

# КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

*бакалавр*

(назва освітнього ступеня)

на тему: *Комп'ютерна система безконтактного керування IoT-пристроями  
на основі розпізнавання та інтерпретації жестів*

Виконав: студент 4 курсу, групи СІ-42

спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

(підпис)

*Ткачук С.А.*

(прізвище та ініціали)

Керівник

(підпис)

*Паламар А.М.*

(прізвище та ініціали)

Нормоконтроль

(підпис)

*Тихо Є.В.*

(прізвище та ініціали)

Завідувач кафедри

(підпис)

*Осухівська Г.М.*

(прізвище та ініціали)

Рецензент

(підпис)

*Деркач М.В.*

(прізвище та ініціали)

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)

«24» квітня 2026 р

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня бакалавр

(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

студенту Ткачуку Сергію Андрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютерна система безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів

Керівник роботи Паламар Андрій Михайлович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «24» квітня 2026 року № 4/9-188

2. Термін подання студентом завершеної роботи 15.06.2026 р.

3. Вихідні дані до роботи Технічне завдання

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

Вступ

1. Аналіз технічного завдання

2. Проєктна частина

3. Практична частина

4. Безпека життєдіяльності, основи охорона 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

4. Результати моделювання системи



## АНОТАЦІЯ

Ткачук С.А. Комп'ютерна система безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів : робота на здобуття освітнього ступеня бакалавра: спец. 123 — комп'ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2026.

Ключові слова: IoT-технології, безконтактне керування, розпізнавання жестів, комп'ютерна система, мікроконтролер, сенсори.

Кваліфікаційна робота присвячена розробці комп'ютерної системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів користувача. Актуальність роботи зумовлена зростаючою потребою у зручних, інтуїтивних інтерфейсах взаємодії з інтелектуальними пристроями в межах концепції Інтернету речей.

У першому розділі проведено аналіз технічного завдання, сформульовано основні функціональні вимоги до системи, а також виконано огляд і порівняльний аналіз існуючих аналогів і підходів до реалізації безконтактного керування IoT-пристроями.

У другому розділі розроблено апаратне забезпечення системи, зокрема структурну схему та електричну принципову схему. Обґрунтовано вибір елементної бази з урахуванням вимог до точності розпізнавання, енергоефективності, надійності та вартості.

Третій розділ присвячено розробленню алгоритму роботи системи та програмного забезпечення. Виконано інтеграцію з IoT-платформою для передавання й обробки даних, а також проведено тестування працездатності та коректності функціонування розробленої системи. Отримані результати підтверджують доцільність обраних технічних рішень.

## ANNOTATION

Tkachuk S.A. Computer System for Contactless Control of IoT Devices Based on Gesture Recognition and Interpretation. Bachelor's Graduation Thesis: speciality 123 — Computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2026.

Keywords: IoT technologies, contactless control, gesture recognition, computerized system, microcontroller, sensors.

The qualification work is devoted to the development of a computer system for contactless control of IoT devices based on gesture recognition and interpretation. The relevance of the research is driven by the growing demand for intuitive, user-friendly interaction methods with smart devices within the Internet of Things paradigm.

In the first chapter, the technical specification is analyzed, the functional requirements of the system are defined, and a review and comparative analysis of existing analogs and approaches to contactless IoT control are conducted.

The second chapter focuses on the development of the system's hardware. A structural diagram and an electrical schematic diagram are designed, and the selection of the element base is justified considering accuracy, energy efficiency, reliability, and cost.

The third chapter is devoted to the development of the system algorithm and software. Integration with an IoT platform for data transmission and processing is implemented, and system testing is carried out to verify functionality and performance. The obtained results confirm the effectiveness and feasibility of the proposed technical solutions.

## ЗМІСТ

СПИСОК СКОРОЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ.....	10
1.1 Огляд та аналіз сфер застосування системи безконтактного керування пристроями за допомогою жестів.....	10
1.2 Аналіз вимог до комп'ютерної системи безконтактного керування пристроями за допомогою жестів.....	12
1.3 Огляд існуючих засобів для безконтактного керування пристроями.....	13
1.4 Аналіз можливих рішень поставленого завдання.....	17
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА .....	20
2.1 Структура системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів .....	20
2.2 Розроблення апаратного забезпечення безконтактного керування IoT-пристроями.....	22
2.2.1 Мікроконтролерний модуль TTGO ESP32 .....	22
2.2.2 Модуль давача жестів APDS-9960 .....	26
2.2.3 Модуль з RGB світлодіодами WS2812 .....	28
2.2.4 Сервопривід SG90 .....	31
2.2.5 Модуль реле .....	33
2.3 Розроблення електричної принципової схеми .....	35
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА .....	38
3.1 Алгоритм роботи системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів.....	38
3.2 Розробка програмного забезпечення.....	41
3.2.1 Ініціалізація апаратних та програмних ресурсів системи.....	41
3.2.2 Основний цикл роботи системи та обробка подій.....	43

					<i>КС КРБ 123.196.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ткачук С.А.</i>			<i>Комп'ютерна система безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірів</i>		<i>Паламар А.М.</i>					5	82
<i>Рецензент</i>		<i>Деркач М.В.</i>				<i>ТНТУ, каф. КС, гр. СІ-42</i>		
<i>Н. Контр.</i>		<i>Тиш Е.В.</i>						
<i>Зав. каф.</i>		<i>Осунівська Г.М.</i>						

3.2.3 Реалізація підсистеми відображення інформації на дисплеї .....	44
3.2.4 Реалізація обробки команд від IoT платформи Vlynk .....	46
3.2.5 Реалізація алгоритму обробки жестів та формування керуючих дій .....	47
3.3 Налаштування хмарної IoT платформи Vlynk.....	50
3.4 Результати моделювання та тестування системи .....	52
<b>РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ .....</b>	<b>56</b>
4.1 Ергономічні проблеми безпеки життєдіяльності.....	56
4.2 Соціальне значення охорони праці .....	59
<b>ВИСНОВКИ.....</b>	<b>62</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>63</b>
Додаток А Технічне завдання	
Додаток Б Перелік елементів	
Додаток В Лістинг програми	

					<i>КС КРБ 123.196.00.00 ПЗ</i>	Арк.
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		6

## СПИСОК СКОРОЧЕНЬ

APDS – Advanced Proximity and Gesture Sensor

GPIO – General Purpose Input/Output

I2C – Inter-Integrated Circuit

IoT – Internet of Things

PWM – Pulse Width Modulation

RGB – Red Green Blue

TFT – Thin Film Transistor

ВМ – виконавчий механізм

КСБК – комп'ютерна система безконтактного керування

МКМ – мікроконтролерний модуль

ЦКП – цифровий канал передачі

					КС КРБ 123.196.00.00 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВСТУП

Стрімкий розвиток Інтернету речей (IoT) зумовлює широке впровадження інтелектуальних пристроїв у побуті, промисловості, медицині та системах «розумного середовища». Більшість таких пристроїв орієнтовані на взаємодію з користувачем через традиційні засоби керування, зокрема кнопки, сенсорні екрани або мобільні додатки. Однак зазначені підходи не завжди є зручними, інтуїтивними та безпечними, особливо в умовах підвищених вимог до гігієни, доступності та швидкості керування.

У зв'язку з цим актуальною є задача розроблення безконтактних інтерфейсів взаємодії, які забезпечують керування IoT-пристроями без фізичного контакту з органами керування. Використання технологій розпізнавання та інтерпретації жестів дозволяє підвищити ергономічність, зменшити залежність від зовнішніх пристроїв керування та розширити можливості інтелектуальних систем. Незважаючи на наявність окремих рішень у цій галузі, проблема створення доступних, масштабованих та енергоефективних комп'ютерних систем безконтактного керування IoT-пристроями залишається актуальною і потребує подальших досліджень та практичної реалізації.

Метою кваліфікаційної роботи є розроблення комп'ютерної системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів, а також обґрунтування апаратних і програмних рішень, необхідних для її реалізації.

Для досягнення поставленої мети в роботі необхідно розв'язати такі задачі:

- проаналізувати технічне завдання та сформулювати вимоги до комп'ютерної системи безконтактного керування;
- виконати огляд і порівняльний аналіз існуючих аналогів і методів керування IoT-пристроями на основі жестів;
- розробити структурну схему та електричну принципову схему апаратної частини системи;

					КС КРБ 123.196.00.00 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

- обґрунтувати вибір елементної бази з урахуванням функціональних і технічних вимог;
- розробити алгоритм роботи системи та створити програмне забезпечення;
- реалізувати інтеграцію розробленої системи з IoT-платформною;
- провести тестування системи та проаналізувати результати її функціонування.

					<i>КС КРБ 123.196.00.00 ПЗ</i>	<i>Арк.</i>
						9
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

## РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

### 1.1 Огляд та аналіз сфер застосування системи безконтактного керування пристроями за допомогою жестів

Розвиток комп'ютерних систем керування та широке впровадження IoT технологій зумовлюють зростання вимог до способів взаємодії людини з технічними засобами. Традиційні інтерфейси керування, засновані на використанні механічних кнопок, сенсорних панелей або мобільних додатків, не завжди забезпечують необхідний рівень зручності, оперативності та безпеки. У цьому контексті безконтактні системи керування, зокрема на основі розпізнавання та інтерпретації жестів, набувають дедалі більшої актуальності та знаходять застосування в різних сферах.

Однією з основних сфер застосування комп'ютерної системи безконтактного керування є побутова автоматизація та системи «розумного дому». У таких середовищах жести можуть використовуватися для керування освітленням, кліматичними установками, мультимедійними пристроями або системами безпеки. Безконтактне керування підвищує комфорт користувача, зменшує кількість фізичних взаємодій з поверхнями та забезпечує більш інтуїтивний спосіб керування пристроями, особливо для користувачів без спеціальної технічної підготовки.

Важливе значення такі системи мають у медичній сфері та закладах охорони здоров'я, де дотримання санітарно-гігієнічних вимог є критично важливим. Безконтактне керування медичним обладнанням, інформаційними панелями або допоміжними пристроями дозволяє мінімізувати ризик перехресного забруднення та підвищити безпеку персоналу й пацієнтів. Аналогічні підходи можуть

					<i>КС КРБ 123.196.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ткачук С.А.</i>			<i>Аналіз технічного завдання</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Паламар А.М.</i>					<i>10</i>	<i>10</i>
<i>Рецензент</i>		<i>Деркач М.В.</i>				<i>ТНТУ, каф. КС, гр. СІ-42</i>		
<i>Н. Контр.</i>		<i>Тиш Е.В.</i>						
<i>Зав. каф.</i>		<i>Осужівська Г.М.</i>						

застосовуватися в лабораторіях і чистих приміщеннях, де небажаний прями́й контакт з елементами керування.

Перспективною сферою застосування є промислова автоматизація та виробничі процеси. У виробничих умовах безконтактні системи керування можуть використовуватися для управління обладнанням у ситуаціях, коли руки оператора зайняті або використання традиційних органів керування ускладнене через захисні засоби, підвищену вологість, пил чи інші несприятливі фактори. Інтеграція жестових інтерфейсів з IoT-пристроями дає змогу підвищити оперативність керування та зменшити кількість допоміжних пристроїв.

Окрему увагу слід приділити застосуванню комп'ютерних систем безконтактного керування в транспорті та інфраструктурних об'єктах. У транспортних засобах жести можуть використовуватися для керування мультимедійними системами або допоміжними функціями без відволікання водія від керування. В інтелектуальних будівлях і громадських закладах такі системи сприяють створенню адаптивних середовищ, що реагують на дії користувача в реальному часі.

