – дає змогу створити централізовану та взаємодіючу екосистему.

Таким чином, існує кілька можливих рішень для побудови системи керування поштовою скринькою. Конкретний вибір компонентів залежатиме від технічного завдання, бюджету, умов експлуатації та рівня автоматизації, який бажає досягти користувач [5].

1.3 Огляд існуючих варіантів реалізації системи

У сучасному світі, де автоматизація побутових процесів набуває дедалі більшого поширення, активно розвиваються й комп'ютеризовані системи керування поштовими скриньками. Такі системи призначені для підвищення зручності, безпеки та надійності доставки кореспонденції, а також для інтеграції в екосистему розумного будинку.

Серед промислових рішень можна виділити системи Danby Parcel Guard, Yale Smart Delivery Box, iParcelBox, а також BenchSentry Package Box, які отримали широке поширення на ринку Північної Америки та Європи. Усі ці продукти мають електронне керування замком, сенсорну систему виявлення посилок та підключення до мобільного додатку. Наприклад, Yale Smart Delivery Box (рис. 1.1) підтримує Wi-Fi-зв'язок і дозволяє користувачеві керувати доступом до скриньки через застосунок, віддалено блокуючи або відкриваючи її.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 1.1 – Зовнішній вигляд Yale Smart Delivery Box

Система iParcelBox пропонує двосторонню комунікацію: власник може дозволити доставку в реальному часі, переглядати статус пристрою й отримувати push-сповіщення.

Усі зазначені рішення орієнтовані на інтеграцію з цифровими екосистемами, такими як Amazon Alexa, Google Assistant, Apple HomeKit, що дозволяє керувати пристроями за допомогою голосових команд, автоматизованих сценаріїв або централізованих мобільних платформ. Крім того, пристрої мають функції відеоспостереження, GPS-трекінгу, захисту від несанкціонованого доступу та шифрування переданих даних.

Варто зазначити, що більшість комерційних систем забезпечують високу надійність і стабільність роботи, проте мають суттєвий недолік – високу вартість. Це обумовлено використанням якісних матеріалів, фірмового програмного забезпечення та необхідністю сертифікації для ринку. У багатьох випадках ці пристрої вимагають щомісячної підписки для використання частини хмарних функцій.

Технологічні тенденції також вказують на перехід до автономних, енергоефективних систем із можливістю живлення від акумулятора чи сонячної панелі. Наприклад, Danby Parcel Guard (рис.1.2) пропонує живлення як від електромережі, так і від резервного джерела енергії. Такі рішення

особливо актуальні у приватному секторі, де скринька розташована на значній відстані від будинку.



Рисунок 1.2 – Зовнішній вигляд Danby Parcel Guard

У майбутньому очікується ще більша інтеграція поштових скриньок із розумними будинками, включно з підключенням до систем відеоспостереження, датчиків руху, сигналізацій та біометричних засобів автентифікації. Подібні функціональні модулі вже розглядаються виробниками як стандарт для нових моделей.

Загалом, комп'ютеризовані системи керування поштовими скриньками демонструють динамічний розвиток і мають значний потенціал у сфері автоматизації доставки. Комерційні рішення, що вже представлені на ринку, можуть стати основою для розробки нових, більш доступних, але не менш функціональних систем.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

2.1 Вибір апаратного та програмного забезпечення проєктованої системи

У цьому розділі розглядається апаратне забезпечення комп'ютеризованої системи керування поштовою скринькою, яка призначена для контролю доступу, фіксації факту надходження кореспонденції та інформування користувача про стан скриньки. Наведено опис основних апаратних модулів, що входять до складу системи, їх функціональні призначення та особливості взаємодії з мікроконтролером. Особливу увагу приділено критеріям вибору апаратних компонентів відповідно до вимог щодо надійності, енергозбереження, безпеки та можливості подальшої інтеграції в середовище розумного будинку [6].

2.1.1 Вибір мікроконтролера та його характеристики

ESP8266 виступає центральним обчислювальним модулем у комп'ютеризованій системі керування поштовою скринькою. Його основна роль полягає в обробці сигналів від датчиків, керуванні периферійними пристроями, реалізації логіки доступу, а також забезпеченні бездротового зв'язку з користувачем. Вибір саме цього мікроконтролера зумовлений його функціональною універсальністю, наявністю вбудованого Wi-Fi модуля та високою обчислювальною продуктивністю.

Мікроконтролер оснащено 32-розрядним двоядерним процесором Tensilica Xtensa LX6, що працює на частоті до 240 МГц. Така обчислювальна потужність дає змогу паралельно опрацьовувати сигнали від датчиків руху, вводити код з клавіатури, керувати сервоприводом і дисплеєм, а також

					КС КРБ 123.243.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Цейко В.Л.			Проектна частина		
Перевір.		Жаровський Р.О.					
Реценз.		Бойко І.В.					
Н. Контр.		Тиш С.В.					
Затверд.		Осухівська Г.М.					
					Літ.	Арк.	Аркушів
						16	
					ТНТУ, каф. КС, гр. СІ-42		

підтримувати активний зв'язок із віддаленим сервером або мобільним додатком [7].



Рисунок 2.1 – Мікроконтролер ESP8266

Завдяки підтримці енергозберігаючих режимів, включно з глибоким сном (deep sleep), ESP8266 (рис.2.1) здатен функціонувати в автономному режимі протягом тривалого часу, що особливо актуально при живленні від акумулятора або сонячної панелі. Це дозволяє зменшити частоту обслуговування пристрою й підвищити його надійність у віддалених умовах.

Постійний аналіз стану інфрачервоного сенсора руху дозволяє ESP8266 оперативно реагувати на фізичну активність біля скриньки. Після виявлення події мікроконтролер чекає на введення динамічного коду з клавіатурного модуля, проводить перевірку введених даних та, у разі правильного коду, активує сервомеханізм для відкриття замка. Всі дії супроводжуються відображенням інструкцій або повідомлень на OLED-дисплеї, який під'єднаний через інтерфейс I2C. Додатково для візуального зворотного зв'язку використовується світлодіод, що сигналізує про успішність операції.

Ще однією ключовою перевагою ESP8266 є його здатність підтримувати підключення до мережі Інтернет, що дозволяє реалізувати функції віддаленого

					КС КРБ 123.243.00.00 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

моніторингу, оповіщень через Telegram або push-повідомлень. Це відкриває широкі можливості для інтеграції пристрою в екосистеми «розумного будинку».

На рисунку 2.2 зображено структурну схему внутрішньої конструкції мікроконтролера ESP8266.

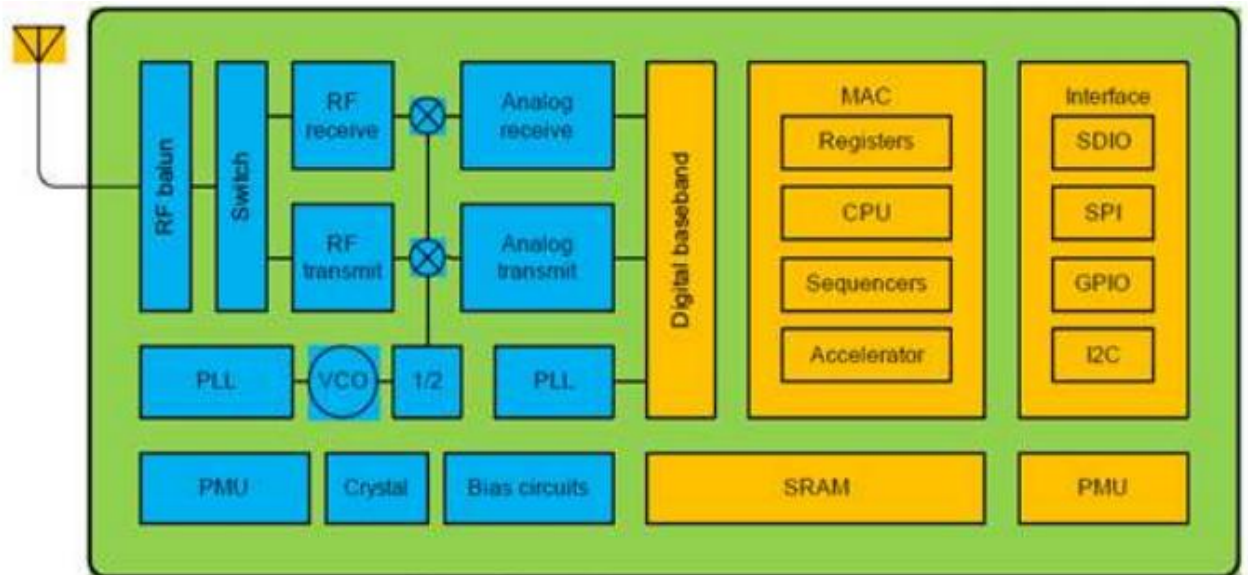


Рисунок 2.2 – Структурна схема внутрішньої будови мікроконтролера ESP8266

Розвинена периферія мікроконтролера (I2C, SPI, UART, PWM, ADC) забезпечує просте та ефективне підключення всіх необхідних компонентів. Завдяки цьому реалізується стабільна, координована робота системи, яка зберігає гнучкість для подальшого розширення функціоналу або модернізації обладнання [8].

Таким чином, ESP8266 ідеально підходить як обчислювальна платформа для проєктованої системи, забезпечуючи баланс між енергоефективністю, функціональністю та продуктивністю.

Мікроконтролер ESP8266, який виступає центральним елементом системи керування поштовою скринькою, забезпечує підтримку широкого спектра інтерфейсів, зокрема UART, I2C, SPI, GPIO, PWM, а також має

вбудовані аналогово-цифрові перетворювачі (ADC). Це дає змогу легко інтегрувати різноманітні периферійні пристрої та сенсори, що розширює функціональність системи.

Ключові технічні характеристики мікроконтролера ESP8266 зображено в таблиці 2.1.

Таблиця 2.1 – Характеристики мікроконтролера ESP8266

Параметр	Значення
Напруга живлення	3.3 В
Протоколи	802.11 b/g/n
Захист	WPA/WPA2/WEP/WPS
Діапазон робочих температур	-40°C ~ 125°C
Апаратні засоби та інтерфейси	UART, SPI, I ² C, PWM, ADC
Тактова частота	80 МГц – 160 МГц

Наявність ADC дозволяє обробляти аналогові сигнали, у разі використання відповідних датчиків. За допомогою I2C-шини підключено OLED-дисплей, який виводить актуальну інформацію про роботу пристрою, включаючи інструкції з введення пароля та повідомлення про стан доступу. Для керування сервоприводом і зчитування даних з датчика руху та цифрової клавіатури застосовуються входи/виходи GPIO.

Ключовою перевагою ESP8266 є інтегрований Wi-Fi, що дає змогу реалізувати бездротове з'єднання з користувачем. Завдяки цьому система може відправляти повідомлення про події у режимі реального часу через такі протоколи, як MQTT, HTTP або з використанням Telegram Bot API. На рисунку 2.3 представлено схему призначення виводів мікроконтролера ESP8266.

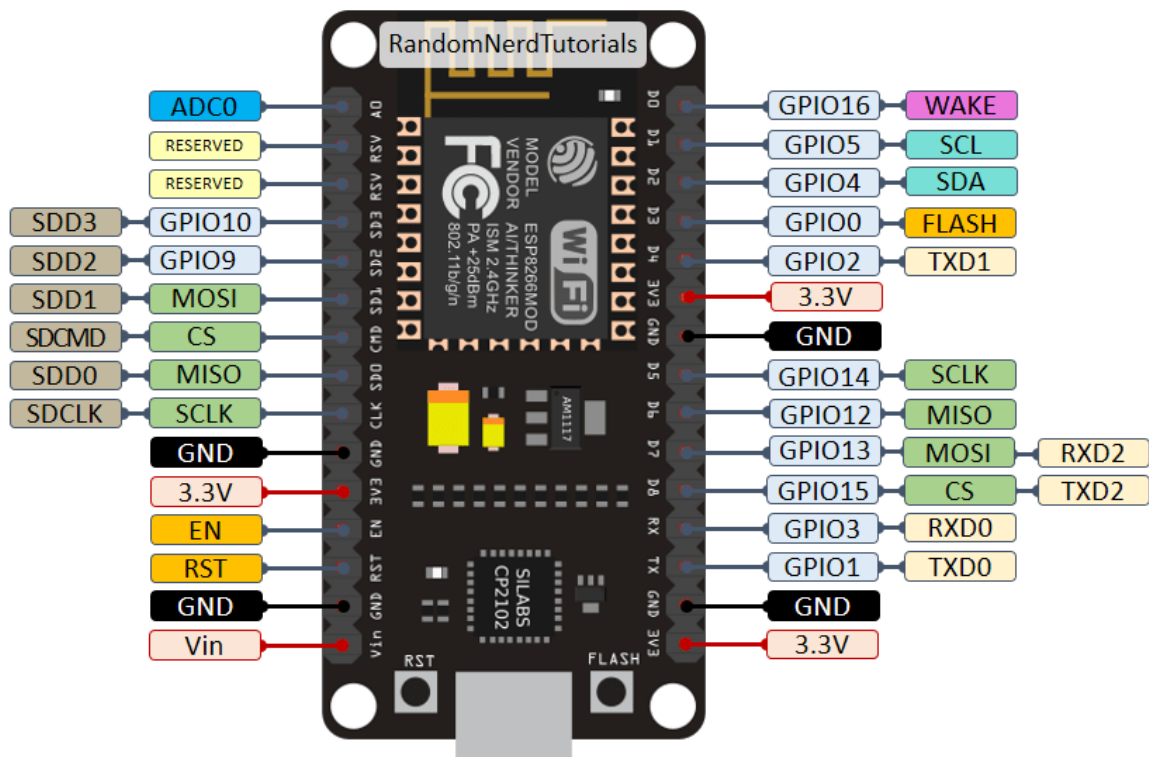


Рисунок 2.3 – Піни ESP8266

Мікроконтролер ESP8266 виконує ключову роль в системі керування поштовою скринькою, забезпечуючи зчитування даних із датчиків, керування виконавчими елементами та організацію бездротової передачі інформації до користувача через Wi-Fi. На відміну від автономних рішень з локальною обробкою, система на основі ESP8266 орієнтована на мережеву інтеграцію, що дозволяє користувачеві в режимі реального часу отримувати повідомлення про події, пов'язані зі скринькою.

Незважаючи на обмежену обчислювальну потужність у порівнянні з більш продуктивними мікроконтролерами, ESP8266 ефективно виконує поставлені завдання завдяки підтримці цифрових інтерфейсів (GPIO, I²C, UART) і вбудованому Wi-Fi-модулю. Такий підхід дозволяє поєднати просту реалізацію з функціональністю IoT-рішення.