Крім того, комп'ютерні системи на основі розпізнавання жестів знаходять застосування в освітній сфері, інтерактивних навчальних середовищах та демонстраційних установках. Вони дозволяють реалізувати нові форми взаємодії з навчальними матеріалами та технічними засобами, підвищуючи залученість користувачів і ефективність навчального процесу.

Аналіз наведених сфер застосування свідчить про універсальність та високий потенціал комп'ютерної системи безконтактного керування IoT-пристроями. Широкий спектр можливих сценаріїв використання зумовлює необхідність створення гнучкої, масштабованої та адаптивної системи, здатної працювати в різних умовах експлуатації та відповідати сучасним вимогам до функціональності, надійності й зручності використання.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

## 1.2 Аналіз вимог до комп'ютерної системи безконтактного керування пристроями за допомогою жестів

Насамперед до комп'ютерної системи безконтактного керування IoT-пристроями висуваються функціональні вимоги, пов'язані з реалізацією безконтактного керування. Система повинна забезпечувати стабільне зчитування даних про жести користувача, їх попередню обробку, розпізнавання та коректну інтерпретацію у вигляді керуючих команд для IoT-пристроїв. Важливою вимогою є робота в реальному часі з мінімальними затримками, що забезпечує природність і зручність взаємодії користувача з системою.

Не менш важливими є нефункціональні вимоги, серед яких ключове місце займають вимоги до надійності та точності. Система повинна коректно функціонувати впродовж тривалого часу без збоїв, бути стійкою до перешкод, шумів і помилкових спрацьовувань. Точність розпізнавання жестів має бути достатньою для запобігання неправильному виконанню команд, що особливо актуально при керуванні кількома IoT-пристроями або складними сценаріями автоматизації.

Окрему групу становлять вимоги до апаратного забезпечення. Мікроконтролери та сенсорні модулі повинні мати достатні обчислювальні ресурси, підтримку необхідних інтерфейсів зв'язку та відповідати вимогам енергоефективності. Компактність і модульність апаратної частини є важливими для забезпечення можливості інтеграції системи в різні середовища та її подальшого масштабування.

Вимоги до програмного забезпечення передбачають реалізацію гнучкої та масштабованої архітектури, яка дозволяє змінювати алгоритми розпізнавання, налаштовувати параметри системи та інтегрувати її з зовнішніми IoT-платформами. Програмне забезпечення повинно забезпечувати стабільний обмін даними, зберігання та візуалізацію інформації, а також підтримувати віддалений доступ і моніторинг стану системи.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

Значну увагу необхідно приділити вимогам до сумісності та інтеграції. Система має підтримувати стандартні протоколи зв'язку та бути сумісною з поширеними IoT-платформами, що розширює можливості її практичного застосування. Крім того, важливо враховувати вимоги до зручності використання, які передбачають інтуїтивність керування, мінімальні налаштування з боку користувача та адаптацію до різних сценаріїв експлуатації.

Отже, комплексний аналіз вимог до комп'ютерної системи безконтактного керування IoT-пристроями створює основу для обґрунтованого вибору апаратних і програмних рішень та забезпечує досягнення поставлених цілей розробки.

### 1.3 Огляд існуючих засобів для безконтактного керування пристроями

У сучасному технологічному середовищі безконтактні інтерфейси взаємодії з пристроями на основі розпізнавання жестів стають дедалі популярнішими. Такі рішення застосовуються в різних галузях – від розумного дому і побутових IoT-систем до автоматизації виробничих процесів і транспортних засобів. Різноманітні апаратні та програмні засоби, що вже представлені на ринку або у вигляді науково-прикладних прототипів, демонструють потенціал жестового керування, проте водночас виявляють низку суттєвих недоліків та обмежень, які обґрунтовують потребу в подальшому розвитку відповідних рішень [1].

Одним із прикладів комерційних продуктів, що використовують розпізнавання жестів для безконтактної взаємодії з пристроями, є системи керування жестами, інтегровані в сучасні автомобілі преміум-класу [2]. Вони дозволяють водієві управляти мультимедійними та навігаційними функціями без фізичного контакту з елементами керування, що підвищує рівень безпеки та зручності (рис. 1.1). Однак ці рішення, як правило, прив'язані до конкретної платформи та дорогого обладнання, що обмежує їх доступність та гнучкість для широкого кола користувачів й IoT-сценаріїв малої вартості.

					КС КРБ 123.196.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13



Рисунок 1.1 – Система керування жестами, інтегрована в сучасні автомобілі

Існують також апаратні контролери, що використовують датчі руху або для розпізнавання жестів у просторі, як у системі GestureKeeper, яка здатна визначати початок та тип жесту за допомогою аналізу часових рядів даних з акселерометрів [3]. Такі рішення демонструють відносно високу точність розпізнавання на рівні понад 90 %, але вони потребують носіння спеціальних сенсорних пристроїв або браслетів, що знижує загальну зручність використання порівняно з безконтактними методами, які не вимагають одягання додаткового обладнання (рис. 1.2).



Рисунок 1.2 – Портативний контролер жестів GestureKeeper

					КС КРБ 123.196.00.00 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

Більш універсальні підходи ґрунтуються на використанні комп'ютерного зору – аналізу відеопотоку з камер для виділення та інтерпретації жестів рук у реальному часі. Такі методи застосовуються в академічних розробках і демонструють високу гнучкість: вони можуть працювати з загальнодоступними камерами без потреби в додаткових давачах [4]. Наприклад, у роботах, що розглядають обробку зображень для дистанційного керування пристроями, використовується бібліотека MediaPipe від Google, яка дозволяє розпізнавати ключові точки руки та перетворювати їх у команди для системи керування (рис. 1.3).

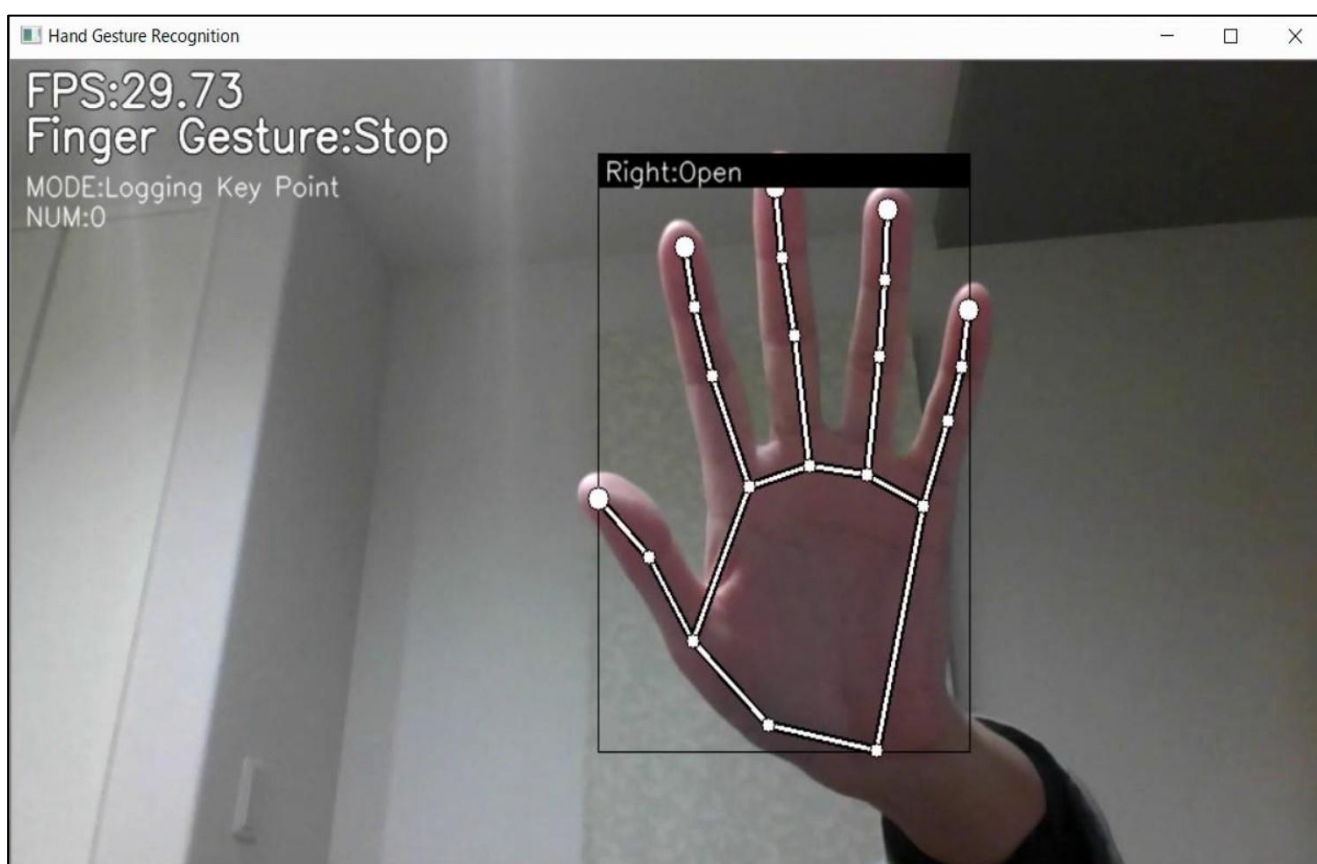


Рисунок 1.3 – Система розпізнавання жестів рук на основі комп'ютерного зору

Подібні рішення мають важливу перевагу – можливість адаптації під різні сценарії використання, від розумного будинку до промислових систем. Проте вони вразливі до змін умов зовнішнього освітлення, складного фону та різних положень руки, що суттєво впливає на точність та стабільність розпізнавання.

Ще однією проблемою сучасних систем, що базуються на відеоаналізі, є їх висока ресурсомісткість. Обробка відеопотоку з високою частотою кадрів і точністю розпізнавання вимагає значних обчислювальних потужностей, що обмежує застосування таких рішень на енергоефективних або вбудованих платформах без потужних графічних чи нейронних процесорів. Це робить їх менш придатними для IoT-пристроїв зі строгими обмеженнями енергоспоживання та обчислювальних ресурсів, наприклад для автономних сенсорних модулів у розумному домі чи побутових пристроїв.

Комерційна продукція на ринку також включає спрощені жести для керування окремими пристроями, наприклад жест для запуску дрона або активації певної функції камери [5]. Такі рішення, як у деяких моделях безпілотних літальних апаратів, дозволяють одним фрагментарним рухом ініціювати просту дію, але не можуть забезпечити широкий набір команд чи складні сценарії взаємодії через обмежений набір жестів, які можуть бути розпізнаними. Це демонструє, що багато існуючих комерційних реалізацій прагнуть забезпечити певну маркетингову функцію, а не повноцінний інтерфейс загального призначення.

Також слід зазначити, що наукові та академічні проєкти часто спрямовані на розробку концептуальних прототипів або методологій, які не завжди мають завершену інженерну реалізацію. Наприклад, деякі роботи пропонують алгоритмічні підходи до класифікації жестів із застосуванням складних математичних моделей, таких як приховані марковські моделі для траєкторій руху пальців, що надає високі показники точності на тестових наборах даних, але їх практична реалізація у вигляді готового продукту є складною та ресурсомісткою [6].

Усі вищезгадані засоби мають як суттєві переваги – інтуїтивність, потенціал безконтактної взаємодії, гнучкість – так і значні недоліки:

– обмеженість набору розпізнаваних жестів та неможливість масштабування під різні сценарії;

					КС КРБ 123.196.00.00 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

- висока залежність від умов зовнішнього середовища, таких як освітлення, фон та положення користувача;
- висока обчислювальна складність та ресурсна вимогливість, що обмежує ефективність на IoT-платформі;
- неуніверсальність апаратних рішень, що вимагають носіння спеціальних давачів;
- висока вартість розробки та інтеграції повноцінних систем, особливо для малого бізнесу.