Принципова електрична схема плати на базі мікроконтролера ESP8266 представлена на рисунку 2.4.

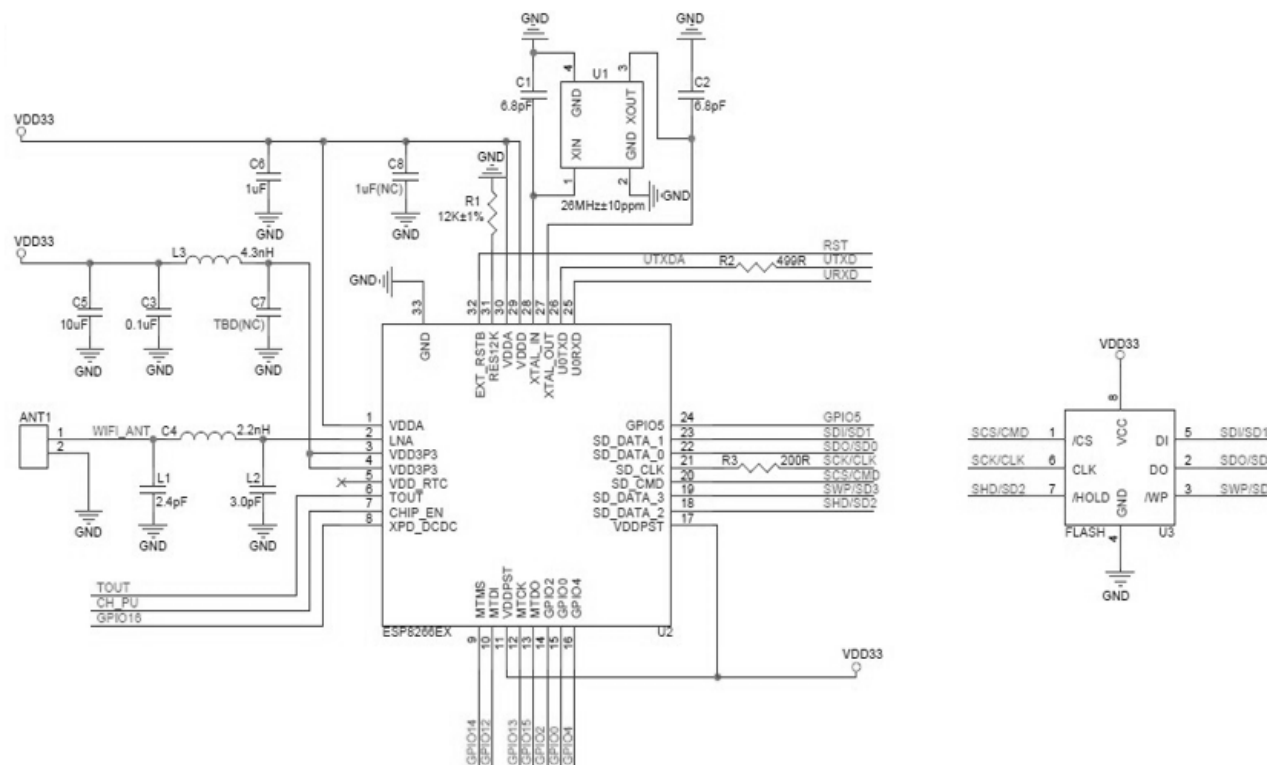


Рисунок 2.4 – Схема електрична принципова плати мікроконтролера ESP8266

Ця принципова електрична схема мікроконтролера ESP8266EX дозволяє зрозуміти його структуру живлення, схему узгодження з Wi-Fi антеною, використання кварцового резонатора на 26 МГц для тактування, а також підключення зовнішньої SPI флеш-пам'яті для зберігання прошивки. Вона показує, які GPIO використовуються для стандартної роботи та завантаження мікроконтролера, як організовано UART-зв'язок для програмування і обміну даними, та яким чином здійснюється обмін по I2C або SPI. Також зображені необхідні компоненти (конденсатори, котушки, резистори), що забезпечують фільтрацію, стабілізацію живлення та правильну роботу всіх модулів ESP8266.

2.1.2 Вибір клавіатури

Вибір 4×4 клавіатурного модуля (рис. 2.5) для комп'ютеризованої системи керування поштовою скринькою обґрунтовується низкою практичних і технічних переваг, які роблять цей компонент оптимальним для реалізації функції введення цифрового коду доступу.



Рисунок 2.5 – Зовнішній вигляд клавіатури 4×4

З технічного погляду, дана клавіатура працює на принципі матричної структури (4 рядки × 4 стовпці), що значно зменшує кількість необхідних входів/виходів мікроконтролера: для опитування всіх клавіш, що зручно для проєктів з обмеженим числом доступних портів. До того ж, реалізація опитування клавіш можлива через стандартні програмні бібліотеки (наприклад, Keypad.h в Arduino IDE), що спрощує програмування.

Ще однією важливою перевагою є простота інтеграції. Клавіатурний модуль не потребує складного електронного підключення – достатньо прямого з'єднання з GPIO через шлейф або конектор. Це дозволяє легко монтувати клавіатуру на корпус пристрою, забезпечуючи зручний доступ користувача до системи.

Також клавіатура має високу надійність і тривалий термін служби, не потребує додаткового живлення, не чутлива до змін освітлення, температури чи пилу (в порівнянні, наприклад, з сенсорними екранами), що важливо для використання пристрою в різних середовищах.

Таким чином, вибір 4×4 клавіатури є логічним з точки зору

					КС КРБ 123.243.00.00 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

функціональності, енергоефективності, легкості реалізації та вартості. Вона забезпечує надійне введення інформації, що є критично важливим для системи з обмеженим фізичним доступом до вмісту поштової скриньки.

2.1.3 Вибір сервоприводу

Для виконання проєкту було використано сервопривід SG90 (рис.2.6), який виконує роль виконавчого елемента для механічного блокування або відкриття поштової скриньки. Цей мініатюрний сервомеханізм є ідеальним вибором завдяки своїй компактності, невеликій вазі, низькому енергоспоживанню та достатній силі для переміщення замка чи механічного важеля. Він забезпечує кут повороту до 180 градусів, чого цілком достатньо для реалізації функції відкривання та закривання.



Рисунок 2.6 – Зовнішній вигляд сервоприводу

Керування сервоприводом здійснюється мікроконтролером ESP8266 за допомогою сигналів широтно-імпульсної модуляції (PWM), що дозволяє точно задавати необхідний кут повороту. Сигнал надсилається лише у момент зміни положення, що сприяє економії енергії, особливо в автономному режимі роботи пристрою.

Після введення правильного коду з клавіатури або підтвердження доступу з мобільного застосунку, мікроконтролер активує сервомеханізм,

					КС КРБ 123.243.00.00 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

який повертає вал і відкриває кришку або знімає блокування. По завершенні дії привод повертається у вихідне положення, блокуючи доступ до вмісту скриньки до наступної авторизованої команди.

Завдяки простоті інтеграції, надійності й швидкому відгуку SG90 є оптимальним вибором для системи, що вимагає недорогого, але ефективного рішення для механічного керування.

2.1.4 Вибір дисплею

Для реалізації текстового інтерфейсу у даній системі управління поштовою скринькою було обрано дисплей LCD 1602 із інтерфейсом I2C. Цей модуль виявився оптимальним варіантом завдяки поєднанню компактності, низького енергоспоживання та зручності інтеграції з мікроконтролером.

LCD 1602 (рис.2.7) дозволяє виводити до 32 символів одночасно на двох рядках, що є цілком достатнім для відображення інструкцій, повідомлень про стан пристрою, результатів введення коду чи сповіщень про доставку. Завдяки текстовому формату, інформація залишається чіткою та зрозумілою навіть при мінімальних апаратних ресурсах.

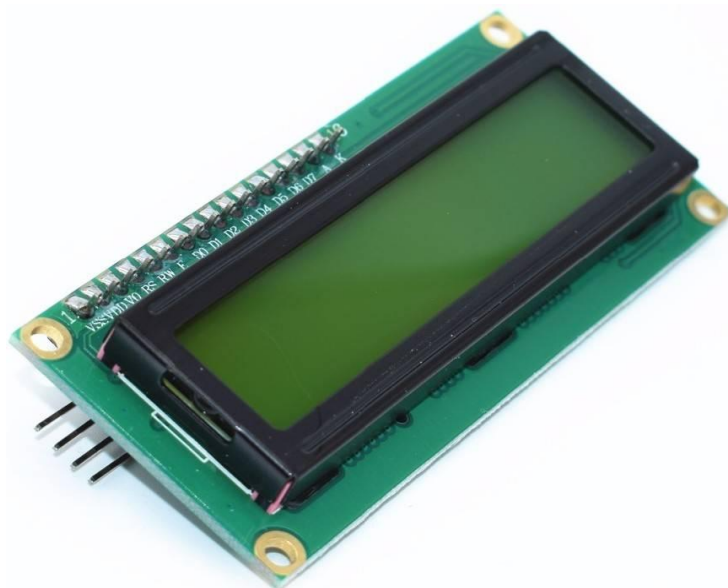


Рисунок 2.7 – Зовнішній вигляд дисплею

					КС КРБ 123.243.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

Інтерфейс I2C, яким обладнаний дисплей, значно зменшує кількість необхідних з'єднань: для обміну даними з ESP8266 використовується лише два сигнальні виводи – SDA та SCL. Це особливо важливо у випадках, коли кількість доступних GPIO на мікроконтролері обмежена. Крім того, підключення по I2C спрощує схему, зменшує кількість потенційних помилок при монтажі та сприяє більш охайному розміщенню компонентів.

Ще одним вагомим аргументом на користь вибору цього дисплея є його економність у споживанні енергії. У пристрої, який працює від батареї або павербанку, це критично важливо. LCD 1602 не створює суттєвого навантаження на джерело живлення, що дає змогу подовжити час автономної роботи системи.

Додатково варто зазначити, що цей модуль широко підтримується в Arduino-середовищі: існує багато стабільних бібліотек, які полегшують програмну реалізацію функцій відображення. Це суттєво пришвидшує розробку та зменшує кількість технічних труднощів.

Отож, вибір LCD 1602 з інтерфейсом I2C є обґрунтованим з позицій функціональності, простоти реалізації, енергоефективності та сумісності з обраною платформою, що робить його надійним і доцільним рішенням у складі розроблюваної системи.

2.1.5 Вибір датчику руху та принцип його роботи

Для реалізації функції виявлення факту внесення кореспонденції до поштової скриньки в системі застосовується інфрачервоний датчик руху типу PIR (Passive Infrared Sensor), розміщений всередині корпусу та орієнтований безпосередньо на вхідний отвір. Така конфігурація дає змогу фіксувати рух теплокровного об'єкта – зокрема руки людини або частини тіла – у момент вкладання листа чи посилки, що дозволяє точно визначити момент фактичного надходження пошти.

Вибір PIR-сенсора HC-SR501 зумовлений його здатністю розпізнавати зміну інфрачервоного випромінювання в полі зору, що виникає при

					КС КРБ 123.243.00.00 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

потраплянні тіла, яке випромінює тепло, в зону дії. Це забезпечує чутливу реакцію на наближення руки до отвору скриньки, навіть без використання оптичних або механічних елементів, які могли б ускладнити конструкцію [9].

Принцип дії PIR-сенсора базується на детекції змін теплового фону – при появі об'єкта з температурою, вищою за температуру навколишнього середовища, відбувається збурення, яке викликає генерацію імпульсу. В межах цієї системи такий імпульс інтерпретується як подія надходження пошти, що може слугувати тригером для фіксації події, відправки повідомлення користувачу або зміни статусу скриньки в інтерфейсі моніторингу.

Перевагою такого розташування сенсора є ізоляція від зовнішніх джерел руху – він не реагує на випадкове проходження повз скриньку, а активується виключно в момент взаємодії з отвором.

Цифровий вихід PIR-сенсора безпосередньо з'єднується з GPIO-портом мікроконтролера ESP8266, що спрощує інтеграцію та не потребує складної схеми обробки сигналів. Крім того, він характеризується низьким енергоспоживанням і стабільною роботою в умовах закритого простору, що робить його оптимальним для внутрішнього монтажу. Завдяки точному зональному реагуванню, надійності та простоті використання, PIR-сенсор ефективно виконує роль детектора надходження кореспонденції.

Ключові технічні характеристики HC-SR501 зображено в таблиці 2.2 [8].

Таблиця 2.2 – Характеристики HC-SR501

Параметр	Значення
Робоча напруга	4.5 - 20 В
Струм споживання	35.5 мкА в режимі очікування та 1.5-1.6 мА в активному режимі
Час затримки	регульований (0,3 секунди 18 секунд)
Вихідний сигнал	високий 3.3 В, низький 0 В
Кут огляду	120 градусів, в межах 7 метрів

2.2 Опис електричної принципової схеми модуля керування

Схема (рис.2.8) описує роботу керованого пристрою на базі мікроконтролера ESP8266, який координує роботу кількох компонентів: клавіатури, LCD-дисплея, сервоприводу SG90, інфрачервоного датчика руху HC-SR501 та світлодіода.

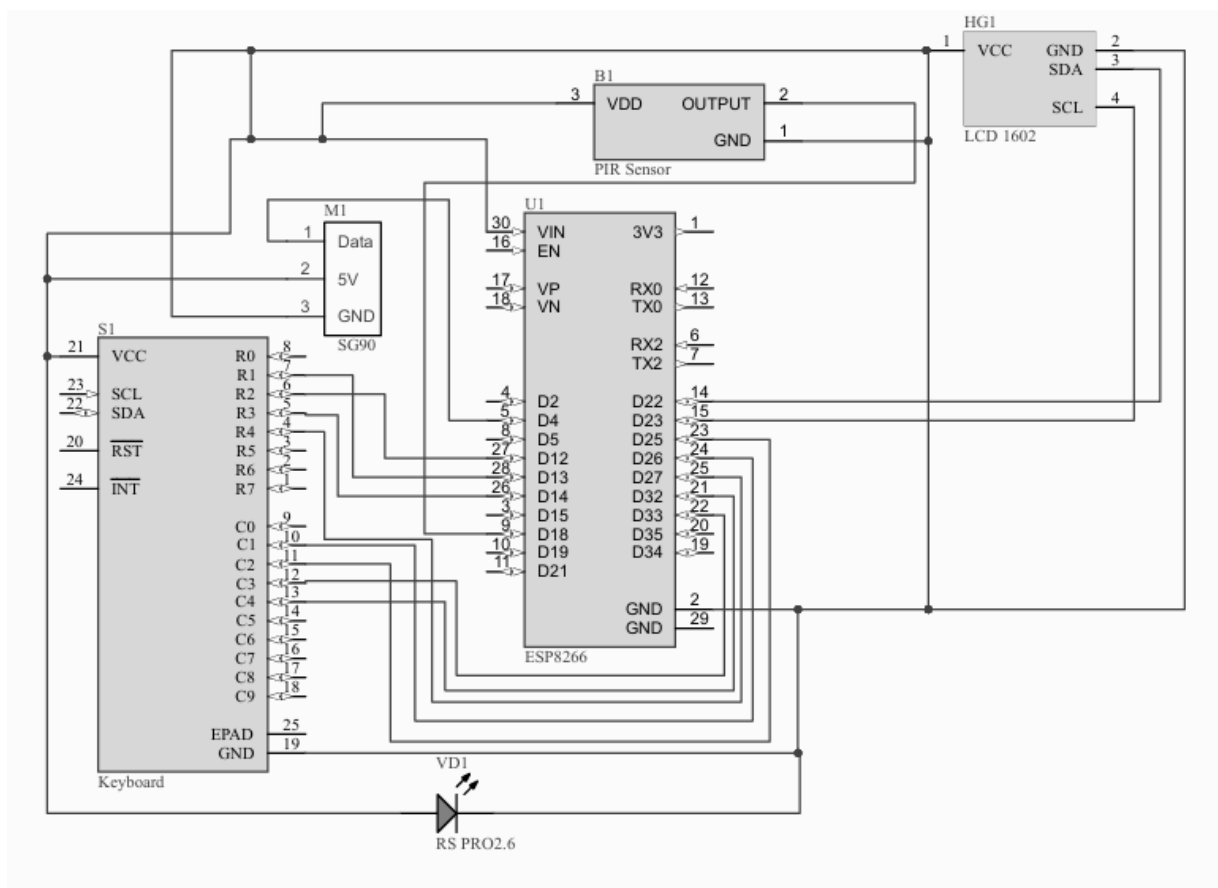


Рисунок 2.8 – Схема електрична принципова комп'ютеризованої системи

ESP8266 отримує живлення через контакт VIN, куди подається 5 вольт. Він з'єднаний із загальною землею через два виводи GND. Контролер підключений до різних периферійних пристроїв через цифрові GPIO-виводи. Наприклад, сервопривід SG90 підключений до ESP8266 через пін D5, куди подається керуючий сигнал. Живлення для сервоприводу йде безпосередньо від 5V, а загальне заземлення підключене до GND.

PIR сенсор (HC-SR501), що виявляє рух, має три виходи: VDD, GND і Output. Він живиться також від 5 вольт, а сигнал з виходу Output подається на пін D6 мікроконтролера. Коли PIR виявляє рух, на цей пін надходить логічна «1», що дозволяє ESP8266 почати подальшу логіку обробки події.

LCD-дисплей 1602 підключений через інтерфейс I2C. Його пін SDA з'єднаний з піном D3 мікроконтролера, а пін SCL – з D4. Таке підключення дозволяє передавати текстову інформацію, наприклад повідомлення для користувача, про очікування введення коду, про успішний доступ або помилку.

Клавіатура у схемі – матрична, з підключенням через I2C-інтерфейс. Живлення подається 3.3 вольт на відповідний пін VCC клавіатури. Зв'язок здійснюється через пін SDA (D1 на ESP8266) і SCL (D0). Окремо подано сигнал скидання клавіатури через пін RST, що з'єднаний із піном D2 ESP8266. Пін INT клавіатури підключено до D7 контролера, що дозволяє генерувати переривання при натисканні кнопок.

Також у схемі є світлодіод VD1 типу RS PRO 2.6. Його анод підключений до піну D8 ESP8266, а катод – до землі. Через цей світлодіод може подаватися візуальний сигнал користувачу – наприклад, при успішному введенні коду або при надходженні кореспонденції.

2.3 Структура комп'ютеризованої системи керування поштовою скринькою

Структурна схема комп'ютеризованої системи керування поштовою скринькою наведена на рисунку 2.9. Ця структурна схема показує взаємозв'язок між різними компонентами системи [10].

					КС КРБ 123.243.00.00 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

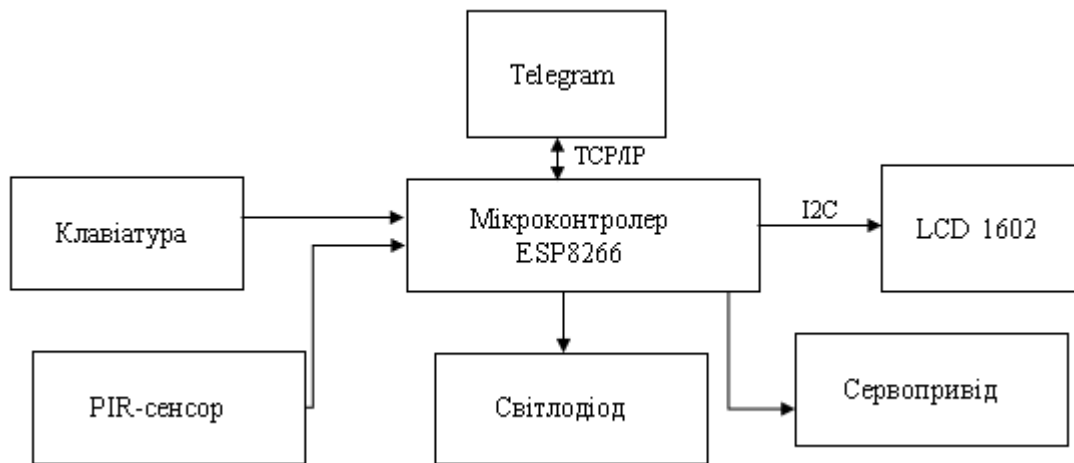


Рисунок 2.9 – Структурна схема

ESP8266 є головним елементом системи, який виконує обробку сигналів, керування пристроями, обмін даними та зв'язок із мережею. Завдяки вбудованому Wi-Fi-модулю мікроконтролер забезпечує можливість надсилання повідомлень, взаємодії з мобільним застосунком або Telegram-ботом. ESP8266 приймає сигнали з сенсорів, обробляє введення користувача, керує сервомеханізмом і дисплеєм.

До ESP8266 підключено клавіатурний модуль 4x4, який дозволяє вводити цифровий код доступу. Цей код використовується для авторизації користувача з метою відкриття поштової скриньки. Введені дані передаються на обробку до ESP8266, де здійснюється порівняння з внутрішнім еталонним значенням (паролем).

До системи входить датчик руху (у схемі – PIR-сенсор), який виявляє присутність або рух поблизу поштової скриньки. При виявленні події мікроконтролер реєструє факт доступу, що може бути використано для надсилання повідомлення власнику про відкриття скриньки або спробу взаємодії.

Система включає сервопривід, який фізично відкриває або блокує замок поштової скриньки після успішного введення коду. Сигнал керування формується мікроконтролером після проходження авторизації. У випадку невірного коду – привід залишається заблокованим.

Для візуального сповіщення про стан системи використовується світлодіод (LED). Він може інформувати про помилку введення, успішну авторизацію або зміну стану замка. Індикація допомагає користувачеві оперативно оцінити стан пристрою без необхідності додаткових засобів.

До ESP8266 підключено OLED-дисплей, який відображає інформацію про поточний стан системи: запит на введення коду, підтвердження доступу, повідомлення про помилку, інструкції для користувача тощо. Він підключений через інтерфейс I2C, що спрощує підключення та зменшує кількість необхідних виводів.

Завдяки вбудованому Wi-Fi в ESP8266, система підтримує зв'язок з мережею Інтернет. Це дозволяє реалізувати віддалене сповіщення користувача через Telegram, email або мобільний застосунок. Також можливе використання хмарних сервісів для моніторингу та збереження даних.

2.4 Вибір ПЗ для реалізації системи

2.4.1 Середовище розробки ПЗ для мікроконтролера

Для розробки програмного забезпечення комп'ютеризованої системи керування поштовою скринькою було обрано середовище Arduino IDE. Основною причиною такого вибору є його сумісність із мікроконтролером ESP8266, який використовується як основа системи. Arduino IDE забезпечує зручне середовище розробки, що дозволяє писати, компілювати та завантажувати скетчі (програми) безпосередньо на контролер через USB-підключення.

Arduino IDE підтримує велику кількість бібліотек, зокрема для роботи з сервоприводами, LCD-дисплеями (через I2C), клавіатурами та інфрачервоними датчиками руху, що значно спрощує реалізацію логіки пристрою. Це дозволило швидко інтегрувати всі необхідні компоненти без потреби написання низькорівневого коду для кожного модуля окремо [11].

					КС КРБ 123.243.00.00 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

Крім того, Arduino IDE має простий і інтуїтивно зрозумілий інтерфейс, що знижує поріг входу для розробника. Завдяки активній спільноті, документації та численним прикладам, реалізація складніших алгоритмів (наприклад, обробки введення коду, керування доступом чи відображення повідомлень) стала значно ефективнішою.

Ще однією перевагою Arduino IDE є можливість інтеграції з онлайн-сервісами (через Wi-Fi модуль ESP8266) та підтримка серійної відладки, що дало змогу ефективно перевіряти роботу системи на етапі налагодження. Головне вікно застосунку Arduino IDE представлено на рисунку 2.10.



Рисунок 2.10 – Головне вікно застосунку Arduino IDE

Таким чином, вибір Arduino IDE як основного середовища розробки дозволив поєднати простоту, функціональність і гнучкість, що є ключовими факторами у створенні комп'ютеризованої системи керування поштовою скринькою.

2.4.2 Telegram сервіс для дистанційного керування системою

Для реалізації функції дистанційного моніторингу та керування в комп'ютеризованій системі було обрано сервіс Telegram, який забезпечує гнучку та захищену взаємодію між користувачем і пристроєм через спеціально створеного бота. Telegram Bot API дозволяє реалізувати як односпрямоване надсилання сповіщень, так і приймання команд від користувача для віддаленого керування [12].

Принцип роботи полягає у використанні бота, який реагує на визначені команди або події, що надходять із боку пристрою. Наприклад, система може надсилати повідомлення про певний стан (відкриття, тривога, рух тощо), а користувач у свою чергу може керувати пристроєм – надіслати команду для відкриття доступу або створення одноразового PIN-коду.

Взаємодія відбувається за принципом:

- тригер – певна подія в системі (наприклад, спрацювання сенсора або введення PIN-коду);
- дія – відповідь Telegram-бота у вигляді повідомлення користувачеві або виконання дії, ініційованої командою.

Система періодично перевіряє наявність нових повідомлень від користувача через API Telegram, обробляє введені команди та надсилає відповідні відповіді. Такий підхід не потребує використання додаткових платформ типу IFTTT чи зовнішніх серверів, що робить його простим у впровадженні й ефективним у роботі.

Завдяки використанню Telegram:

- забезпечується зручність користування – сповіщення надходять безпосередньо на смартфон;
- гарантується захист даних – передача інформації здійснюється через захищені з'єднання (TLS/SSL);
- можлива гнучка обробка команд – система може розрізняти дозволених користувачів, реагувати лише на авторизовані запити тощо.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

Вікно чату Telegram для створення бота представлено на рисунку 2.11.

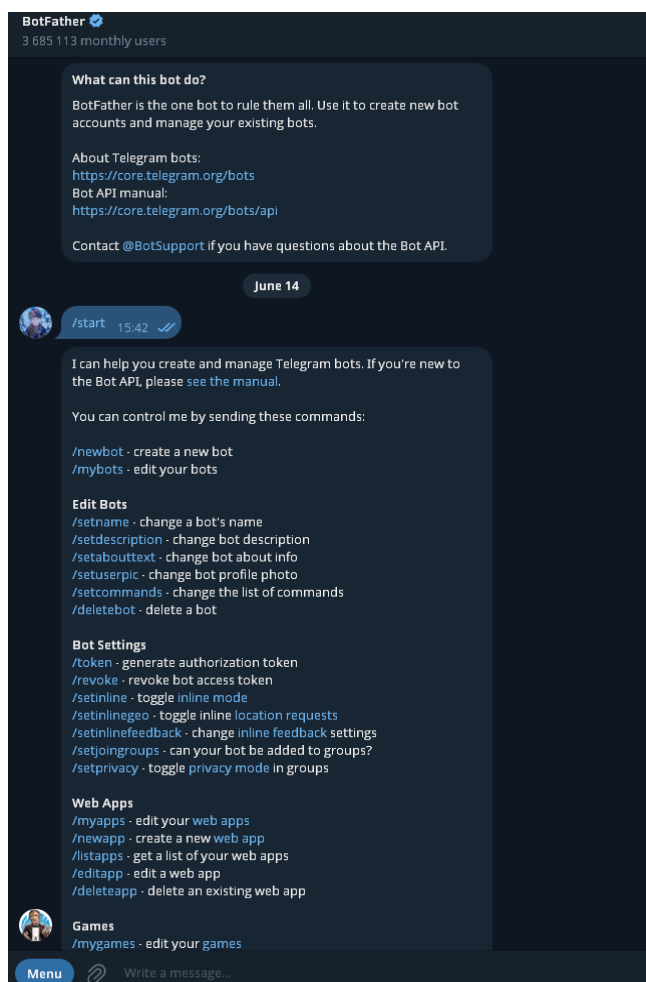


Рисунок 2.11 – Вікно чату для створення бота

Таким чином, Telegram виступає як надійний канал комунікації між користувачем і системою, дозволяючи отримувати сповіщення та керувати пристроєм у режимі реального часу з будь-якого місця, де є доступ до Інтернету.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Розробка алгоритму роботи комп'ютеризованої системи керування поштовою скринькою

Алгоритм роботи програми для мікроконтролера ESP8266 модуля NodeMCU, яка реалізує розумну систему контролю поштової скриньки, передбачає послідовність взаємодій, починаючи з налаштування апаратних засобів і завершуючи обробкою запитів Telegram-користувачів:

Здійснюється підключення необхідних бібліотек, що забезпечують функціональність мережевого з'єднання, обробку клавіатури, управління LCD-дисплеєм, серводвигуном та обмін повідомленнями через Telegram API. Водночас оголошуються глобальні змінні для зберігання станів пристрою, PIN-кодів, дозволених користувачів, параметрів таймерів.