Отже, огляд існуючих засобів для безконтактного керування пристроями показує, що жодне з існуючих рішень не забезпечує одночасно високого рівня точності розпізнавання, універсальності застосування, низьких обчислювальних вимог, доступної вартості та повної адаптованості до умов IoT-середовища. Це створює науково-прикладну потребу в розробці нової комп'ютерної системи, що поєднуватиме переваги сучасних підходів та мінімізуватиме їхні недоліки, забезпечуючи підвищену надійність, гнучкість та ефективність у практичних сценаріях використання.

#### 1.4 Аналіз можливих рішень поставленого завдання

Розв'язання задачі створення комп'ютерної системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів може бути реалізоване різними технічними та програмними підходами. Вибір оптимального рішення потребує аналізу можливих варіантів реалізації з урахуванням вимог технічного завдання, обмежень апаратних ресурсів, умов експлуатації та перспектив подальшого розвитку системи.

Одним із можливих рішень є використання оптичних систем розпізнавання жестів, що базуються на аналізі відеопотоку з камери. Такий підхід дозволяє реалізувати безконтактне керування без необхідності використання додаткових носимих пристроїв і забезпечує відносно широкий набір розпізнаваних жестів. Оптичні методи є гнучкими та легко масштабуються програмними засобами,

					<i>КС КРБ 123.196.00.00 ПЗ</i>	<i>Арк.</i>
						17
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

зокрема шляхом застосування бібліотек комп'ютерного зору. Водночас вони мають суттєві обмеження, пов'язані з вимогами до обчислювальних ресурсів, чутливістю до освітлення та фону, а також складністю реалізації на енергоефективних вбудованих платформах, що обмежує їх застосування в автономних IoT-системах.

Іншим підходом є використання сенсорних рішень на основі інерційних вимірювальних модулів, ультразвукових або інфрачервоних датчиків для фіксації рухів руки. Такі системи характеризуються меншою залежністю від зовнішніх умов та стабільнішою роботою в різних середовищах. Вони добре підходять для реалізації на мікроконтролерних платформах з обмеженими ресурсами та забезпечують достатню швидкодію. Недоліком цього підходу є обмежена кількість жестів, які можуть бути коректно розпізнані, а також зниження природності взаємодії порівняно з оптичними методами.

Перспективним напрямом є комбінований підхід, який передбачає використання кількох типів сенсорів та поєднання різних методів обробки даних. Такий варіант дозволяє компенсувати недоліки окремих рішень і підвищити надійність та точність розпізнавання жестів. Проте реалізація комбінованих систем ускладнює апаратну частину, збільшує вартість та потребує більш складних алгоритмів обробки даних.

З точки зору програмної реалізації можливими є рішення на основі класичних алгоритмів обробки сигналів і простих методів класифікації, а також використання елементів машинного навчання. Класичні алгоритми мають перевагу у вигляді передбачуваної роботи та низьких обчислювальних витрат, однак вони обмежені в адаптації до індивідуальних особливостей користувача. Алгоритми машинного навчання дозволяють підвищити точність та адаптивність системи, але потребують додаткових обчислювальних ресурсів і етапу навчання моделей.

З урахуванням вимог до енергоефективності, вартості та можливості інтеграції з IoT-платформами доцільним є використання мікроконтролерної платформи з вбудованими засобами бездротового зв'язку та підтримкою

					КС КРБ 123.196.00.00 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

стандартних протоколів обміну даними. Таке рішення забезпечує баланс між функціональністю, складністю реалізації та практичною придатністю системи.

Отже, аналіз можливих рішень показує, що оптимальним підходом до реалізації поставленого завдання є використання гнучкої мікроконтролерної системи з сенсорним модулем розпізнавання жестів, ефективними алгоритмами обробки даних та інтеграцією з IoT-платформою. Це дозволяє створити універсальну, масштабовану та економічно доцільну систему безконтактного керування, що відповідає сучасним вимогам і умовам експлуатації.

					<i>КС КРБ 123.196.00.00 ПЗ</i>	<i>Арк.</i>
						19
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

## РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

### 2.1 Структура системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів

На рисунку 2.1 наведено структурну схему системи, яка відображає загальну організацію взаємодії між сенсорними, обчислювальними, виконавчими та комунікаційними компонентами.

Центральним елементом системи є мікроконтролерний модуль, який виконує функції збору, обробки та аналізу даних, а також формування керуючих сигналів для виконавчих пристроїв. До входу мікроконтролера підключено давач жестів, який забезпечує безконтактне зчитування рухів користувача у вигляді дискретних команд. Отримані від сенсора дані передаються до мікроконтролера, де відбувається їх інтерпретація відповідно до закладеного алгоритму роботи системи.

Важливою складовою структури є підсистема виконавчих пристроїв, яка включає світлодіодний модуль, сервопривід, реле, індикаторний світлодіод та п'єзодинамік. Світлодіодний модуль використовується для формування освітлення з можливістю зміни яскравості та кольору. Сервопривід забезпечує керування механічними елементами, наприклад, відкриванням або закриванням ролетів. Реле призначене для комутації зовнішніх електричних пристроїв, що дозволяє розширити функціональні можливості системи та застосовувати її для керування побутовими навантаженнями. Індикаторний світлодіод виконує функцію візуальної сигналізації стану системи, тоді як п'єзодинамік забезпечує звукове сповіщення.

Окремим функціональним блоком є підсистема відображення інформації, реалізована на базі вбудованого дисплея мікроконтролерного модуля. Вона

					<i>КС КРБ 123.196.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ткачук С.А.</i>			<i>Проектна частина</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
<i>Перевірив</i>		<i>Паламар А.М.</i>					<i>20</i>	<i>18</i>
<i>Рецензент</i>		<i>Деркач М.В.</i>				<i>ТНТУ, каф. КС, гр. СІ-42</i>		
<i>Н. Контр.</i>		<i>Тиш Е.В.</i>						
<i>Зав. каф.</i>		<i>Осунівська Г.М.</i>						

забезпечує виведення службових повідомлень, інформації про стан підключення до мережі, розпізнані жести, а також поточні значення параметрів системи, що підвищує зручність взаємодії користувача з пристроєм.

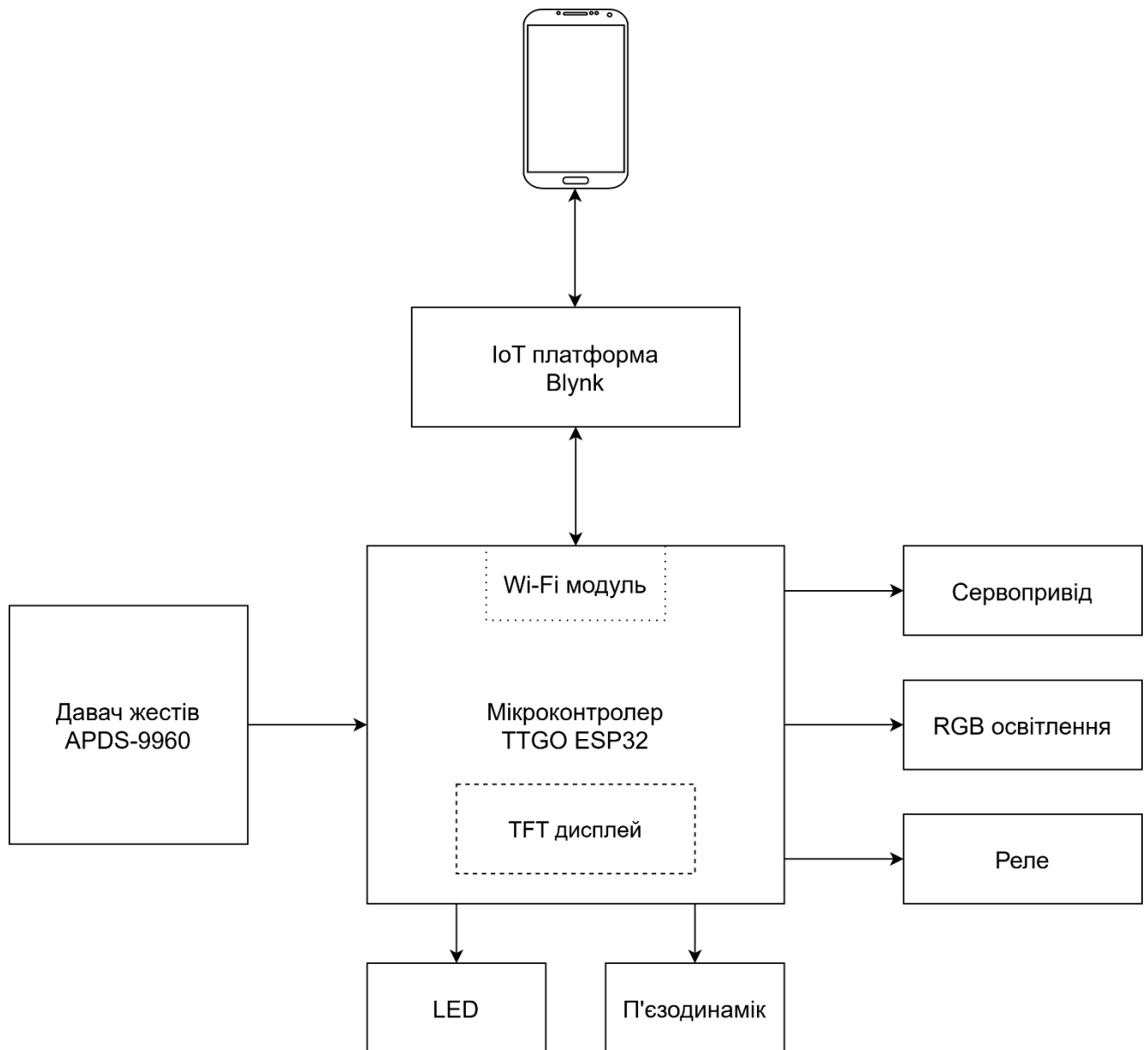


Рисунок 2.1 – Структурна схема системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів

Комунікаційна підсистема представлена бездротовим інтерфейсом Wi-Fi, який забезпечує обмін даними між мікроконтролером і хмарною IoT платформою. Через цей інтерфейс здійснюється передача телеметричної інформації та приймання керуючих команд від користувача. У якості хмарного середовища

використовується Blynk, що надає можливість реалізувати віддалений моніторинг і керування системою за допомогою мобільного додатку.

Взаємодія між основними блоками системи організована таким чином. Користувач формує жест, який реєструється давачем і передається до мікроконтролера. Після обробки сигналу формується керуюча дія, що спрямовується до відповідного виконавчого пристрою. Паралельно інформація про подію та стан системи може передаватися до хмарної платформи та відображатися на дисплеї. У разі отримання команд з IoT платформи вони також обробляються мікроконтролером і впливають на стан виконавчих елементів.

Запропонована структура системи забезпечує інтеграцію сенсорів, обчислювального ядра, виконавчих механізмів та мережевих сервісів, що дозволяє реалізувати ефективно безконтактне керування пристроями як у локальному, так і у віддаленому режимах.

## 2.2 Розроблення апаратного забезпечення безконтактного керування IoT-пристроями

### 2.2.1 Мікроконтролерний модуль TTGO ESP32

Мікроконтролерний модуль TTGO ESP32 є сучасною платформою для розробки вбудованих систем, що поєднує обчислювальні можливості мікроконтролера сімейства ESP32 з інтегрованими засобами відображення інформації та бездротового зв'язку. Основою модуля є мікросхема ESP32, яка містить двоядерний 32-бітний процесор архітектури Xtensa LX6, що забезпечує достатню продуктивність для реалізації алгоритмів обробки даних у реальному часі (рис. 2.2).

Будова модуля включає кілька основних функціональних блоків. Центральним елементом є мікроконтролер ESP32, який містить процесорні ядра, оперативну пам'ять, периферійні інтерфейси та вбудовані модулі бездротового зв'язку. До мікроконтролера підключена зовнішня флеш-пам'ять обсягом до 16 МБ, яка використовується для зберігання програмного коду та даних.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 2.2 – Мікроконтролерний модуль TTGO ESP32

Особливістю модуля є наявність вбудованого кольорового TFT-дисплея діагоналю 1,14 дюйма на базі драйвера ST7789, який підключається до мікроконтролера через SPI-інтерфейс. Дисплей забезпечує відображення графічної інформації, тексту та інтерфейсів користувача, що є важливим для реалізації людино-машинної взаємодії (рис. 2.3).

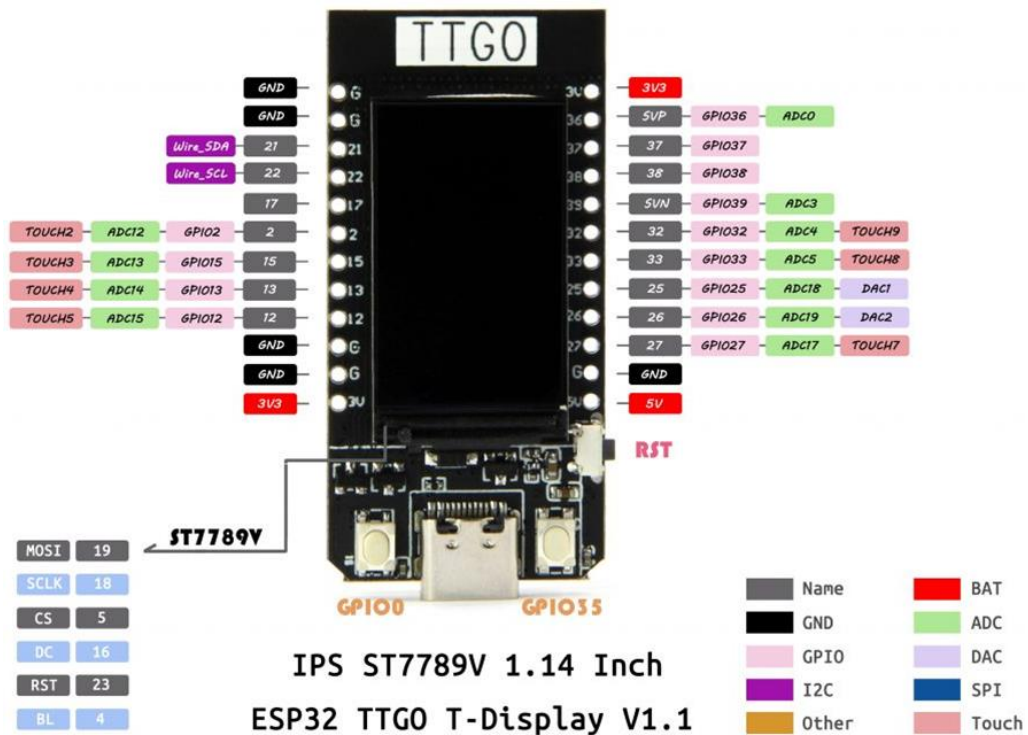


Рисунок 2.3 – Призначення виводів модуля TTGO ESP32

Для забезпечення зв'язку з зовнішніми пристроями модуль має широкий набір інтерфейсів, зокрема UART, SPI, I<sup>2</sup>C, ADC, PWM та інші. Це дозволяє підключати різноманітні сенсори, виконавчі механізми та периферійні модулі. Крім того, вбудовані модулі Wi-Fi і Bluetooth забезпечують можливість бездротової передачі даних і інтеграції з IoT-платформами.

Живлення модуля може здійснюватися як через USB-інтерфейс (5 В), так і від літій-іонного акумулятора напругою 3,7 В. На платі передбачено стабілізацію напруги та контролер заряджання акумулятора, що розширює можливості автономного використання системи. Основні параметри модуля TTGO ESP32 приведені в таблиці 2.1.

Таблиця 2.1 – Параметри модуля TTGO ESP32

Параметр	Значення
Мікроконтролер	ESP32 (Xtensa LX6, 2 ядра)
Тактова частота	до 240 МГц
Оперативна пам'ять	520 КБ SRAM
Флеш-пам'ять	до 16 МБ
Бездротовий зв'язок	Wi-Fi 802.11 b/g/n, Bluetooth 4.2
Дисплей	1.14" TFT IPS (ST7789)
Роздільна здатність дисплея	135 × 240
Інтерфейси	UART, SPI, I <sup>2</sup> C, ADC, PWM, I2S
Живлення	5 В (USB) або 3,7 В (Li-ion)
Робоча напруга	2,7–4,2 В
Споживаний струм	≈60–67 мА
Струм у режимі сну	≈120–350 мкА
Робоча температура	від –40 до +85 °С

Принцип роботи мікроконтролерного модуля полягає у виконанні програмного коду, який реалізує алгоритми обробки вхідних сигналів, прийняття рішень і формування керуючих впливів. Дані з сенсорів надходять через відповідні

інтерфейси до мікроконтролера, де вони обробляються, після чого результати використовуються для керування виконавчими пристроями або передаються через мережу Wi-Fi до IoT-платформи. Паралельно забезпечується виведення інформації на дисплей, що дозволяє реалізувати зворотний зв'язок з користувачем. Завдяки підтримці мережевих протоколів (TCP/IP, HTTP, MQTT) модуль може працювати як клієнт або сервер у складі розподілених IoT-систем.

Вибір модуля TTGO T-Display ESP32 для реалізації комп'ютерної системи безконтактного керування IoT-пристроями є технічно обґрунтованим. Висока обчислювальна потужність двоядерного процесора забезпечує можливість реалізації алгоритмів обробки жестів, керування периферійними пристроями та одночасної роботи з мережевими сервісами без суттєвих затримок. Це є критично важливим для систем реального часу.

Наявність широкого набору інтерфейсів дозволяє легко інтегрувати модуль із сенсорами (зокрема давачем жестів), виконавчими механізмами (сервопривід, світлодіоди) та іншими периферійними пристроями без потреби у додаткових модулях розширення.

Вбудовані засоби бездротового зв'язку забезпечують пряму інтеграцію з IoT-платформами, що дозволяє реалізувати функції віддаленого моніторингу, керування та сповіщення без використання додаткових комунікаційних модулів.

Наявність вбудованого дисплея дозволяє реалізувати зручний інтерфейс користувача без необхідності підключення зовнішніх пристроїв відображення, що спрощує апаратну частину системи та зменшує її габарити.

Крім того, модуль підтримує сучасні програмні платформи, що значно спрощує процес розробки ПЗ та забезпечує наявність великої кількості бібліотек і прикладів.

Мікроконтролерний модуль TTGO ESP32 є оптимальним вибором для реалізації проєктованої комп'ютерної системи, оскільки забезпечує необхідну продуктивність, функціональність, гнучкість і можливість інтеграції з IoT-середовищем при відносно низькій вартості та простоті використання.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

### 2.2.2 Модуль давача жестів APDS-9960

Модуль давача жестів APDS-9960 є інтегрованим оптоелектронним сенсором, призначеним для безконтактного визначення жестів, освітленості, кольору та наближення об'єктів. Він широко застосовується у вбудованих системах і пристроях людино-машинної взаємодії завдяки компактності, низькому енергоспоживанню та можливості роботи без складних обчислювальних алгоритмів, характерних для систем комп'ютерного зору (рис. 2.4).



Рисунок 2.4 – Модуль давача жестів APDS-9960

Конструктивно модуль складається з мікросхеми APDS-9960, інтегрованих інфрачервоних світлодіодів, фотодіодів, АЦП та цифрового інтерфейсу зв'язку. В середині сенсора реалізовано чотири фотодіоди, розташовані у різних напрямках (вгору, вниз, ліворуч, праворуч), що дозволяє визначати напрямок руху об'єкта. Інфрачервоні світлодіоди випромінюють сигнал, який відбивається від руки користувача та реєструється фотодіодами. Мікросхема також містить окремі канали для вимірювання інтенсивності освітлення та кольору (червоний, зелений, синій), що розширює можливості сенсора.

Принцип роботи модуля базується на аналізі змін інтенсивності відбитого інфрачервоного випромінювання. Коли користувач виконує жест у зоні дії сенсора, фотодіоди фіксують різницю сигналів залежно від напрямку руху. Вбудований алгоритм обробки дозволяє інтерпретувати ці зміни як конкретні жести, такі як рух вгору, вниз, ліворуч або праворуч, а також наближення або віддалення об'єкта.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

Отримані дані передаються до мікроконтролера через цифровий інтерфейс I<sup>2</sup>C, що забезпечує просту інтеграцію з більшістю сучасних мікроконтролерних платформ.

Окрім розпізнавання жестів, модуль підтримує функції визначення наближення та вимірювання освітленості і кольору. Це дозволяє використовувати його не лише як інтерфейс введення, а й як багатофункціональний сенсор для аналізу параметрів навколишнього середовища. Вбудовані налаштування чутливості, коефіцієнтів підсилення та порогових значень дозволяють адаптувати роботу сенсора до різних умов експлуатації. В таблиці 2.2 зведені технічні параметри жестів APDS-9960.

Таблиця 2.2 – Параметри датчика жестів APDS-9960

Параметр	Значення
Тип сенсора	Оптичний
Інтерфейс	I <sup>2</sup> C
Робоча напруга	2,4 – 3,6 В
Споживаний струм	до 100 мА (залежно від режиму)
Дальність виявлення жестів	до 10–20 см
Дальність визначення наближення	до 100 мм
Кількість фотодіодів	4 (напрямкові)
Джерело випромінювання	ІЧ світлодіоди
Вбудовані функції	Gesture, Proximity, RGB, ALS
Частота опитування	до декількох сотень Гц
Робоча температура	від –30 до +85 °С

Використання модуля APDS-9960 у розроблюваній комп'ютерній системі безконтактного керування є технічно доцільним та обґрунтованим. Перш за все, цей сенсор забезпечує реалізацію безконтактного інтерфейсу керування без необхідності використання камер і складних алгоритмів обробки зображень, що значно знижує вимоги до обчислювальних ресурсів системи.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

Другим важливим фактором є простота інтеграції з мікроконтролером завдяки використанню стандартного інтерфейсу I<sup>2</sup>C, що дозволяє зменшити складність апаратної частини та скоротити кількість необхідних з'єднань. Крім того, наявність готових бібліотек для роботи з сенсором значно спрощує розробку програмного забезпечення.

Важливою перевагою є багатофункціональність модуля, оскільки він може одночасно виконувати функції розпізнавання жестів, визначення наближення та вимірювання освітленості. Це дозволяє зменшити кількість окремих сенсорів у системі та підвищити її компактність.

Крім того, APDS-9960 характеризується низьким енергоспоживанням, що є важливим для вбудованих систем, а також достатньою точністю та швидкістю для реалізації керування в реальному часі. Діапазон виявлення жестів є оптимальним для використання в умовах, коли користувач знаходиться на близькій відстані від сенсора, що відповідає сценарію застосування системи для людей з обмеженою рухливістю.