Мікроконтролер встановлює Wi-Fi-з'єднання з локальною мережею за вказаними обліковими даними. У разі успішного підключення виводиться IP-адреса в монітор порту, після чого ініціалізуються всі підключені пристрої – LCD-дисплей, клавіатура, сервомотор і датчик руху. Сервомотор переводиться у замкнений стан, на дисплеї з'являється повідомлення про необхідність введення PIN-коду.

У головному циклі програма постійно сканує натискання кнопок на клавіатурі. При введенні користувачем коду відображаються зірочки, що маскують символи. Кнопка “*” очищає введення, а “#” запускає перевірку коду. Якщо введений код збігається з активним тимчасовим PIN-кодом, система відкриває замок, відображає повідомлення про успішний доступ і надсилає відповідне сповіщення в Telegram. У разі неправильного коду

					КС КРБ 123.234.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата	Практична частина		
Розроб.		Цейко В.Л.					
Перевір.		Жаровський Р.О.					
Реценз.		Бойко І.В.					
Н. Контр.		Тиш Є.В.					
Затверд.		Осухівська Г.М.					
					Літ.	Арк.	Аркушів
						34	
					ТНТУ, каф. КС, гр. СІ-42		

виводиться повідомлення про помилку, і також надсилається повідомлення про невдалу спробу доступу.

Паралельно контролюється стан PIR-сенсора. Якщо фіксується рух поблизу скриньки (що інтерпретується як надходження нової пошти), програма надсилає повідомлення всім авторизованим Telegram-користувачам із інформацією про це і порадою згенерувати тимчасовий PIN.

Програма регулярно (раз на 5 секунд) перевіряє наявність нових повідомлень у Telegram. Якщо користувач надсилає команду /start, він додається до списку дозволених. Команда /generate генерує одноразовий PIN-код, який дійсний протягом 5 хвилин і надсилається в чат. Команда /open дозволяє дистанційно відкрити замок і отримати повідомлення про відкриття. Команда /getlog надсилає вміст лог-файлу. Команда /clearlog очищує лог-файл і фіксує це у новому записі.

Усі події записуються у файл журналу log.txt у внутрішній пам'яті (SPIFFS) з мітками часу. Якщо файл перевищує встановлений ліміт у 2048 байт, він автоматично видаляється та створюється новий файл з відповідним записом, і користувачі Telegram отримують повідомлення про це.

Крім цього, контролюється час дії тимчасового PIN-коду. Якщо термін його дії закінчився, PIN анулюється автоматично. У разі, якщо замок був відкритий, через 5 секунд він автоматично зачиняється, і система повертається до режиму очікування нового PIN-коду.

Блок-схема алгоритму роботи основної програми зображена на рисунку 3.1.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

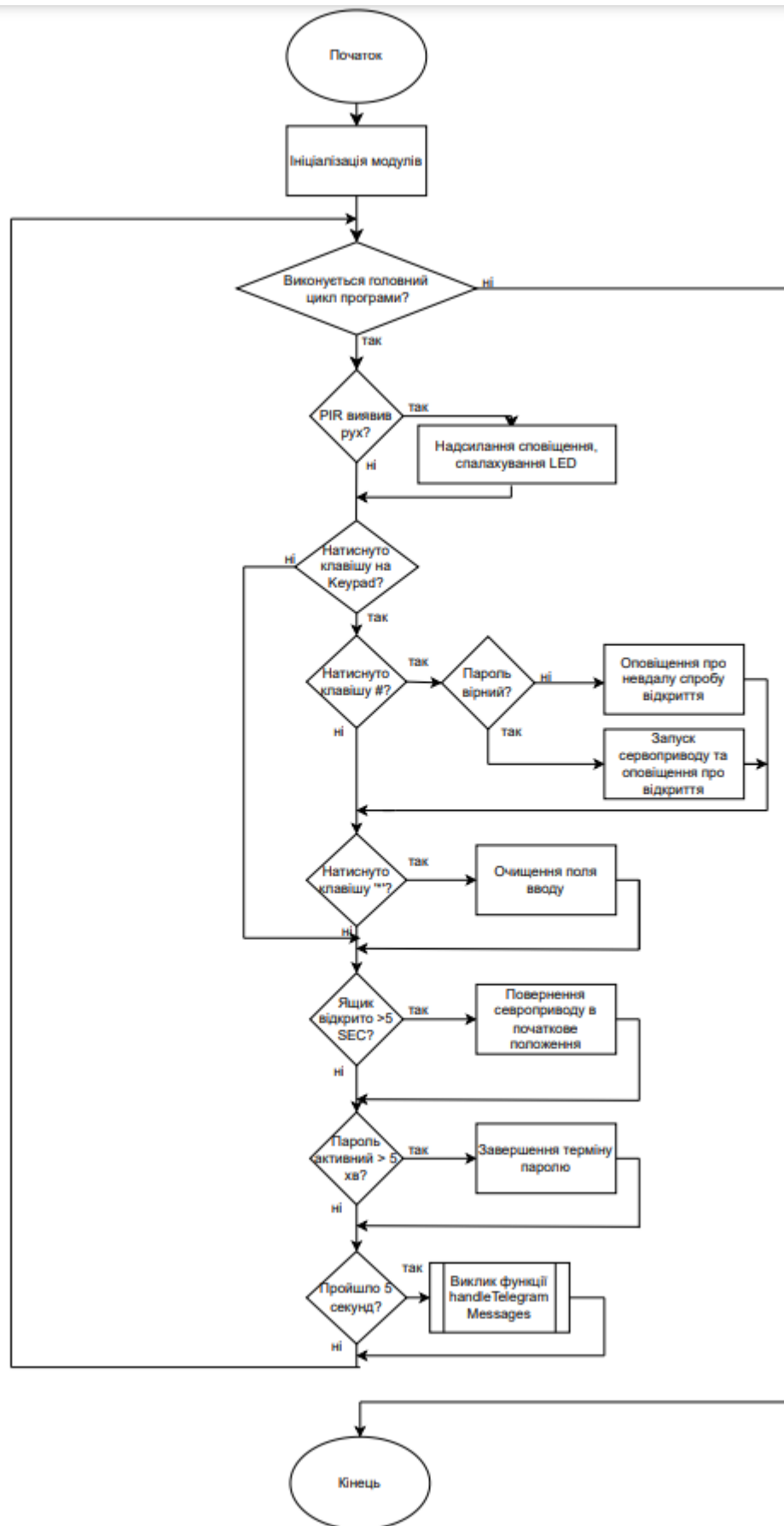


Рисунок 3.1 – Блок-схема алгоритму роботи основної програми

Змн.	Арк.	№ докум.	Підпис	Дата

КС КРБ 123.243.00.00 ПЗ

Арк.

36

На рисунку 3.2 зображена блок-схема алгоритму роботи функції handleTelegramMessages.

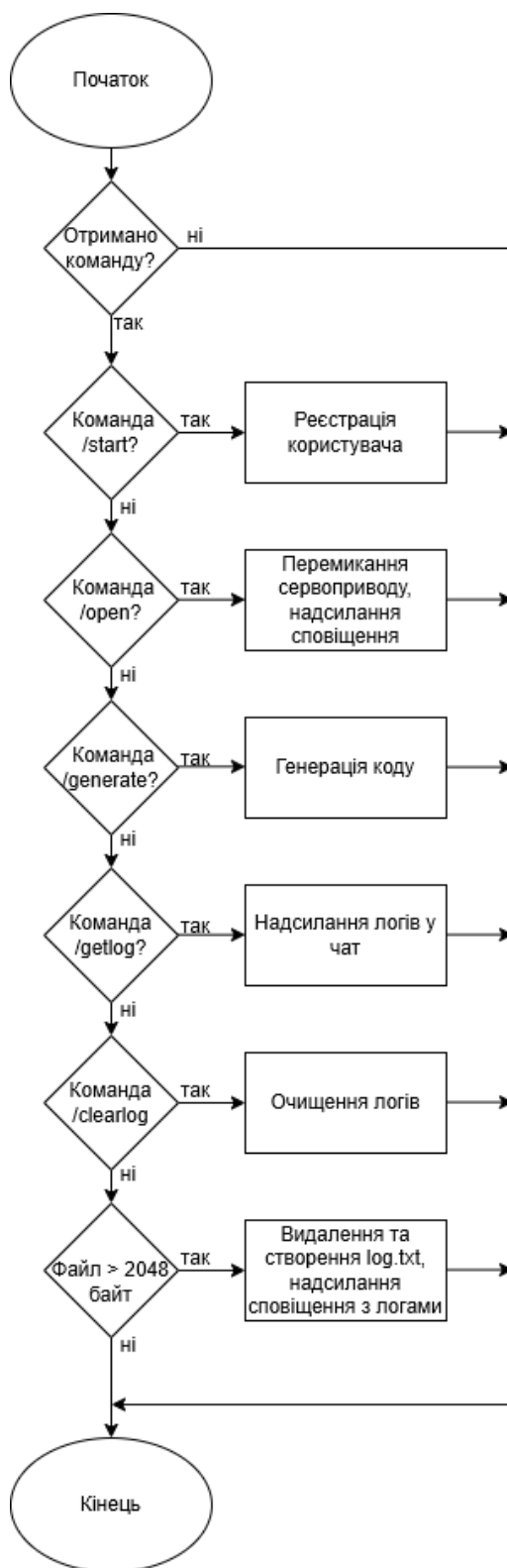


Рисунок 3.2 – Блок-схема алгоритму роботи функції handleTelegramMessages

3.2 Створення макету системи

Для створення макету комп'ютеризованої системи керування поштовою скринькою було використано симулятор Wokwi, який забезпечує зручне візуальне середовище для розробки та тестування проєктів на основі мікроконтролерів.

Wokwi дозволяє моделювати роботу Arduino, ESP8266, ESP32 та інших мікроконтролерів без потреби в реальному обладнанні, що дає змогу швидко перевірити працездатність схеми, логіку коду, а також взаємодію між різними компонентами: датчиками, дисплеями, клавіатурами, сервоприводами тощо. Це значно спрощує процес розробки, особливо на ранніх етапах, коли ще не всі фізичні елементи доступні. На рисунку 3.3 зображено схему макета системи. [13].

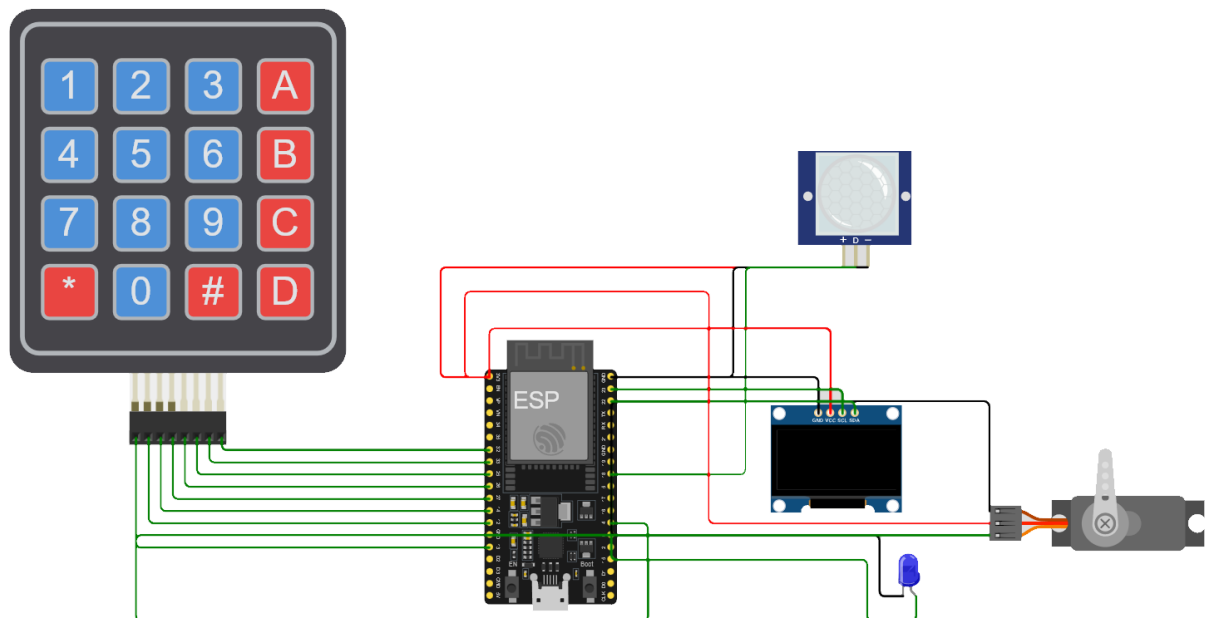


Рисунок 3.3 – Схема тестового макета

У результаті проведеного моделювання жодних помилок у роботі пристрою виявлено не було. Усі компоненти взаємодіяли відповідно до заданої логіки, сигнали коректно передавалися між мікроконтролером та підключеними модулями, а алгоритм доступу виконувався згідно з очікуваним

					КС КРБ 123.243.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		38

сценарієм. Це підтверджує правильність складеної схеми та обраної структури програми, а також дозволило перейти до етапу фізичного складання пристрою з упевненістю в його функціональності.

3.3 Опис програмних функцій та модулів

3.3.1 Ініціалізація бібліотек, пінів та констант.

У коді проєкту задіяно бібліотеки, що подані на рисунку 3.4.

```
1  #include <ESP8266WiFi.h>
2  #include <Wire.h>
3  #include <LiquidCrystal_I2C.h>
4  #include <Keypad.h>
5  #include <Servo.h>
6  #include <WiFiClientSecure.h>
7  #include <ArduinoJson.h>
8  #include <UniversalTelegramBot.h>
9  #include <FS.h>
10 #include <time.h>
11
```

Рисунок 3.4 – Лістинг коду, який відповідає за ініціалізацію бібліотек

Опис пінів та констант:

- PIRpin (D8) – пін, до якого підключено датчик руху, що реагує на вкладання листів;
- SERVOpin (D4) – пін для керування сервоприводом, який відповідає за механізм блокування/розблокування дверцят поштової скриньки;
- LCD_SDAPin (D2) – пін для передавання даних до LCD-дисплея по інтерфейсу I²C;
- LCD_SCLpin (D1) – пін для синхронізації сигналу дисплея через шину I²C;

- KEYPADrowPins (D6, D7) – піни, що підключені до рядків матричної клавіатури;
- KEYPADcolPins (D3, D0, D5) – піни, що підключені до стовпців клавіатури, що дозволяє зчитувати введення PIN-коду;
- WiFi_SSID та WiFi_PASSWORD – параметри для підключення пристрою до локальної Wi-Fi мережі;
- TelegramBotToken – токен, який ідентифікує Telegram-бота та дозволяє надсилати йому повідомлення через Telegram API;
- TempPinLifetime ($5 * 60 * 1000$) – константа, що визначає термін дії тимчасового PIN-коду (5 хвилин у мілісекундах);
- MaxUsers (5) – максимальна кількість користувачів, яким дозволено отримувати сповіщення від системи;
- UnlockDuration (5000 мс) – час, на який відкривається замок після введення коректного PIN-коду.