### 2.2.3 Модуль з RGB світлодіодами WS2812

Модуль з 8 RGB світлодіодами WS2812 є адресованим світлодіодним пристроєм, призначеним для створення динамічного світлового відображення з можливістю індивідуального керування кожним світлодіодом (рис. 2.5).



Рисунок 2.5 – Модуль з RGB світлодіодами WS2812

					КС КРБ 123.196.00.00 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

Завдяки інтеграції керуючої електроніки безпосередньо в корпус кожного світлодіода, даний модуль широко використовується у вбудованих системах, зокрема в IoT-проектах та системах людино-машинної взаємодії. Конструктивно модуль являє собою друковану плату, на якій розміщено вісім світлодіодів типу WS2812 у корпусі SMD 5050. Кожен світлодіод містить три окремі кристали (червоний, зелений та синій), а також інтегрований драйвер із широтно-імпульсною модуляцією (ШИМ) і схемою цифрового керування. Вбудований контролер забезпечує приймання, збереження та передачу цифрових даних, що дозволяє організувати послідовне з'єднання великої кількості світлодіодів у вигляді ланцюга.

Кожен світлодіод має чотири основні виводи: живлення (VCC), загальний провід (GND), вхід даних (DIN) та вихід даних (DOUT). Сигнал з мікроконтролера подається на перший світлодіод, який зчитує перші 24 біти інформації (по 8 біт на кожен колір), зберігає їх у внутрішньому регістрі, а решту передає на наступний світлодіод. Такий принцип дозволяє керувати всією лінійкою за допомогою лише одного цифрового виходу мікроконтролера.

Принцип роботи модуля базується на послідовній передачі цифрових даних із високою швидкістю (близько 800 кбіт/с). Кожен світлодіод після отримання даних формує сигнал PWM для керування яскравістю кожного з трьох кольорових каналів. Завдяки 8-бітному представленню кожного кольору досягається 256 рівнів яскравості на канал, що забезпечує формування понад 16 мільйонів кольорів.

Модуль з 8 світлодіодами є компактною реалізацією адресованої LED-матриці. Кожен світлодіод може працювати незалежно від інших, що дозволяє реалізовувати різноманітні світлові ефекти, індикацію станів системи, візуалізацію рівнів або режимів роботи. Важливою особливістю є можливість каскадування — при необхідності кілька модулів можуть бути з'єднані в один ланцюг без використання додаткових керуючих ліній. Технічні параметри модуля з RGB світлодіодами WS2812 наведені в таблицю 2.3.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 2.3 – Технічні параметри давача полум'я KY-026

Параметр	Значення
Тип світлодіодів	WS2812 / WS2812B (RGB, SMD 5050)
Кількість світлодіодів	8
Робоча напруга	5 В
Інтерфейс керування	Однопровідний (DIN/DOUT)
Кольорова глибина	24 біти (8 біт на канал)
Кількість кольорів	≈ 16,7 млн
Максимальний струм світлодіода	до 60 мА
Максимальний струм модуля	до 240–480 мА
Кут випромінювання	≈120°
Частота PWM	≈400–2000 Гц
Швидкість передачі даних	≈800 кбіт/с
Робоча температура	–25...+80 °С

Використання модуля з 8 RGB світлодіодами WS2812 у проєктованій комп'ютерній системі є обґрунтованим з огляду на функціональні вимоги та особливості реалізації системи безконтактного керування.

Адресована структура світлодіодів дозволяє керувати всією лінійкою за допомогою лише одного цифрового виводу мікроконтролера, що зменшує кількість необхідних апаратних ресурсів. Це особливо важливо в умовах обмеженої кількості GPIO-виводів мікроконтролерного модуля.

Також перевагою є простота програмної реалізації. Для роботи з WS2812 існує велика кількість готових бібліотек, що дозволяє швидко реалізувати складні світлові ефекти без необхідності розробки низькорівневих алгоритмів керування.

Крім того, модуль забезпечує достатню яскравість та широкий діапазон кольорів, що дозволяє використовувати його як повноцінний освітлювальний елемент для локального підсвічування. У контексті даної системи це дозволяє реалізувати функцію регулювання яскравості освітлення, що керується жестами користувача.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

Важливим фактором є також компактність і низька вартість модуля, що робить його доцільним для використання в навчальних і дослідницьких проєктах. Модуль легко інтегрується з мікроконтролером ESP32 та іншими компонентами системи, що забезпечує гнучкість і масштабованість рішення.

#### 2.2.4 Сервопривід SG90

Сервопривід типу SG90 є компактим електромеханічним пристроєм, призначеним для точного позиціонування вихідного вала в заданому кутовому положенні. Він широко застосовується у вбудованих системах, робототехніці та IoT-проєктах завдяки простоті керування, малим габаритам і достатній точності. У проєктованій комп'ютерній системі сервопривід використовується як виконавчий механізм для керування елементами середовища, зокрема відкривання та закривання рошетів (рис. 2.6).



Рисунок 2.6 – Сервопривід SG90

Конструктивно сервопривід SG90 складається з кількох основних вузлів: мікродвигуна постійного струму, редуктора (системи пластикових зубчастих передач), потенціометра зворотного зв'язку та керуючої електронної плати. Електродвигун забезпечує обертальний рух, який через редуктор передається на вихідний вал із зменшенням швидкості та збільшенням крутного моменту.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

Потенціометр механічно з'єднаний з валом і використовується для вимірювання поточного кута повороту, що дозволяє реалізувати замкнену систему керування.

Принцип роботи сервопривода базується на використанні ШІМ. Мікроконтролер формує керуючий сигнал із періодом близько 20 мс (частота 50 Гц), у якому тривалість імпульсу визначає бажане положення вала. Наприклад, імпульс тривалістю приблизно 1 мс відповідає куту близько 0°, 1,5 мс — середньому положенню (90°), а 2 мс — куту близько 180°.

Вбудована електроніка сервопривода порівнює задане значення (отримане з ШІМ-сигналу) з фактичним положенням, яке визначається потенціометром. У разі розбіжності формується сигнал керування двигуном, який обертає вал у відповідному напрямку до досягнення заданого кута. Коли різниця між заданим і фактичним положенням зникає, двигун зупиняється. Таким чином забезпечується точне позиціонування та утримання положення.

Сервопривід має трипровідне підключення: живлення (VCC), загальний провід (GND) і керуючий сигнал (PWM). Це значно спрощує інтеграцію з мікроконтролерними платформами, зокрема ESP32, оскільки для керування достатньо одного цифрового виходу з підтримкою PWM. Параметри сервоприводу SG90 наведені таблицю 2.4.

Таблиця 2.4 – Параметри сервоприводу SG90

Параметр	Значення
Тип	Мікросервопривід
Робоча напруга	4,8 – 6 В
Керуючий сигнал	PWM (50 Гц)
Діапазон обертання	0° – 180°
Крутний момент	~1,3–1,8 кг·см
Швидкість обертання	~0,10–0,12 с / 60°
Споживаний струм	~100–200 мА (робочий)
Тип редуктора	Пластиковий

Використання сервопривода SG90 у розроблюваній системі є технічно обґрунтованим з огляду на вимоги до функціональності, вартості та складності реалізації. Перш за все, сервопривід забезпечує точне позиціонування вала в заданому кутовому діапазоні, що є необхідним для реалізації механічного керування, наприклад відкривання або закривання ролетів.

Другим важливим фактором є простота керування. Для роботи сервопривода достатньо одного ШІМ-сигналу, який легко генерується мікроконтролером ESP32 без використання складних алгоритмів або додаткових апаратних засобів.

Сервопривід SG90 має невеликі габарити та масу, що дозволяє інтегрувати його в компактні пристрої без значного впливу на конструкцію системи. Крім того, його крутий момент є достатнім для виконання демонстраційних і малонавантажених задач, характерних для даної комп'ютерної системи.

Важливою перевагою є також низька вартість і широка доступність сервопривода, що робить його доцільним вибором для навчальних і дослідницьких проєктів. Наявність великої кількості бібліотек і прикладів використання значно спрощує процес розробки та налагодження.

Сервопривід SG90 повністю відповідає вимогам проєктованої системи, забезпечуючи необхідну функціональність, простоту інтеграції та економічну доцільність, що робить його оптимальним вибором для реалізації виконавчого механізму в даному проєкті.

### 2.2.5 Модуль реле

Модуль реле є електромеханічним пристроєм, призначеним для керування електричними колами з підвищеною напругою або струмом за допомогою низьковольтного сигналу мікроконтролера. У комп'ютерних системах, зокрема IoT, релейні модулі широко застосовуються для комутації навантажень, таких як освітлення, електроприлади, двигуни та інші виконавчі пристрої. Основною перевагою реле є гальванічна розв'язка між керуючою та силовою частинами, що забезпечує захист мікроконтролера від впливу високих напруг (рис. 2.7).

					КС КРБ 123.196.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		33

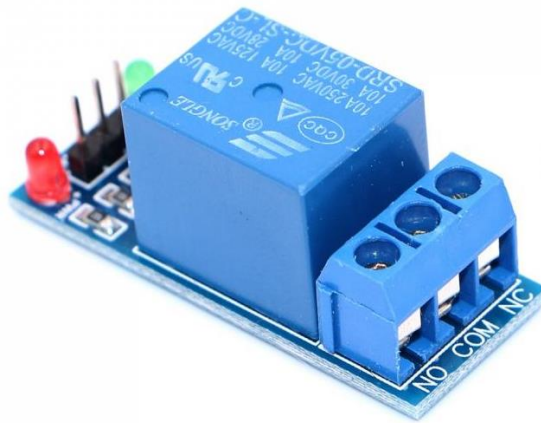


Рисунок 2.7 – Модуль реле

Конструктивно модуль реле складається з двох основних частин: власне електромеханічного реле та керуючої електронної схеми. До складу реле входять котушка (електромагніт), рухомий якір та контактна група, яка включає нормально розімкнені (NO), нормально замкнені (NC) та спільний (COM) контакти. Керуюча частина модуля містить транзисторний ключ, захисний діод, резистори, світлодіод індикації та, у деяких випадках, оптопару для електричної ізоляції.

Принцип роботи модуля реле базується на явищі електромагнітної індукції. При подачі керуючого сигналу з мікроконтролера на вхід модуля активується транзистор, який подає струм на котушку реле. У результаті в котушці створюється магнітне поле, що притягує рухомий якір і змінює стан контактів: нормально розімкнені контакти замикаються, а нормально замкнені — розмикаються. При відключенні сигналу магнітне поле зникає, і контакти повертаються у вихідний стан під дією пружини.

Модуль реле має дві групи виводів: керуючу (низьковольтну) та силову (високовольтну). До керуючої частини належать контакти VCC, GND та вхідний сигнал (IN), які підключаються до мікроконтролера. Силова частина включає контакти COM, NO та NC, до яких підключається навантаження. Така структура дозволяє безпечно керувати високовольтними або високострумними пристроями, використовуючи слабкі сигнали цифрових виходів.

Використання модуля реле у проєктованій комп'ютерній системі є доцільним з огляду на необхідність керування зовнішніми електричними

					КС КРБ 123.196.00.00 ПЗ	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		

пристроями, які можуть працювати з вищими напругами або струмами, ніж допускає мікроконтролер. Основною перевагою є можливість керування такими пристроями за допомогою низьковольтного сигналу, що відповідає рівням логіки ESP32.

Другим важливим фактором є забезпечення електричної ізоляції між мікроконтролером і силовим колом. Завдяки цьому зменшується ризик пошкодження мікроконтролера внаслідок перенапруг, коротких замикань або імпульсних завад, що особливо важливо при роботі з реальними електроприладами.

Модуль реле є простим у використанні та не потребує складних схем керування, оскільки всі необхідні елементи (транзистор, діод, індикація) вже інтегровані на платі. Це значно спрощує розробку апаратної частини системи та підвищує її надійність.

У контексті даної системи модуль реле може використовуватися як додатковий виконавчий елемент для керування зовнішніми пристроями, наприклад, освітленням або іншими побутовими навантаженнями. Це розширює функціональні можливості системи та дозволяє інтегрувати її в реальні умови експлуатації.

Модуль реле є універсальним, надійним і економічно доцільним компонентом, який забезпечує безпечне керування електричними навантаженнями та добре інтегрується з мікроконтролерними системами, що робить його обґрунтованим вибором для реалізації даного проєкту.

### 2.3 Розроблення електричної принципової схеми

На рисунку 2.3 зображено електричну принципову схему системи, яка включає мікроконтролерний модуль, давач жестів, виконавчі пристрої, елементи індикації та модуль живлення. Усі з'єднання виконані з урахуванням електричних параметрів компонентів і рівнів логічних сигналів.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

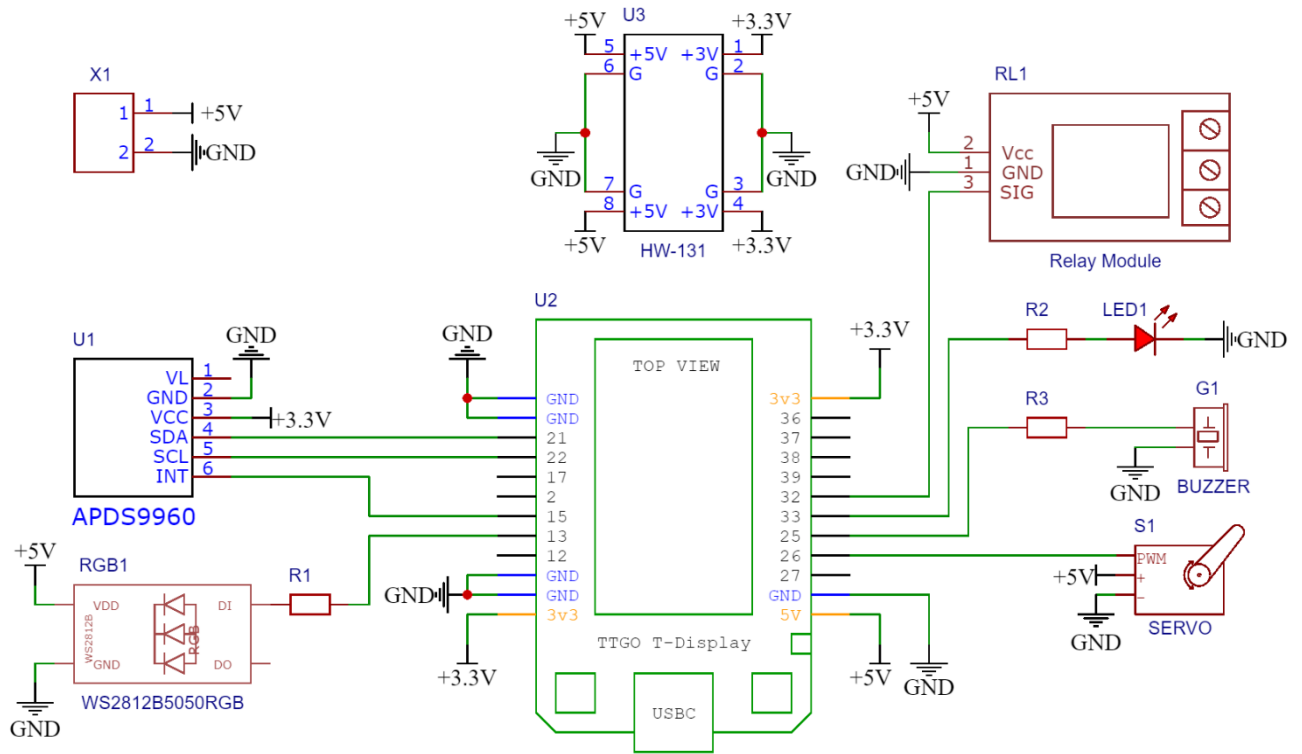


Рисунок 2.8 – Електрична схема пристрою для безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів

Центральним елементом схеми є мікроконтролерний модуль TTGO ESP32, який забезпечує обробку сигналів та керування всіма периферійними пристроями. Живлення системи здійснюється від модуля живлення для макетної плати, який формує стабілізовані напруги 5 В та 3,3 В. Напруга 5 В використовується для живлення сервопривода, світлодіодного модуля WS2812 та реле, тоді як напруга 3,3 В застосовується для живлення датчика жестів та логічних входів мікроконтролера.

Модуль датчика жестів APDS-9960 підключається до мікроконтролера за інтерфейсом I2C. Лінія SDA підключається до GPIO21, а лінія SCL — до GPIO22. Додатково використовується вивід переривання INT, який підключається до GPIO15 мікроконтролера. Це дозволяє реалізувати подієву модель обробки жестів без постійного опитування сенсора.

Сигнальний провід сервоприводу SG90 підключається до GPIO26 мікроконтролера. Керування положенням сервопривода здійснюється за допомогою ШІМ.

					КС КРБ 123.196.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

Модуль з 8 RGB світлодіодами WS2812 LED strip підключається через один сигнальний провід до GPIO13. Для підвищення надійності передачі даних у сигнальну лінію включений послідовний резистор номіналом 330 Ом.

Релейний модуль підключається до мікроконтролера через керуючий вхід, який з'єднується з GPIO32. Реле використовується для комутації зовнішнього навантаження, що дозволяє керувати електричними приладами. У випадку використання модуля з оптронною розв'язкою забезпечується додатковий захист мікроконтролера від перешкод.

Індикаторний світлодіод підключається до GPIO33 через обмежувальний резистор номіналом 220 Ом. Даний елемент використовується для індикації стану системи або виконання певних дій.

П'єзодинамік підключається до GPIO25, який забезпечує формування звукового сигналу за допомогою генерації імпульсів відповідної частоти. Для обмеження струму та захисту може використовуватися послідовний резистор.

					<i>КС КРБ 123.196.00.00 ПЗ</i>	<i>Арк.</i>
						<i>37</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

## РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

### 3.1 Алгоритм роботи системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів

Алгоритм роботи розробленої комп'ютерної системи визначає послідовність дій, які виконуються мікроконтролерним модулем для забезпечення безконтактного керування виконавчими пристроями, обробки жестів, відображення інформації та взаємодії з хмарною IoT платформою. Алгоритм побудований за циклічним принципом із використанням подієвого підходу до обробки сигналів від давача жестів.

На початковому етапі після подачі живлення відбувається ініціалізація апаратних та програмних ресурсів системи. Налаштовуються інтерфейси введення-виведення мікроконтролера, ініціалізується дисплей, встановлюються початкові значення змінних, а також активуються периферійні модулі, зокрема давач жестів APDS-9960, світлодіодний модуль WS2812 LED та сервопривід SG90. Після цього на дисплей виводиться повідомлення про запуск системи.

Далі здійснюється спроба підключення до бездротової мережі Wi-Fi. У разі успішного підключення активується взаємодія з хмарною IoT платформою Blynk, що дозволяє передавати телеметричні дані та приймати керуючі команди. Якщо підключення не встановлено протягом визначеного часу, система переходить у автономний режим роботи, при цьому її основний функціонал зберігається. Інформація про стан підключення відображається на дисплеї.

Після завершення ініціалізації система переходить до виконання основного циклу роботи (рис. 3.1). У кожній ітерації циклу, за наявності підключення до мережі, виконується обробка подій IoT платформи, що забезпечує можливість дистанційного керування параметрами системи, такими як яскравість освітлення, положення сервопривода, режим роботи та стан реле.

					<i>КС КРБ 123.196.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ткачук С.А.</i>			<i>Практична частина</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Паламар А.М.</i>					<i>38</i>	<i>18</i>
<i>Рецензент</i>		<i>Деркач М.В.</i>				<i>ТНТУ, каф. КС, гр. СІ-42</i>		
<i>Н. Контр.</i>		<i>Тиш Е.В.</i>						
<i>Зав. каф.</i>		<i>Осуківська Г.М.</i>						

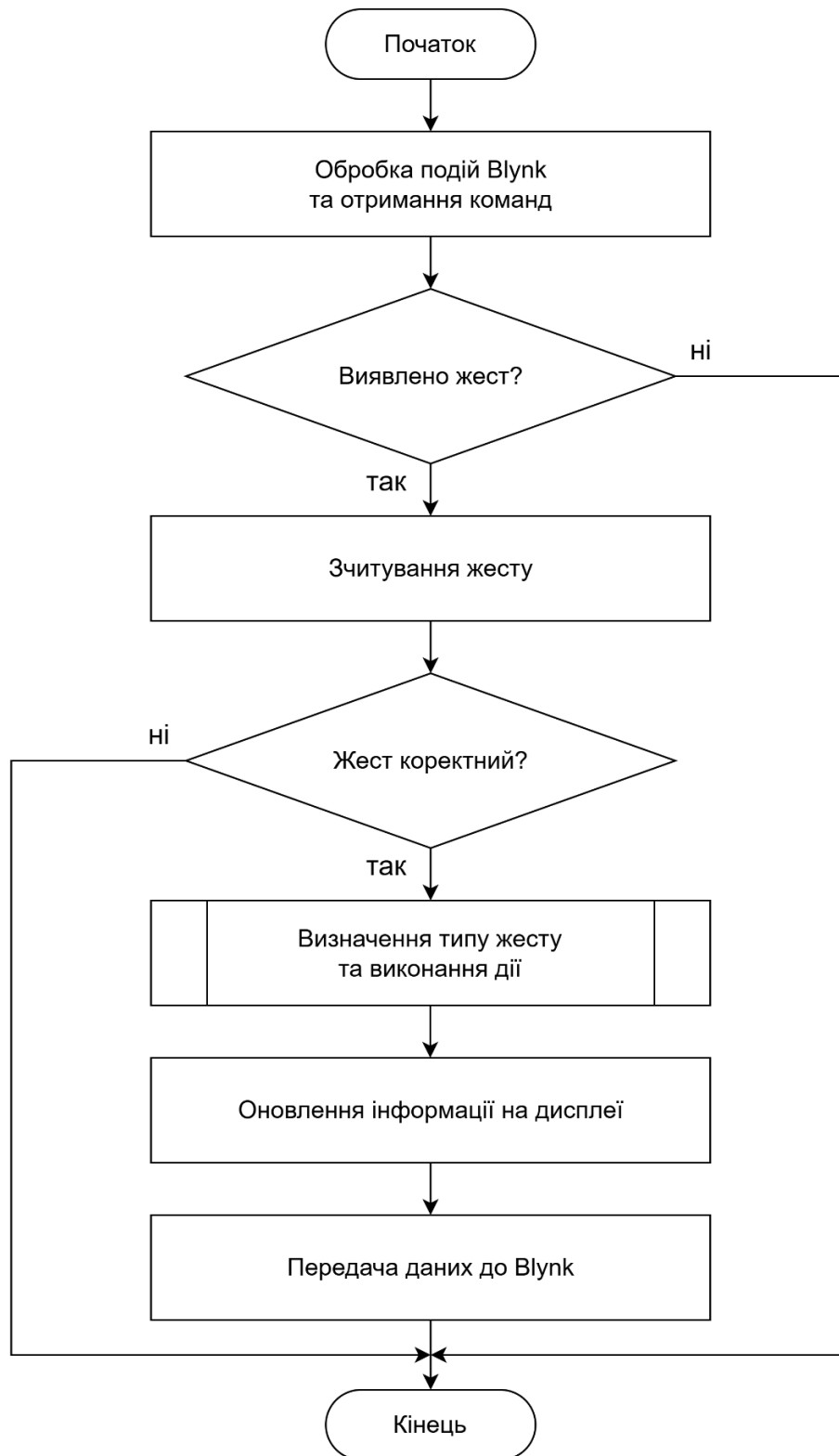


Рисунок 3.1 – Блок-схема алгоритму роботи основного циклу програми у системі безконтактного керування IoT-пристроями

Обробка жестів реалізується за принципом виявлення подій із використанням сигналу переривання від давача жестів. При виникненні відповідного сигналу встановлюється ознака наявності нового жесту, після чого у

циклі перевіряється доступність даних для зчитування. Для підвищення надійності роботи алгоритму застосовано механізм фільтрації жестів, який виключає повторну обробку однакових жестів протягом заданого інтервалу часу, що дозволяє усунути ефект випадкових спрацювань.