3.3.2 Опис програмних функцій

Функція setup() виконується один раз під час запуску пристрою. Вона відповідає за початкове налаштування системи: ініціалізує серійний порт для виводу інформації, підключає ESP8266 до Wi-Fi мережі, встановлює безпечне HTTPS-з'єднання без перевірки сертифікатів. Також відбувається ініціалізація LCD-дисплея, клавіатури, підключення сервоприводу з початковим положенням та налаштування піну датчика руху як вхідного. Наприкінці виводиться початкове повідомлення на дисплей. На рисунку 3.5 зображено лістнинг функції setup().

					КС КРБ 123.243.00.00 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

```

void setup() {
  Serial.begin(115200);
  Serial.println("Booting...");
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println();
  Serial.print("Connected, IP: ");
  Serial.println(WiFi.localIP());
  secured_client.setInsecure();
  Wire.begin(D2, D1);
  lcd.begin(16, 2);
  lcd.backlight();
  servo.attach(SERVO_PIN);
  servo.write(0);
  pinMode(PIR_PIN, INPUT);
  lcd.setCursor(0, 0);
  lcd.print("Enter PIN: ");
}

```

Рисунок 3.5 – Лістинг функції setup()

Функція loop() постійно виконується й відповідає за обробку натискань клавіш на клавіатурі: зчитує введений символ, виводить його в монітор порту, а при натисканні # перевіряє, чи активний тимчасовий PIN і чи збігається він із введеним; у разі успіху відкриває скриньку (керує серво), показує повідомлення на екрані та надсилає сповіщення в Telegram, а при помилці – виводить повідомлення "Wrong PIN" і також інформує через Telegram (рис. 3.6)

```

void loop() {
  char key = keypad.getKey();
  if (key) {
    Serial.print("KEYPAD: "); Serial.println(key);
    if (key == '#') {
      if ((tempPinActive && pinInput == tempPin)) {
        lcd.clear();
        lcd.print("Access OK");
        servo.write(180);
        isUnlocked = true;
        unlockTime = millis();
        tempPinActive = false;
        sendTelegramMessage("📧 Mailbox opened via keypad  
using temporary PIN.");
      } else {
        lcd.clear();
        lcd.print("Wrong PIN");
        sendTelegramMessage("❌ Incorrect PIN entered via  
keypad.");
      }
    }
  }
}

```

Рисунок 3.6 – Частина лістингу функції loop()

					КС КРБ 123.243.00.00 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

Продовження функції `loop()` обробляє введення PIN-коду: якщо після відкриття скриньки PIN був правильним, введення очищується, виводиться повідомлення й знову запрошення до введення; якщо натиснуто *, введення також очищується і показується напис "Cleared"; якщо натиснута цифра, вона додається до PIN і на екрані виводяться зірочки замість символів; окрім цього, якщо скринька була відкрита і минуло більше 5 секунд, вона автоматично зачиняється, стан скидається, і знову виводиться запрошення до введення PIN (рис.3.7)

```

        pinInput = "";
        delay(1500);
        lcd.clear();
        lcd.print("Enter PIN:");
    } else if (key == '*') {
        pinInput = "";
        lcd.setCursor(0, 1);
        lcd.print("Cleared          ");
        delay(1000);
        lcd.clear();
        lcd.print("Enter PIN:");
    } else {
        pinInput += key;
        lcd.setCursor(0, 1);
        lcd.print("          ");
        lcd.setCursor(0, 1);
        for (int i = 0; i < pinInput.length(); i++) {
            lcd.print("*");
        }
    }
}

if (isUnlocked && millis() - unlockTime > 5000) {
    servo.write(0);
    isUnlocked = false;
    lcd.clear();
    lcd.print("Enter PIN:");
}

```

Рисунок 3.7 – Продовження лістингу функції `loop()`

Функція `handleTelegramMessages()` відповідає за обробку вхідних повідомлень від Telegram-бота. Вона перевіряє наявність нових команд і виконує відповідні дії. Наприклад, команда `/start` додає нового користувача до списку дозволених, `/generate` створює тимчасовий PIN-код, який надсилається користувачеві, а команда `/open` відкриває замок дистанційно. Крім того, є

					КС КРБ 123.243.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

команди для надсилання (/getlog) та очищення (/clearlog) лог-файлів. Всі дії фіксуються в журналі подій. Ця функція забезпечує інтерактивний зв'язок між системою та користувачами через Telegram.

Функція `sendTelegramMessage(const String& msg)` надсилає текстове повідомлення всім користувачам, які мають дозвіл на отримання сповіщень. Вона проходить по списку дозволених користувачів і через Telegram-бота надсилає їм вказаний текст. Результати надсилання логуються у серійному моніторі.

Функція `isUserAllowed(String chatID)` перевіряє, чи входить конкретний ID користувача у список дозволених. Вона порівнює вхідний chatID із усіма записаними в масиві і повертає true, якщо користувач дозволений, або false в іншому випадку.

Функція `addUser(String chatID)` додає нового користувача до списку дозволених, якщо для цього є вільне місце і користувач ще не доданий. Після додавання виводиться повідомлення у серійний монітор із підтвердженням нового користувача.

Функція `logEvent()` відповідає за створення записів у лог-файлі /log.txt у файловій системі SPIFFS. Кожен запис містить мітку часу (timestamp) і опис події. Якщо розмір лог-файлу перевищує встановлений ліміт (2048 байтів), файл автоматично надсилається в телеграм та видаляється, створюється новий, і записується інформація про очищення. Завдяки цій функції забезпечується історія подій системи для подальшого аналізу або діагностики.

Функція `getTimestamp()` створює текстовий рядок з поточною датою і часом у форматі YYYY-MM-DD HH:MM:SS. Вона використовується в `logEvent()` для створення точних міток часу для кожного запису журналу. Завдяки використанню NTP-серверів, система отримує точний час з Інтернету.

Ця функція використовується для зчитування вмісту лог-файлу /log.txt та надсилання його в Telegram у вигляді текстового повідомлення. Перед надсиланням до вмісту додається заголовок (content). Функція корисна для

					КС КРБ 123.243.00.00 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

отримання історії подій прямо в месенджері, без потреби фізичного доступу до пристрою.

3.4 Налаштування середовища розробки програми для модуля NodeMCU

Щоб мати змогу створювати програмне забезпечення для мікроконтролера плати NodeMCU у середовищі Arduino IDE [16], спершу потрібно додати підтримку цієї плати. Для цього необхідно відкрити вікно налаштувань (меню File → Preferences) та у полі для додаткових URL-адрес менеджера плат вставити посилання на JSON-файл із конфігурацією підтримки NodeMCU (рис. 3.8):

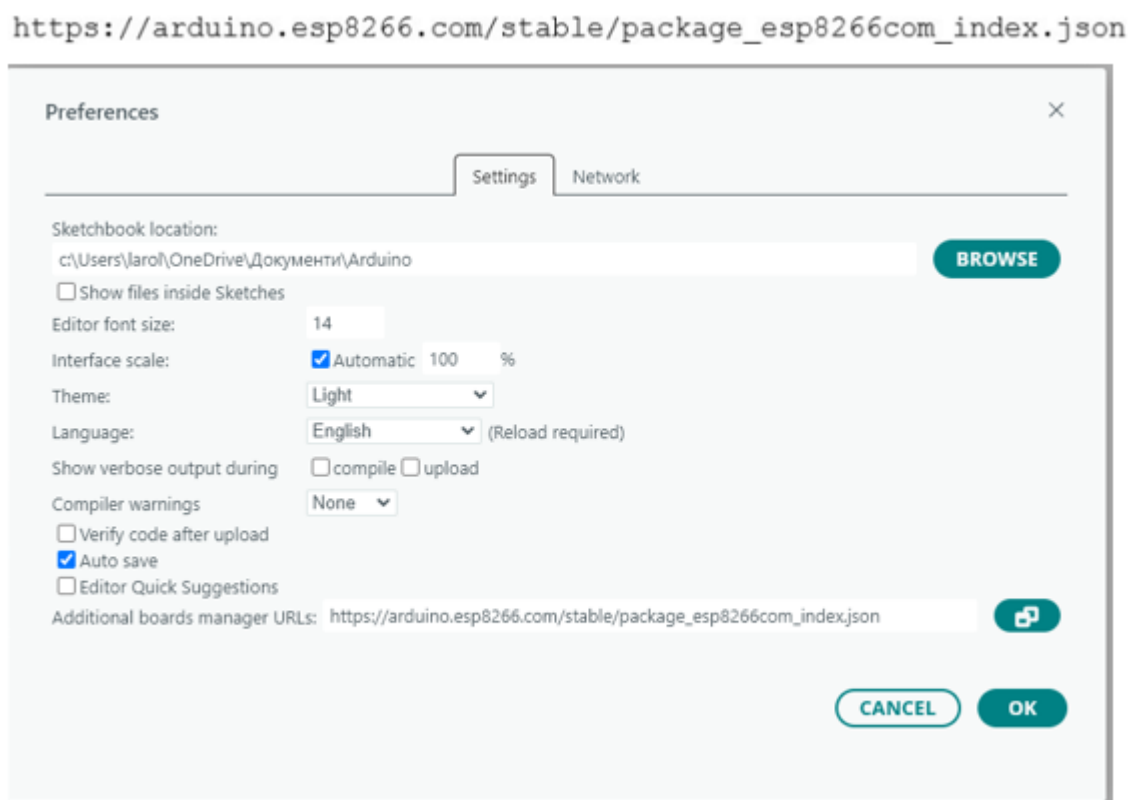


Рисунок 3.8 – Процес налаштування Arduino IDE

Після цього у вікні «Boards Manager», яке відкривається через меню Tools → Board, було здійснено пошук модуля esp8266 та виконано його

встановлення для забезпечення підтримки відповідної плати у середовищі Arduino IDE. (рис. 3.9).



Рисунок 3.9 – Процес встановлення модуля в Arduino IDE для роботи з платформами, які працюють на базі мікроконтролера ESP8266

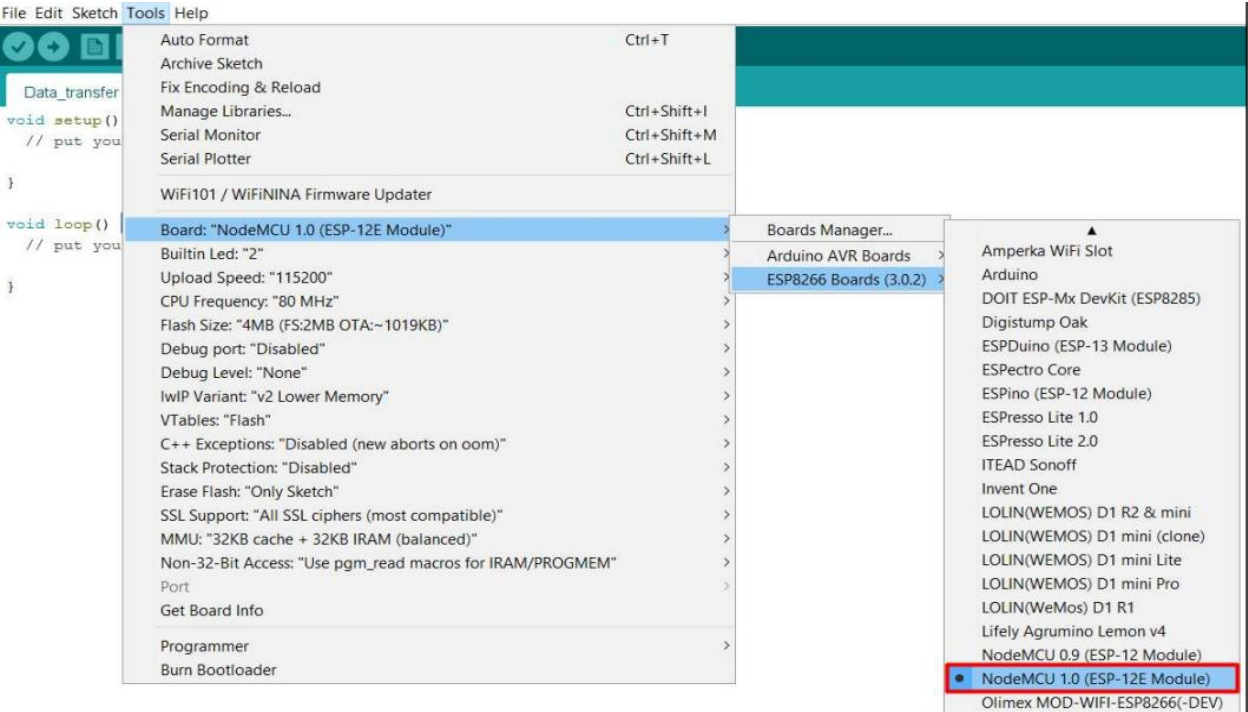


Рисунок 3.10 – Вибір платформи NodeMCU для проекту в Arduino IDE

Тепер в меню Board з'являється можливість обрати тип плати NodeMCU 1.0 для подальшого програмування (рис. 3.10).

3.5 Тестування прототипу комп'ютеризованої системи керування поштовою скринькою

На рисунку 3.11 представлено прототип комп'ютеризованої системи керування поштовою скринькою. Усі компоненти системи з'єднані між собою відповідно до розробленої електричної схеми з використанням макетної плати та монтажних дротів. Такий підхід дозволив зібрати повноцінну модель пристрою для тестування функціоналу, перевірки взаємодії між мікроконтролером ESP8266, сенсорами та виконавчими елементами, а також забезпечити зручність у подальшому налагодженні системи [17,18].

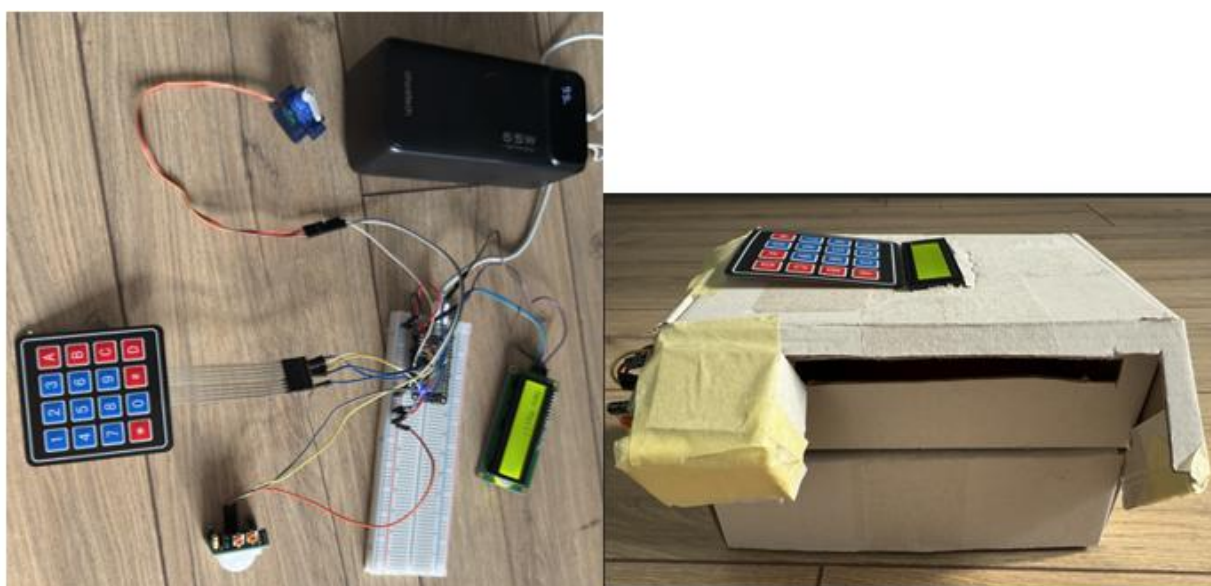


Рисунок 3.11 – Прототип системи

На рисунку 3.12 продемонстровано реалізацію однієї з ключових функцій розумної поштової скриньки – виявлення надходження кореспонденції та подальше автоматичне надсилення повідомлення користувачу через месенджер Telegram.



Рисунок 3.12 – Виявлення надходження кореспонденції

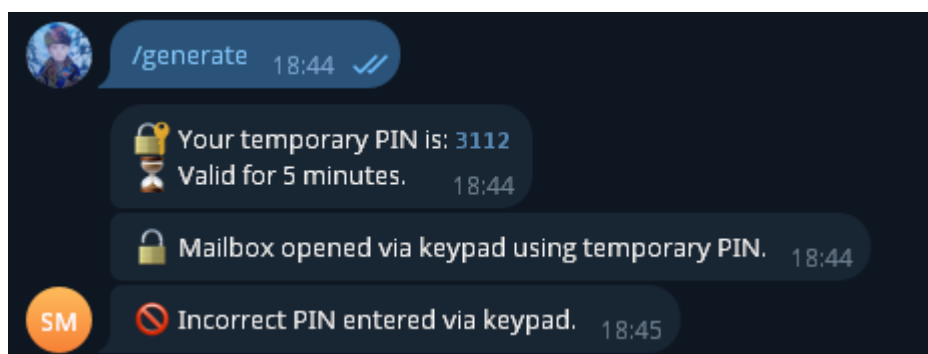


Рисунок 3.13 – Приклад взаємодії з телеграм-ботом

На рисунку 3.13 продемонстровано приклад взаємодії користувача з телеграм-ботом, який є частиною розумної системи поштової скриньки. Комунікація відбувається в реальному часі і включає кілька ключових етапів:

Система виявляє нове надходження кореспонденції, про що надсилає відповідне повідомлення в Telegram. Що свідчить про те що сенсор усередині скриньки зафіксував факт потрапляння посылки або листа.

Користувачу пропонується згенерувати тимчасовий PIN-код командою /generate, який слугуватиме одноразовим ключем для доступу до вмісту скриньки. Цей код має обмежений термін дії, що підвищує рівень безпеки системи. Після введення коду з клавіатури на пристрої, надходить підтвердження. Це означає, що доступ до скриньки було успішно отримано, і користувач відкрив її за допомогою отриманого одноразового пароля.

Окрім цього, система передбачає фіксацію та повідомлення про помилкові спроби введення коду. У випадку, якщо користувач або стороння особа вводить неправильний PIN-код, надходить сповіщення.

Цей механізм дозволяє власнику в реальному часі контролювати всі дії, пов'язані з доступом до поштової скриньки, і забезпечує базовий рівень безпеки. Повідомлення фіксуються в чаті, що також може використовуватись для ведення журналу подій та подальшого аналізу.

На рисунку 3.14 продемонстровано ще одну з функцій системи – дистанційне відкриття поштової скриньки через Telegram.



Рисунок 3.14 – Дистанційне відкриття

Користувач надсилає команду /open у чат-бот, після чого система одразу реагує повідомленням.

Це означає, що замок скриньки було відкрито за допомогою сервопривода на запит користувача без потреби вводити PIN-код вручну на клавіатурі. Така можливість особливо зручна у випадках, коли необхідно відкрити скриньку іншій довірєній особі або при виникненні проблем із локальним доступом.

Ця функція демонструє гнучкість і сучасність системи, дозволяючи управляти пристроєм з будь-якого місця, де є доступ до Інтернету, та забезпечує додаткову зручність для користувача.

На рисунку 3.15 зображено Telegram-журнал логів роботи комп'ютеризованої поштової скриньки, отриманий у відповідь на команду /getlog, а також очищення журналу завдяки команді /clearlog.



Рисунок 3.15 – Логи та їх очищення

					КС КРБ 123.243.00.00 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунку 3.16 зображено сценарій автоматичного очищення журналу подій через перевищення його допустимого розміру (ліміту в 2048 байт).

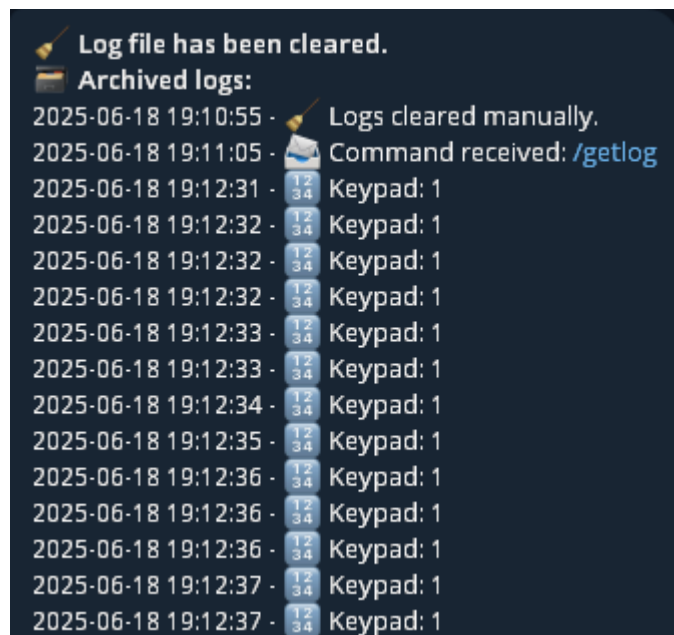


Рисунок 3.16 – Сценарій очищення через перевищення ліміту

Прототип комп'ютеризованої системи керування поштовою скринькою успішно пройшов усі заплановані тестування, підтвердивши свою функціональність та надійність. Система коректно фіксувала відкриття скриньки, надходження нової кореспонденції та взаємодію з клавіатурою завдяки інфрачервоним і контактним сенсорам. Авторизація користувача відбувалася через цифрову клавіатуру та за допомогою віддаленого доступу і всі спроби несанкціонованого доступу були виявлені та відповідним чином сповіщені.

Також ефективно працювали віддалені сповіщення через Telegram-бот, надсилаючи миттєві повідомлення про всі важливі події: надходження кореспонденції, відкриття скриньки та спроби некоректного введення PIN-коду. Інформаційний інтерфейс на OLED-дисплеї чітко відображав статус скриньки, результати авторизації, підказки та повідомлення про помилки.

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Ергономічні проблеми безпеки життєдіяльності.

У сучасних умовах цифровізації трудових процесів значна частина населення працює у сфері інтелектуальної діяльності, де основним інструментом взаємодії є комп'ютеризовані системи. У таких умовах ергономіка стає важливою складовою безпеки життєдіяльності, оскільки від правильного проектування робочого місця, програмного інтерфейсу та загального середовища залежить фізичне і психоемоційне здоров'я людини. Ігнорування ергономічних вимог призводить до підвищеного рівня захворюваності, зниження продуктивності праці та зростання ризику професійних ушкоджень.

Основною ергономічною проблемою офісного середовища є статичне навантаження на опорно-руховий апарат людини. Тривале перебування в сидячому положенні зазвичай поєднується з неадаптованими робочими меблями або неправильним розташуванням елементів робочого простору. Це спричиняє розвиток порушень постави, болю в шийі, плечах і спині, а також загальну м'язову втомлюваність. Згідно з результатами досліджень, більше ніж 60% користувачів комп'ютерів у середньому віці скаржаться на хронічні болі у попереку вже після 5 років роботи в офісі [19].

Окрему категорію ризиків становить надмірне зорове навантаження. Робота з монітором викликає напруження зорового аналізатора через надто дрібний шрифт, контрастне зображення, мерехтіння екрана та погане освітлення. Як наслідок, виникає синдром комп'ютерного зору, який включає подразнення очей, тимчасове погіршення зору, головні болі й зниження концентрації уваги. В умовах тривалого використання моніторів без

					КС КРБ 123.243.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Цейко В.Л.			Безпека життєдіяльності, основи охорони праці		
Перевірив		Жаровський Р.О.					
Консульт.		Пилипець М.І.					
Н. Контр.		Тиш Є.В.					
Затверд.		Осухівська Г.М.					
					Літ.	Арк.	Аркушів
						51	
					ТНТУ, каф. КС, гр. СІ-42		

організації перерв і відпочинку очей це стає хронічною проблемою. Ергономічні норми, зокрема міжнародні стандарти ISO 9241, рекомендують враховувати відстань до екрана, розмір символів та освітленість, однак на практиці ці вимоги часто ігноруються [20].

Ще один аспект – ергономіка програмного забезпечення. Інтерфейси, що не відповідають принципам зручності та інтуїтивності, змушують користувача виконувати зайві дії, викликають інформаційне перевантаження і сприяють підвищенню кількості помилок. Це особливо важливо у комп'ютеризованих системах, які мають керівну або критичну функцію, у системах обробки конфіденційної інформації, включаючи електронну пошту. Відсутність стандартизованих підходів до дизайну інтерфейсів, які враховували б когнітивні особливості користувачів, знижує загальний рівень ергономічної безпеки на робочому місці. Нерідко це також призводить до затримки виконання операцій, що критично при обробці часу чутливих запитів або повідомлень. Особливо це стосується користувачів із мінімальним досвідом або зниженими когнітивними функціями.

Психоемоційне навантаження є невіддільним наслідком поганої ергономіки. Постійна робота у стані дискомфорту, відсутність відчуття контролю над середовищем, надлишок інформації – усе це провокує стрес, тривожність, професійне вигорання. У контексті безпеки життєдіяльності такий стан підвищує ризик допущення помилок, зниження мотивації до роботи та навіть розвиток психосоматичних порушень. Згідно з міжнародними дослідженнями, у 30–50% випадків причиною зниження продуктивності праці у цифрових професіях є саме психоемоційні наслідки ергономічних недоліків [21].

Розв'язання цих проблем передбачає застосування інтегрованого підходу. Поєднання технічного забезпечення (ергономічні меблі, налаштоване ПЗ, правильне освітлення), організаційні заходи та навчальні програми. Важливим є постійний моніторинг і адаптація середовища до змін антропометричних та психофізіологічних характеристик працівників.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

4.2 Вплив електромагнітних полів (ЕМП) на людину та заходи щодо зменшення їх впливу на обслуговуючий персонал.

У сучасних офісних умовах під час створення та експлуатації комп'ютеризованих систем обробки електронної пошти працівники постійно перебувають у зоні дії електричних та магнітних полів, що виникають від моніторів, системних блоків, мережевих пристроїв і побутових розеток. Хоча інтенсивність таких полів зазвичай низька, тривале опромінення може мати накопичувальний ефект, тому заходи з гігієнічного контролю робочого середовища стають обов'язковими відповідно до ДСанПіН 3.3.6.096-2002 [22].

Генерація ЕМП на робочому місці відбувається на екстремально низьких частотах (50–60 Гц) від електромереж і комп'ютерних блоків, а також на радіочастотах (Wi-Fi, Bluetooth) – до сотень мегагерц. Рекспозиція до рівнів, що перевищують встановлені нормами – особливо у замкнутих приміщеннях з концентрованою технікою – може викликати субклінічні ефекти: зниження концентрації, втому, розлади сну та покращення тривожності. Водночас більшість сучасних офісів перебуває в межах рекомендацій, якщо дотримуються організаційно-технічних регламентів.

Згідно з рекомендаціями ICNIRP, межі безпечного впливу ЕМП змінюються залежно від частотного спектра. У межах 100 кГц–300 ГГц обмеження визначені з запасом безпеки для професійної роботи, а для населення – із додатковим коефіцієнтом. Ці стандарти враховують потенційні нагрівальні (теплові) та нетеплові ефекти, навіть якщо точні механізми останніх ще не повністю досліджені [23].

Наукові комісії, включно з WHO та IARC, класифікували радіочастоти як можливий канцероген (група 2B), проте ця оцінка спирається переважно на епідеміологічні спостереження при високих дозах. Останні звіти зазначають комп'ютерної техніки, основним ризиком вважається нервово-рефлекторне відсутність чітких доказів канцерогенності при офісному використанні подразнення та гострі симптоми, а не довгострокові ефекти. Для зменшення

					КС КРБ 123.243.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

навантаження ЕМП персонал дотримується технічних заходів: розміщення обладнання на відстані щонайменше 0,3–0,5 м від операторів; влаштування надійного заземлення; використання екранованих кабелів та корпусів з матеріалів із високим коефіцієнтом відбивання поля. Такі механізми ефективно мінімізують побічну емісію навіть у щільно заселеному середовищі [24].

Організаційний рівень включає обмеження тривалості безперервної роботи в зоні відчутних рівнів, чергування операторів між районами з різною інтенсивністю поля, а також попередню регулярно узгоджену інструкцію працівників щодо правил розміщення пристроїв і спостереження за симптомами надмірного опромінення.