Після зчитування жесту виконується його класифікація та відповідна обробка залежно від поточного режиму роботи системи. У режимі керування освітленням реалізується зміна яскравості світлодіодного модуля: жест «вгору» збільшує яскравість, а жест «вниз» — зменшує її у межах допустимого діапазону. У режимі керування механізмом жести «вліво» та «вправо» змінюють кут повороту сервопривода з обмеженням граничних значень.

Окремий режим передбачено для керування релейним модулем. У цьому режимі відповідний жест використовується для перемикання стану реле, що дозволяє вмикати або вимикати підключений електричний пристрій. Перемикання між режимами роботи системи здійснюється за допомогою спеціального жесту наближення, який циклічно змінює активний режим.

Жест віддалення використовується для активації аварійного режиму роботи. У цьому випадку формується звуковий сигнал за допомогою п'єзодинаміка, активується індикаторний світлодіод, а також, за наявності підключення до мережі, надсилається повідомлення через IoT платформу. Це забезпечує можливість виклику допомоги користувачем без фізичного контакту з пристроями.

Після виконання відповідної дії здійснюється оновлення інформації на дисплеї, де відображаються розпізнаний жест, поточний режим роботи, значення яскравості, положення сервоприводу та стан реле. Це забезпечує наочний зворотний зв'язок із користувачем. Одночасно, у разі активного мережевого з'єднання, відповідні параметри передаються до хмарної платформи для віддаленого моніторингу.

Розроблений алгоритм забезпечує комплексну обробку жестів, керування виконавчими пристроями, інтерактивну взаємодію з користувачем та підтримку як локального, так і віддаленого режимів роботи системи.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

## 3.2 Розробка програмного забезпечення

### 3.2.1 Ініціалізація апаратних та програмних ресурсів системи

Підпрограма `setup()` виконує початкову ініціалізацію всіх апаратних та програмних компонентів системи, забезпечуючи її готовність до подальшого функціонування. Даний етап є критично важливим, оскільки саме тут визначаються режими роботи периферійних пристроїв, встановлюються початкові параметри та виконується перевірка працездатності основних модулів (рис. 3.2).

```
void setup() {  
  Serial.begin(115200);  
  tft.init();  
  tft.setRotation(1);  
  showMessage("System Init...");  
  pinMode(STATUS_LED, OUTPUT);  
  pinMode(BUZZER_PIN, OUTPUT);  
  pinMode(RELAY_PIN, OUTPUT);  
  digitalWrite(RELAY_PIN, LOW);  
  // INT  
  pinMode(APDS_INT, INPUT_PULLUP);  
  attachInterrupt(digitalPinToInterrupt(APDS_INT), apdsISR, FALLING);  
  // Servo  
  servo.attach(SERVO_PIN);  
  servo.write(servoPos);  
  // LEDs  
  strip.begin();  
  strip.setBrightness(brightness);  
  strip.show();  
  // I2C  
  Wire.begin(21, 22);  
  // APDS  
  if (!apds.begin()) {  
    showMessage("APDS ERROR");  
    while (1);  
  }  
  apds.enableGesture(true);  
  connectWiFi();  
  showMessage("READY");  
  delay(1000);  
}
```

Рисунок 3.2 – Лістинг коду підпрограми `setup()`

					КС КРБ 123.196.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		41

На початку підпрограми виконується ініціалізація TFT-дисплея за допомогою викликів `tft.init()` та `tft.setRotation(1)`, що забезпечує правильну орієнтацію зображення. Після цього на дисплей виводиться початкове повідомлення про запуск системи, що реалізує зворотний зв'язок із користувачем.

Наступним етапом є налаштування цифрових виводів мікроконтролера. Виводи, до яких підключено індикаторний світлодіод, п'єзодинамік та релейний модуль, конфігуруються як виходи. Початковий стан реле встановлюється у вимкнений (логічний нуль), що забезпечує безпечний старт системи без випадкового вмикання навантаження.

Особливу увагу приділено налаштуванню лінії переривання від давача жестів. Вивід `APDS_INT` конфігурується як вхід із підтягуванням до логічної одиниці, після чого за допомогою функції `attachInterrupt` підключається обробник переривання, який спрацьовує за спадним фронтом сигналу. Такий підхід дозволяє реалізувати модель обробки жестів за подіями, що значно підвищує ефективність роботи системи порівняно з постійним опитуванням сенсора.

Далі виконується ініціалізація сервопривода шляхом прив'язки його до відповідного GPIO-виводу. Встановлюється початкове положення вала, що забезпечує передбачуваний стан механічної частини системи після запуску. Паралельно здійснюється ініціалізація світлодіодного модуля: викликається метод `begin()`, встановлюється початковий рівень яскравості та оновлюється стан світлодіодів.

Після цього ініціалізується інтерфейс I2C за допомогою функції `Wire.begin(21, 22)`, де визначаються виводи для ліній даних і тактового сигналу. Це забезпечує коректний обмін даними з давачем жестів. Ініціалізація самого сенсора виконується через виклик `apds.begin()`. У разі виникнення помилки система виводить відповідне повідомлення на дисплей та переходить у нескінченний цикл, що унеможливує подальшу роботу з некоректно ініціалізованим обладнанням. У випадку успішної ініціалізації активується режим розпізнавання жестів.

Завершальним етапом є підключення до бездротової мережі за допомогою окремої функції `connectWiFi()`. Це дозволяє організувати взаємодію з IoT

					КС КРБ 123.196.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

платформою та забезпечити передачу даних. Після встановлення з'єднання на дисплей виводиться повідомлення про готовність системи до роботи, що сигналізує про успішне завершення етапу ініціалізації.

### 3.2.2 Основний цикл роботи системи та обробка подій

Підпрограма `loop()` реалізує основний цикл функціонування системи, який безперервно виконується після завершення етапу ініціалізації. У межах цього циклу забезпечується обробка подій від хмарної IoT платформи, прийом та інтерпретація жестів користувача, а також виклик відповідних функцій керування виконавчими пристроями. Алгоритм побудовано за принципом подієво-орієнтованої обробки, що дозволяє оптимізувати використання обчислювальних ресурсів мікроконтролера (рис. 3.3).

```
void loop() {  
  if (isOnline) {  
    Blynk.run();  
  }  
  if (gestureFlag) {  
    gestureFlag = false;  
    while (apds.gestureAvailable()) {  
      int gesture = apds.readGesture();  
      processGesture(gesture);  
    }  
  }  
  delay(10);  
}
```

Рисунок 3.3 – Лістинг коду підпрограми `loop()`

На початку кожної ітерації циклу перевіряється стан підключення до мережі за допомогою змінної `isOnline`. У разі наявності активного з'єднання викликається функція `Blynk.run()`, яка забезпечує обробку вхідних та вихідних даних IoT платформи. Це дозволяє приймати команди з мобільного додатку, а також передавати актуальні параметри системи для віддаленого моніторингу. У випадку відсутності підключення дана частина алгоритму пропускається, що забезпечує коректну роботу системи в автономному режимі.

					КС КРБ 123.196.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

Наступним етапом є перевірка наявності нового жесту. Для цього використовується змінна-прапорець `gestureFlag`, яка встановлюється у обробнику апаратного переривання при надходженні сигналу від давача жестів. Такий підхід дозволяє уникнути постійного опитування сенсора та зменшити навантаження на процесор. Якщо прапорець встановлений, система переходить до обробки жесту, після чого значення прапорця скидається.

Обробка жестів реалізується у циклі `while`, який виконується доти, доки у буфері давача доступні нові дані. У кожній ітерації цього циклу виконується зчитування жесту за допомогою функції `readGesture()`, після чого отримане значення передається до окремої функції `processGesture()`. Така структуризація програмного коду дозволяє відокремити логіку зчитування даних від логіки їх обробки, що підвищує читабельність та спрощує подальшу модифікацію програми.

Функція обробки жестів виконує аналіз типу розпізнаного руху та формує відповідні керуючі дії. Залежно від жесту здійснюється регулювання яскравості світлодіодного модуля, зміна положення сервопривода, перемикання стану реле або активація аварійного режиму. Таким чином, основний цикл лише координує процес обробки, тоді як детальна логіка реакції винесена в окремий програмний модуль.

Завершальним етапом кожної ітерації є виконання короткої затримки за допомогою функції `delay(10)`. Це дозволяє стабілізувати роботу системи, зменшити навантаження на процесор та уникнути надмірно частого виконання циклу, що може призвести до некоректної обробки сигналів.

### 3.2.3 Реалізація підсистеми відображення інформації на дисплеї

Підсистема відображення відіграє важливу роль у забезпеченні зворотного зв'язку з користувачем, оскільки дозволяє у реальному часі контролювати стан системи, показувати результати розпізнавання жестів та значення керованих параметрів.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

Функція `showMessage()` призначена для виведення коротких службових повідомлень, які використовуються на етапах ініціалізації та зміни стану системи. На початку виконання функції здійснюється очищення екрана шляхом заповнення його чорним кольором, що дозволяє уникнути накладання тексту та забезпечує чітке відображення нової інформації. Далі встановлюється розмір шрифту, після чого задається позиція курсора на екрані. Виведення тексту здійснюється за допомогою функції `println()`, яка автоматично переносить рядок після завершення виведення. Такий підхід забезпечує просту та зрозумілу індикацію поточного стану, наприклад під час запуску системи або у випадку виникнення помилки (рис. 3.4).

```
void showMessage(String msg) {  
    tft.fillScreen(TFT_BLACK);  
    tft.setTextSize(2);  
    tft.setCursor(10, 50);  
    tft.println(msg);  
}
```

Рисунок 3.4 – Лістинг коду підпрограми `showMessage()`

Функція `drawUI()` реалізує більш розширений інтерфейс користувача, який відображає основні параметри роботи системи (рис. 3.5).

```
void drawUI(String gesture) {  
    tft.fillScreen(TFT_BLACK);  
    tft.setTextSize(2);  
    tft.setCursor(0, 0);  
    tft.println("Gesture Control");  
    tft.setTextSize(1);  
    tft.setCursor(0, 30);  
    tft.printf("Mode: %d\n", mode);  
    tft.printf("Gesture: %s\n", gesture.c_str());  
    tft.printf("Bright: %d\n", brightness);  
    tft.printf("Servo: %d\n", servoPos);  
    tft.printf("Relay: %s\n", relayState ? "ON" : "OFF");  
    tft.printf("WiFi: %s\n", isOnline ? "OK" : "OFF");  
}
```

Рисунок 3.5 – Лістинг коду підпрограми `drawUI()`

					КС КРБ 123.196.00.00 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

Аналогічно до попередньої функції, на початку відбувається очищення екрана, що гарантує коректне оновлення інформації. Далі формується заголовок інтерфейсу, який інформує користувача про призначення системи. Після цього встановлюється менший розмір шрифту для компактного відображення технічної інформації.