Для технологічної безпеки проводяться контрольні вимірювання напруженості полів із застосуванням сертифікованих приладів, за методиками NIOSH. Результати фіксуються двічі на рік, із маркуванням "гарячих точок" – поруч із рядками PRS автоматизації, моніторів або серверів – для подальшого аналізу та оперативного коригування розміщення електрообладнання. При цьому враховується орієнтація випромінювачів у просторі, а також взаємодія джерел різних частот, що здатна створювати інтерференційні ефекти. Додатково проводиться оцінка сумарного впливу фонових і локальних ЕМП на типове робоче місце.

Персонал проходить навчання з практичних питань: як користуватися приладами контролю ЕМП, як визначити безпечне розташування пристроїв та кабелів, де звертатися за допомогою у випадку перевищення допустимого рівня поля. Наявність чітких інструкцій і практичних тренінгів підвищує свідомість і дозволяє швидко реагувати на нестандартні ситуації.

Комплексна модель безпеки, яка охоплює технічний, організаційний та освітній рівні, дозволяє утримувати рівні ЕМП у безпечному діапазоні. Це не лише зменшує ризик гострих реакцій та накопичувальних ефектів, а й забезпечує відповідність стандартам ICNIRP та рекомендацій НІОШ, підкріплюючи наукові підходи до технічної безпеки у цифрову епоху [23].

					КС КРБ 123.243.00.00 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У бакалаврській кваліфікаційній роботі здійснено розв’язання практичного інженерного завдання — створено комп’ютеризовану систему керування поштовою скринькою з можливістю віддаленого контролю за допомогою застосунку Telegram.

У процесі виконання кваліфікаційної роботи було досягнуто такі практичні результати:

1. У результаті огляду та аналізу існуючих рішень виявлено переваги застосування бездротових технологій зв’язку, зокрема Wi-Fi, для реалізації зручного та безпечного доступу до пристрою.

2. Обґрунтовано вибір апаратної платформи на базі мікроконтролера ESP8266, що забезпечує стабільне підключення до мережі та взаємодію з Telegram-ботом.

3. Проведено вибір необхідних компонентів системи: матричної клавіатури, PIR-сенсора, LCD-дисплея, сервоприводу, світлодіода та елементів інтерфейсної логіки.

4. Розроблено електричну принципову схему з урахуванням особливостей живлення, взаємозв’язків між модулями та захисту.

5. Складено алгоритм роботи системи.

6. Створено програмне забезпечення для контролю всіх функцій пристрою та налаштування взаємодії із Telegram-ботом.

7. За допомогою симулятора Wokwi виконано попереднє тестування логіки системи на віртуальній макетній платі, що дало змогу переконатися у правильності електричних з’єднань і програмної реалізації.

У результаті виконання кваліфікаційної роботи було виготовлено робочий дослідний макет системи розумної поштової скриньки. Проведене тестування показало, що пристрій успішно виконує всі закладені функції та характеризується зручністю використання.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Жаровський Р.О., Луцик Н.С., Осухівська Г.М., Паламар А.М., Тиш Є.В. Методичні вказівки до виконання кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 39 с.
2. Ясінський Р.В., Осухівська Г.М., Паламар А.М., Величко Д.В. Комп'ютерна система для контролю параметрів мікроклімату теплиць на основі інтернету речей. Актуальні задачі сучасних технологій : збірник тез доповідей XI міжнародної науково-практичної конференції молодих учених та студентів (Тернопіль, 7-8 грудня 2022 року), Тернопіль: ФОП Паляниця В. А., 2022. С. 177.
3. Буров Є.В., Митник М.М. Комп'ютерні мережі. Підручник. Том другий. Львів: «Магнолія 2006», 2024. 204 с.
4. Буров Є.В., Митник М.М. Комп'ютерні мережі. Підручник. Том перший. Львів: «Магнолія 2006», 2024. 333 с.
5. Микитишин А.Г., Митник М.М., Стухляк П.Д., Пасічник В.В. Комп'ютерні мережі. Книга 1. Львів: «Магнолія 2006», 2024. 256 с
6. Оконський М.В., Лупенко С.А., Паламар А.М. Комп'ютерна система для моніторингу метеорологічних параметрів на основі IoT. Актуальні задачі сучасних технологій: зб. тез доп. X Міжнар. наук.-практ. конф. – Тернопіль: ТНТУ, 2021. – С. 112.
7. ESP8266EX Datasheet URL: <https://octopart.com/datasheet/esp8266ex-espressif+systems-73281571> (дата звернення: 25.04.2025).
8. WiFi Плата NodeMCU V2 ESP8266 (CP2102) . URL: <https://arduino.ua/prod1495-wifi-plata-nodemcu-v2-esp8266-cp2102?srsId=AfmBOoogTXMByuylrbsLpTezPH9nS697R-tEEKJW07ZAVW8nBMaI4tJ7> (дата звернення: 15.04.2025).
9. HC-SR501 Інфрачервоний датчик руху URL: <https://myproject.com.ua/hc-sr501-infrachervonij-datchik-ruhu-ua.html?gclid=C>

					КС КРБ 123.243.00.00 ПЗ	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

jwKCAjwgb_CBhBMEiwA0p3oOAN1DZOd4oa7xFbNkk0gAyuY4pg0xbpZpk
DBgW1xUUcSuiCBP1INxhoCzgAQAvD_BwE&utm_source=google&utm_med
ium=cpc&utm_campaign=Fiksovana_Stavka (дата звернення: 09.04.2025).

10. Паламар М.І., Стрембіцький М.О., Паламар А.М. Проектування комп'ютеризованих вимірювальних систем і комплексів. Навчальний посібник. Тернопіль: ТНТУ. 2019. 150 с

11. Ясінський Р.В., Осухівська Г.М., Паламар А.М. Апаратно-програмна система для регулювання мікроклімату теплиць. Матеріали Х НТК «Інформаційні моделі, системи та технології» (7-8 грудня 2022 р.). – Тернопіль: ТНТУ, 2022. – С. 102.

12. Микитишин А. Г., Митник М. М., Стухляк П. Д. Телекомунікаційні системи та мережі. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2017. 384 с.

13. Lupenko S., Orobchuk O., Stadnik N., Zozulya A. Modeling and signals processing using cyclic random functions. In Proceedings of the 13th IEEE International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT). Lviv, Ukraine, 11–14 September, 2018. P. 360–366.

14. Свергун С., Жаровський Р. Тестування програмного забезпечення побудованого на мікросервісній архітектурі. Матеріали Х науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі системи та технології» (7-8 грудня 2022 року). Тернопіль: ТНТУ. 2022. С. 92.

15. Ясінський Р.В., Осухівська Г.М., Паламар А.М., Величко Д.В. Комп'ютерна система для контролю параметрів мікроклімату теплиць на основі інтернету речей. Актуальні задачі сучасних технологій : збірник тез доповідей XI міжнародної науково-практичної конференції молодих учених та студентів (Тернопіль, 7-8 грудня 2022 року), Тернопіль: ФОП Паляниця В. А., 2022. С. 177.

					КС КРБ 123.243.00.00 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

16. Ясінський Р.В., Осухівська Г.М., Паламар А.М. Апаратно-програмна система для регулювання мікроклімату теплиць. Матеріали X науково-технічної конференції «Інформаційні моделі, системи та технології» Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 7-8 грудня 2022 року), Тернопіль: ТНТУ, 2022. С. 102.

17. Слюз І., Жаровський Р. Критерії ефективності тестування комп'ютерної інформаційної системи. Матеріали XI Міжнародна науково-технічна конференція молодих учених та студентів «Актуальні задачі сучасних технологій» (7-8 грудня 2022 року). Тернопіль: ТНТУ. 2022. С. 174

18. Слюз І., Жаровський Р. Принципи та основні етапи комплексного тестування комп'ютерної інформаційної системи. Матеріали X науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі системи та технології» (7-8 грудня 2022 року). Тернопіль: ТНТУ. 2022. С. 93.

19. Sciencedirect – Ergonomic risk factors and musculoskeletal symptoms. URL: <https://www.sciencedirect.com/science/article/pii/S240584402401106X> (дата звернення: 26.05.2025).

20. ISO 9241 – Ergonomics of human–system interaction. URL: <https://www.iso.org/standard/77520.html>. (дата звернення: 01.06.2025).

21. Frontiers in Public Health – Ergonomic risks and worker fatigue. URL: <https://www.frontiersin.org/articles/10.3389/fpubh.2023.1253141/full> (дата звернення: 28.05.2025).

22. ДСанПіН 3.3.6.096-2002. Державні санітарні норми і правила при роботі з джерелами електромагнітних полів. Міністерство охорони здоров'я України. URL: <https://zakon.rada.gov.ua/laws/show/z0203-03#Text> (Дата звернення: 01.06.2025).

23. ICNIRP – Guidelines for limiting exposure to electromagnetic fields (100 kHz – 300 GHz). URL: <https://www.icnirp.org/en/publications/article/rf-guidelines-2020.html> (дата звернення: 29.05.2025).

					КС КРБ 123.243.00.00 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

24. IARC classifies Radiofrequency Electromagnetic Fields as possibly carcinogenic to humans URL:https://www.iarc.who.int/wp-content/uploads/2018/07/pr208_E.pdf (дата звернення: 29.05.2025).

Додаток А

Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

“Затверджую”

Завідувач кафедри КС

_____ Осухівська Г.М.

“ ____ ” _____ 2025 р

Комп'ютеризована система керування поштовою скринькою

ТЕХНІЧНЕ ЗАВДАННЯ

на 9 листках

Вид робіт:

Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»

«ВИКОНАВЕЦЬ»

Керівник кваліфікаційної роботи

Студент групи СІ-42

_____ к.т.н., доц. Жаровський Р.О.

_____ Цейко В.Л.

« ____ » _____ 2025 р.

« ____ » _____ 2025 р.

Тернопіль 2025

1 Загальні відомості

1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп'ютеризована система керування поштовою скринькою».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.243.00.00

1.2 Виконавець

Студент групи СІ-42, факультету комп'ютерно-інформаційних систем і програмної інженерії, кафедри комп'ютерних систем та мереж, Тернопільського національного технічного університету імені Івана Пулюя, Цейко В.Л.

1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№4/7-53 від 27.01.2025 р.)

1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 27.01.2025 р.

Плановий термін завершення виконання кваліфікаційної роботи – 23.06.2025 р.

1.5 Порядок оформлення та пред'явлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу здійснюється у відповідності до чинних норм та правил ISO, ЕСКД, ЕСПД та ДСТУ.

Пред'явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи. Попередній захист кваліфікаційної роботи відбувається при готовності роботи – наявності пояснювальної записки та графічного матеріалу.

Пред'явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

2 Призначення і цілі створення системи

2.1 Призначення системи

Комп'ютеризована система «Розумна поштова скринька» створена для автоматизованого контролю над отриманням поштових відправлень. Основу пристрою становить мікроконтролер ESP8266, який забезпечує з'єднання з мережею Wi-Fi та дозволяє реалізувати функції системи в межах концепції Інтернету речей (IoT).

Система призначена для виявлення ключових подій — відкриття скриньки та надходження листів чи посилок. Для цього використовуються відповідні сенсори: датчики положення для фіксації відкривання дверцят і сенсори вмісту для визначення появи кореспонденції. У разі спрацювання одного з датчиків користувач отримує миттєве повідомлення через Telegram-бот, що дає змогу бути поінформованим у режимі реального часу.

Окрім сповіщень, пристрій підтримує захист скриньки цифровим кодом доступу, який вводиться за допомогою клавіатури. Це дозволяє обмежити доступ до вмісту скриньки для сторонніх осіб.

2.2 Мета створення системи

Мета створення системи полягає в розробці інтелектуального пристрою для моніторингу взаємодії з поштовою скринькою з можливістю дистанційного інформування користувача про надходження кореспонденції та спроби доступу. Завданнями системи є забезпечення стабільного з'єднання з мережею Wi-Fi, інтеграція з Telegram для надсилання сповіщень у режимі реального часу, реалізація захисту доступу за допомогою цифрового коду, а також організація зручного інтерфейсу користувача з клавіатурним введенням і наочним виведенням інформації на дисплей.

2.3 Характеристика об'єкту

Розроблювана система призначена для використання в умовах приватного чи багатоквартирного житлового сектора з метою автоматизованого контролю стану поштової скриньки. Пристрій забезпечує фіксацію факту відкриття скриньки та надходження кореспонденції за допомогою сенсорів, а також надсилання сповіщень у Telegram у режимі реального часу.

3 Вимоги до системи

3.1 Вимоги до системи в цілому

3.1.1 Вимоги до структури та функціонування системи

Комп'ютеризована система керування поштовою скринькою повинна складатися з програмного забезпечення, що виконується на мікроконтролері ESP8266 із використанням бібліотек ESP8266WiFi, WiFiClientSecure, UniversalTelegramBot, FS та інших, необхідних для взаємодії з мережею та обробки сенсорних подій. До апаратних компонентів належать: датчик

присутності об'єкта (ІЧ) для фіксації надходження кореспонденції, клавіатура з цифровим кодом доступу, OLED-дисплей I²C 128×64 для локального відображення інформації

3.1.2 Вимоги до способів та засобів зв'язку між компонентами системи

Для забезпечення узгодженої роботи всіх модулів система використовує безпроводне з'єднання ESP8266 з мережею Wi-Fi за стандартом IEEE 802.11 b/g/n для надсилання повідомлень та отримання команд від Telegram-бота. Обмін даними між мікроконтролером і OLED-дисплеєм здійснюється за допомогою інтерфейсу I²C, а взаємодія з сенсорними елементами (герконами, датчиком наявності пошти, клавіатурою) відбувається через GPIO-піни мікроконтролера.

3.1.3 Вимоги до режимів функціонування системи

Для системи визначено два режими функціонування:

- нормальний режим функціонування;
- аварійний режим функціонування.

Основним режимом функціонування є нормальний режим.

Для забезпечення нормального режиму функціонування системи необхідно виконувати вимоги і дотримуватись умов експлуатації програмного забезпечення і комплексу технічних засобів системи, вказані у відповідних технічних документах (технічна документація, інструкції з експлуатації і т. д.).

Аварійний режим функціонування системи характеризується відмовою одного або декількох компонент програмного і (або) технічного забезпечення. При цьому функції роботи системи продовжують підтримувати роботу системи моніторингу в межах базових налаштувань.

3.1.4 Вимоги по діагностуванню системи

Для перевірки працездатності системи після вмикання необхідно виконати діагностичну послідовність дій. Після подачі живлення

мікроконтролер ESP8266 повинен автоматично підключитися до заданої Wi-Fi мережі та ініціалізувати всі необхідні компоненти: OLED-дисплей, PIR-сенсор руху, LED-індикацію та сервопривід.

Після завантаження на OLED-дисплеї має з'явитися повідомлення про успішне підключення до мережі. Для перевірки PIR-сенсора потрібно змодельовати рух у полі його дії — у відповідь система повинна показати повідомлення про виявлений рух на дисплеї, активувати LED-індикацію та надіслати відповідне сповіщення через Telegram-бота або інший канал оповіщення. Одночасно сервопривід має виконати задану дію.

3.1.5 Перспективи розвитку, проектування системи

У подальшому планується розширення функціональності системи шляхом впровадження підтримки додаткових сервісів та протоколів для підвищення зручності користувачів, зокрема інтеграції з іншими месенджерами та системами безпеки. Також передбачається модернізація апаратної частини за рахунок використання більш енергоефективних компонентів, оптимізації споживання енергії та збільшення автономності роботи за допомогою потужніших акумуляторів або альтернативних джерел живлення.

3.2 Показники призначення

Система має забезпечувати можливість масштабування програмного забезпечення при зміні конфігурації або додаванні нових функцій, таких як підтримка додаткових типів датчиків або нових режимів роботи. Масштабованість досягається за рахунок модульної архітектури коду та налаштовуваних параметрів конфігурації, що дозволяють гнучко адаптувати систему під різні потреби користувачів.

3.2.1 Вимоги до надійності

Система повинна забезпечувати стабільну роботу в автономному режимі не менше 4 годин завдяки вбудованому акумулятору. У разі втрати Wi-Fi-з'єднання мікроконтролер зобов'язаний виконати автоматичну повторну спробу підключення до трьох разів із затримкою 30 секунд між спробами. Помилки при взаємодії з Telegram API або внутрішніми сенсорами не повинні призводити до перезавантаження пристрою, а мають оброблятися із відображенням останніх коректно отриманих значень або статусів.

3.3 Вимоги до безпеки

Усі зовнішні елементи системи, що працюють під напругою, повинні бути надійно ізольовані для запобігання випадковому дотику користувача. Система живлення має бути оснащена засобами захисту від короткого замикання та перевантаження. Загальні вимоги до пожежної безпеки повинні відповідати нормам, встановленим для побутових електронних пристроїв, при цьому матеріали корпусу не повинні виділяти токсичних речовин під час займання.

3.3.1 Вимоги до експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи

Рекомендовані умови для експлуатації та зберігання пристрою: температура навколишнього середовища від $+0^{\circ}\text{C}$ до $+35^{\circ}\text{C}$, відносна вологість повітря в межах 30–80 %, атмосферний тиск 760 ± 25 мм рт. ст. Регламентне технічне обслуговування повинно проводитися не рідше одного разу на рік і включати перевірку справності клавіатурного блоку, сервоприводу, сенсорів та контроль якості паяння з'єднань.

3.4 Вимоги до захисту інформації від несанкціонованого доступу

Система не надає відкритого мережевого доступу до внутрішніх конфігурацій або серійного монітора, що значно знижує ймовірність несанкціонованого втручання. Усі налаштування зберігаються у внутрішній пам'яті модуля ESP8266 у вигляді простого конфігураційного файлу, доступ до якого можливий лише через фізичне з'єднання пристрою з ПК за допомогою USB та Arduino IDE.

3.4.1 Вимоги по збереженню інформації при аваріях

При аварійних ситуаціях система повинна перезавантажитися та успішно увімкнутися. Це забезпечить відновлення після перезавантаження або виходу із-за втрати живлення.

3.4.2 Вимоги по стандартизації і уніфікації

Система повинна відповідати вимогам ергономіки і зручності користування за умови комплектування високоякісним обладнанням (ЕОМ, монітор і інше обладнання), що має необхідні сертифікати відповідності і безпеки.

3.4.3 Вимоги до функцій (завдань), що виконуються системою:

Система повинна забезпечувати зручний інтерфейс користувача для швидкого перегляду поточних даних, зберігати останні отримані значення у внутрішній пам'яті, виконувати оперативний пошук інформації за регіонами та режимами, а також гарантувати високу швидкодію обробки та відображення даних.

4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

- пояснювальної записки;
- графічного матеріалу:
 - а) Структурна схема.
 - б) Блок схема роботи комп'ютеризованої системи.
 - в) Схема електрична принципова.
 - г) Схема електрична монтажна (з'єднань).

*Примітка: У комплект документації можуть вноситися зміни та доповнення в процесі розробки.

5 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи бакалавра

№ етапу	Назва етапу виконання кваліфікаційної роботи	Термін виконання
1	Розробка технічного завдання.	27.01.2025р. – 02.02.2025р.
2	Аналіз технічного завдання	15.02.2025р. – 27.03.2025р.
3	Проектна частина	06.04.2025р. – 02.05.2025р.
4	Практична частина	04.05.2025р. – 25.05.2025р.
5	Безпека життєдіяльності, основи охорони праці	26.05.2025р. – 02.06.2025р.
6	Оформлення пояснювальної записки і графічного матеріалу	03.06.2025р. – 10.06.2025р.
7	Перевірка на академічний плагіат, перевірка керівником та консультантами	9.06.2025р. – 15.06.2025р.
8	Попередній захист кваліфікаційної роботи бакалавра	16.06.2025р. – 22.06.2025р.
9	Захист кваліфікаційної роботи бакалавра	25.06.2025р.

6 Додаткові умови виконання кваліфікаційної роботи

Під час виконання кваліфікаційної роботи у дане технічне завдання можуть вноситися зміни та доповнення.

Додаток Б

Перелік елементів

Додаток В

Лістинг коду

```
#include <ESP8266WiFi.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Keypad.h>
#include <Servo.h>
#include <WiFiClientSecure.h>
#include <ArduinoJson.h>
#include <UniversalTelegramBot.h>
#include <FS.h>
#include <time.h>

// WiFi credentials
const char* ssid = "Wi-Fi_network_name";
const char* password = "Wi-Fi_network_pass";

// Telegram Bot Token
const char* telegramBotToken = "TELEGRAMTOKEN";

// Max number of users
const int MAX_USERS = 5;
String allowedChatIDs[MAX_USERS];
int userCount = 0;

// Pin definitions
#define PIR_PIN D8
#define SERVO_PIN D4

// LCD configuration
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Keypad configuration
const byte ROWS = 2;
const byte COLS = 3;
char keys[ROWS][COLS] = {
    {'1', '2', '3'},
    {'*', '0', '#'}
};
byte rowPins[ROWS] = {D6, D7};
byte colPins[COLS] = {D3, D0, D5};
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

// State variables
Servo servo;
String pinInput = "";
bool isUnlocked = false;
unsigned long unlockTime = 0;
bool pirTriggered = false;
```

```

// Temporary PIN
String tempPin = "";
bool tempPinActive = false;
unsigned long tempPinTime = 0;
const unsigned long TEMP_PIN_LIFETIME = 5 * 60 * 1000;

WiFiClientSecure secured_client;
UniversalTelegramBot bot(telegramBotToken, secured_client);

// Logging
const char* logFilePath = "/log.txt";
const size_t MAX_LOG_SIZE = 2048;

String getTimestamp() {
    time_t now = time(nullptr);
    struct tm* timeinfo = localtime(&now);
    char buffer[20];
    sprintf(buffer, "%04d-%02d-%02d %02d:%02d:%02d",
            timeinfo->tm_year + 1900,
            timeinfo->tm_mon + 1,
            timeinfo->tm_mday,
            timeinfo->tm_hour,
            timeinfo->tm_min,
            timeinfo->tm_sec);
    return String(buffer);
}

void logEvent(const String& event) {
    if (!SPIFFS.begin()) return;
    File logFile = SPIFFS.open(logFilePath, "a+");
    if (!logFile) return;

    if (logFile.size() > MAX_LOG_SIZE) {
        logFile.close();
        sendLogToTelegram("<b>🔪 Log file has been cleared.\n📁  
Archived logs:</b>\n");
        SPIFFS.remove(logFilePath);
        logFile = SPIFFS.open(logFilePath, "w");
        if (logFile) {
            logFile.println(getTimestamp() + " - 📁 Log file cleared  
due to size.");
        }
    } else {
        logFile.println(getTimestamp() + " - " + event);
    }
    logFile.close();
}

void sendLogToTelegram(String content) {
    if (!SPIFFS.begin()) return;
    File logFile = SPIFFS.open(logFilePath, "r");
    if (!logFile) return;

```

```

    while (logFile.available()) {
        content += logFile.readStringUntil('\n') + "\n";
    }
    logFile.close();
    sendTelegramMessage(content);
}

bool isUserAllowed(String chatID) {
    for (int i = 0; i < userCount; i++) {
        if (allowedChatIDs[i] == chatID) return true;
    }
    return false;
}

void addUser(String chatID) {
    if (!isUserAllowed(chatID) && userCount < MAX_USERS) {
        allowedChatIDs[userCount++] = chatID;
        Serial.print("New user added: "); Serial.println(chatID);
    }
}

void setup() {
    Serial.begin(115200);
    Serial.println("Booting...");
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println();
    Serial.print("Connected, IP: ");
    Serial.println(WiFi.localIP());

    secured_client.setInsecure();
    Wire.begin(D2, D1);
    lcd.begin(16, 2);
    lcd.backlight();
    servo.attach(SERVO_PIN);
    servo.write(0);

    pinMode(PIR_PIN, INPUT);
    lcd.setCursor(0, 0);
    lcd.print("Enter PIN:");

    configTime(0, 0, "pool.ntp.org", "time.nist.gov");
    SPIFFS.begin();
    logEvent("📧 Mailbox system started.");
}

void sendTelegramMessage(const String& msg) {
    for (int i = 0; i < userCount; i++) {
        if (bot.sendMessage(allowedChatIDs[i], msg, "HTML")) {

```

```

        Serial.println("☑ Message sent to: " +
allowedChatIDs[i]);
    } else {
        Serial.println("✗ Failed to send message to: " +
allowedChatIDs[i]);
    }
}
}

void handleTelegramMessages() {
    int numNewMessages = bot.getUpdates(bot.last_message_received
+ 1);
    while (numNewMessages) {
        for (int i = 0; i < numNewMessages; i++) {
            String chat_id = String(bot.messages[i].chat_id);
            String text = bot.messages[i].text;
            addUser(chat_id);
            text.trim();
            logEvent("✉ Command received: " + text);

            if (text == "/start") {
                bot.sendMessage(chat_id, "🔔 You are now subscribed to
Smart Mailbox alerts!", "HTML");
            } else if (text == "/generate" && isUserAllowed(chat_id))
{
                tempPin = "";
                for (int i = 0; i < 4; i++) tempPin += String(random(0,
4));
                tempPinActive = true;
                tempPinTime = millis();
                sendTelegramMessage("🔒 Your temporary PIN is: <code>" +
tempPin + "</code>\n⌚ Valid for 5 minutes.");
                logEvent("🔒 PIN generated and sent.");
            } else if (text == "/open" && isUserAllowed(chat_id)) {
                servo.write(180);
                lcd.clear();
                lcd.print("Opened by TG");
                isUnlocked = true;
                unlockTime = millis();
                sendTelegramMessage("📦 Mailbox unlocked remotely.");
                logEvent("📦 Mailbox opened via Telegram.");
            } else if (text == "/getlog") {
                sendLogToTelegram("<b>📖 Logs:</b>\n");
            } else if (text == "/clearlog" && isUserAllowed(chat_id))
{
                if (SPIFFS.begin()) {
                    SPIFFS.remove(logFilePath);
                    File newLog = SPIFFS.open(logFilePath, "w");
                    if (newLog) {
                        newLog.println(getTimestamp() + " - ✍ Logs cleared

```

```

manually.");
        newLog.close();
    }
    sendTelegramMessage("🔪 Log file has been cleared.");
}
}
}
numNewMessages = bot.getUpdates(bot.last_message_received +
1);
}
}

void loop() {
    char key = keypad.getKey();
    if (key) {
        logEvent("🔑 Keypad: " + String(key));
        if (key == '#') {
            if ((tempPinActive && pinInput == tempPin)) {
                lcd.clear();
                lcd.print("Access OK");
                servo.write(180);
                isUnlocked = true;
                unlockTime = millis();
                tempPinActive = false;
                sendTelegramMessage("🔒 Mailbox opened via keypad using
temporary PIN.");
                logEvent("🔒 Opened with correct PIN.");
            } else {
                lcd.clear();
                lcd.print("Wrong PIN");
                sendTelegramMessage("🚫 Incorrect PIN entered via
keypad.");
                logEvent("❌ Wrong PIN attempt.");
            }
            pinInput = "";
            delay(1500);
            lcd.clear();
            lcd.print("Enter PIN:");
        } else if (key == '*') {
            pinInput = "";
            lcd.setCursor(0, 1);
            lcd.print("Cleared          ");
            delay(1000);
            lcd.clear();
            lcd.print("Enter PIN:");
            logEvent("🔒 PIN cleared.");
        } else {
            pinInput += key;
            lcd.setCursor(0, 1);
            lcd.print("          ");
            lcd.setCursor(0, 1);

```

```

        for (int i = 0; i < pinInput.length(); i++) {
            lcd.print("*");
        }
    }

    if (isUnlocked && millis() - unlockTime > 5000) {
        servo.write(0);
        isUnlocked = false;
        lcd.clear();
        lcd.print("Enter PIN:");
        logEvent("🔒 Mailbox auto-locked.");
    }

    static unsigned long lastMotionTime = 0;
    if (!pirTriggered && digitalRead(PIR_PIN) == HIGH && millis()
- lastMotionTime > 10000) {
        lastMotionTime = millis();
        pirTriggered = true;
        lcd.clear();
        lcd.print("New Mail!");
        sendTelegramMessage("📧 New mail detected!\nUse /generate to
receive a one-time PIN.");
        logEvent("📧 Motion detected - new mail.");
        delay(2000);
        lcd.clear();
        lcd.print("Enter PIN:");
    }
    if (digitalRead(PIR_PIN) == LOW) {
        pirTriggered = false;
    }

    static unsigned long lastCheck = 0;
    if (millis() - lastCheck > 5000) {
        lastCheck = millis();
        handleTelegramMessages();
    }

    if (tempPinActive && millis() - tempPinTime >
TEMP_PIN_LIFETIME) {
        tempPinActive = false;
        logEvent("⌚ PIN expired.");
    }
}

```