У межах цієї функції виводяться ключові параметри системи: поточний режим роботи, розпізнаний жест, значення яскравості світлодіодного модуля, положення сервопривода та стан реле. Для виведення форматованих даних використовується функція `printf()`, що дозволяє зручно комбінувати текстову інформацію зі значеннями змінних. Стан реле відображається у текстовому вигляді («ON» або «OFF»), що підвищує зручність сприйняття інформації користувачем.

Окремо відображається стан підключення до бездротової мережі, що є важливим для контролю роботи системи в контексті інтеграції з IoT платформою. У разі відсутності з'єднання користувач одразу отримує відповідну індикацію, що дозволяє оперативно оцінити можливість віддаленого керування.

### 3.2.4 Реалізація обробки команд від IoT платформи Blynk

Обробка вхідних даних організована за допомогою спеціальних обробників подій типу `BLYNK_WRITE`, кожен з яких прив'язаний до відповідного віртуального каналу. Такий підхід дозволяє чітко розділити функціональність керування окремими підсистемами та спрощує масштабування ПЗ (рис. 3.6).

Перший обробник відповідає за керування яскравістю світлодіодного модуля. При отриманні нового значення воно зчитується та зберігається у змінній, що визначає поточний рівень яскравості. Після цього виконується оновлення параметрів світлодіодної стрічки та застосування змін. Таким чином забезпечується миттєва реакція системи на дії користувача в мобільному додатку.

Другий обробник реалізує керування сервоприводом. Отримане значення інтерпретується як кут повороту, після чого передається у відповідну функцію керування сервоприводом. Це дозволяє дистанційно змінювати положення виконавчого механізму, що є важливим для практичного застосування системи.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

```

BLYNK_WRITE(V0) {
  brightness = param.asInt();
  strip.setBrightness(brightness);
  strip.show();
}

BLYNK_WRITE(V1) {
  servoPos = param.asInt();
  servo.write(servoPos);
}

BLYNK_WRITE(V2) {
  mode = param.asInt();
}

BLYNK_WRITE(V4) {
  relayState = param.asInt();
  digitalWrite(RELAY_PIN, relayState);
}

```

Рисунок 3.6 – Лістинг коду для обробки керуючих команд, що надходять із мобільного додатку

Третій обробник відповідає за зміну режиму роботи системи. Значення, отримане з додатку, визначає активний режим, що впливає на інтерпретацію жестів та логіку керування виконавчими пристроями. Завдяки цьому користувач може перемикає режими роботи незалежно від використання жестів.

Останній обробник реалізує керування релейним модулем. Отримане значення визначає стан реле, після чого відповідний сигнал подається на керуючий вивід мікроконтролера. Це дозволяє здійснювати дистанційне вмикання або вимикання підключеного електричного пристрою.

### 3.2.5 Реалізація алгоритму обробки жестів та формування керуючих дій

У наступному фрагменті програмного коду реалізовано ключову функцію системи — обробку жестів користувача та формування відповідних керуючих дій для виконавчих пристроїв. Підпрограма `processGesture()` приймає код розпізнаного жесту як вхідний параметр та виконує його інтерпретацію залежно від поточного режиму роботи системи.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

На початку функції реалізовано механізм фільтрації жестів, який дозволяє уникнути повторної обробки однакових команд, що можуть виникати через особливості роботи сенсора. Для цього використовується перевірка: якщо поточний жест збігається з попереднім, а інтервал часу між ними менший за встановлене порогове значення, функція завершує виконання без обробки. У протилежному випадку оновлюються значення змінних, що зберігають останній жест та час його отримання. Такий підхід підвищує стабільність роботи системи та покращує точність інтерпретації дій користувача (рис. 3.7).

```
void processGesture(int gesture) {
    // фільтрація
    if (gesture == lastGesture && millis() - lastGestureTime < GESTURE_DELAY) return;
    lastGesture = gesture;
    lastGestureTime = millis();
    String name = "";
    switch (gesture) {
        case APDS9960_UP:
            name = "UP";
            if (mode == 0) {
                brightness = min(255, brightness + 20);
                strip.setBrightness(brightness);
                strip.show();
            }
            if (mode == 2) {
                relayState = !relayState;
                digitalWrite(RELAY_PIN, relayState);
            }
            break;
        case APDS9960_DOWN:
            name = "DOWN";
            if (mode == 0) {
                brightness = max(0, brightness - 20);
                strip.setBrightness(brightness);
                strip.show();
            }
            break;
    }
}
```

Рисунок 3.7 – Частина коду підпрограми processGesture() для обробки жестів руху вгору і вниз

Далі виконується визначення типу жесту за допомогою конструкції розгалуження. Для кожного можливого жесту формується текстова назва, яка використовується для відображення на дисплеї, а також виконується відповідна

					КС КРБ 123.196.00.00 ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

дія. У випадку жесту переміщення вгору здійснюється збільшення яскравості світлодіодного модуля, якщо система перебуває у режимі керування освітленням. Обмеження значення максимальною величиною запобігає некоректній роботі світлодіодів. Жест переміщення вниз виконує зменшення яскравості світлодіодів, що дозволяє плавно регулювати рівень освітлення.

Для жестів переміщення вліво та вправо реалізовано керування сервоприводом, яке доступне у відповідному режимі роботи. Зміна положення відбувається з кроком, що забезпечує дискретне, але достатньо точне регулювання, а також враховуються обмеження допустимого діапазону кутів повороту (рис. 3.8).

```
case APDS9960_LEFT:
    name = "LEFT";
    if (mode == 1) {
        servoPos = max(0, servoPos - 20);
        servo.write(servoPos);
    }
    break;
case APDS9960_RIGHT:
    name = "RIGHT";
    if (mode == 1) {
        servoPos = min(180, servoPos + 20);
        servo.write(servoPos);
    }
    break;
```

Рисунок 3.8 – Частина коду підпрограми processGesture() для обробки жестів руху вліво і вправо

Жест наближення використовується для перемикання режимів роботи системи. Кожне спрацювання цього жесту змінює активний режим циклічно, що дозволяє користувачу послідовно переходити між керуванням освітленням, сервоприводом та реле. Такий підхід є інтуїтивно зрозумілим і не потребує додаткових елементів керування (рис. 3.9).

Жест віддалення активує аварійний режим, у межах якого викликається функція сигналізації. Це дозволяє використовувати систему не лише для керування пристроями, але й як засіб виклику допомоги, що є особливо актуальним у контексті застосування для людей з обмеженими можливостями.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    case APDS9960_NEAR:
        name = "MODE";
        mode++;
        if (mode > 2) mode = 0;
        break;
    case APDS9960_FAR:
        name = "ALARM";
        alarmSignal();
        break;
}
drawUI(name);
if (isOnline) {
    Blynk.virtualWrite(V0, brightness);
    Blynk.virtualWrite(V1, servoPos);
    Blynk.virtualWrite(V2, mode);
    Blynk.virtualWrite(V4, relayState);
}
}

```

Рисунок 3.9 – Частина коду підпрограми processGesture() для обробки жестів руху віддалення і наближення

Після обробки жесту виконується оновлення інтерфейсу користувача шляхом виклику функції відображення, якій передається назва розпізнаного жесту. Це забезпечує миттєвий зворотний зв'язок і дозволяє користувачу контролювати виконані дії. Додатково, у разі наявності підключення до мережі, оновлені значення параметрів передаються до IoT платформи через відповідні віртуальні канали. Це гарантує синхронізацію локального стану системи з віддаленим інтерфейсом керування

### 3.3 Налаштування хмарної IoT платформи Blynk

Для забезпечення віддаленого моніторингу та керування розробленою системою було використано хмарну IoT платформу Blynk. Налаштування платформи виконувалося з урахуванням реалізованого програмного забезпечення мікроконтролера, яке передбачає двосторонній обмін даними, роботу в онлайн та офлайн режимах, а також інтеграцію з виконавчими пристроями та підсистемою відображення.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

На першому етапі було створено новий шаблон пристрою (Template) у середовищі Blynk, у якому визначено тип апаратної платформи (ESP32) та спосіб підключення через Wi-Fi. У межах шаблону було налаштовано набір віртуальних змінних (Datastreams), які відповідають параметрам системи та використовуються для обміну даними між мікроконтролером і мобільним додатком. Зокрема, було створено змінні для керування яскравістю освітлення, положенням сервопривода, вибором режиму роботи системи, а також станом реле. Окремо передбачено змінну для формування сигналу тривоги (рис. 3.10).

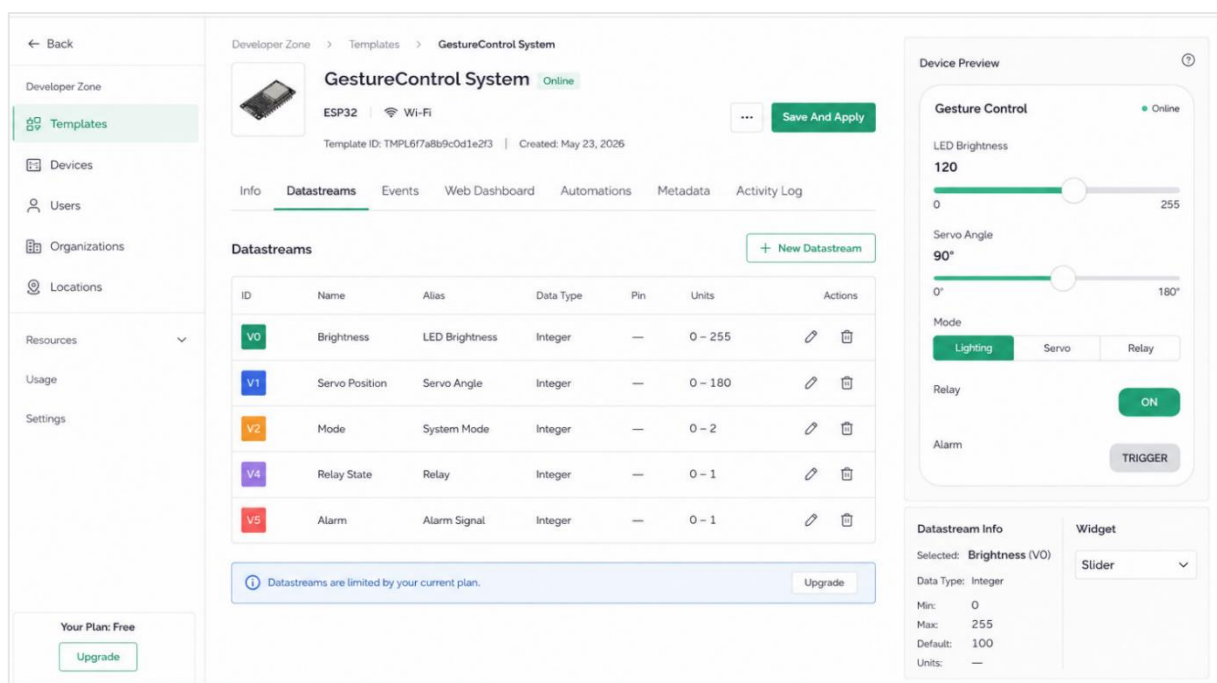


Рисунок 3.10 – Процес налаштування IoT платформи Blynk

Для керування яскравістю світлодіодного модуля було створено віртуальний канал, пов'язаний із повзунком (Slider) у мобільному додатку, що дозволяє задавати значення у відповідному діапазоні. Аналогічно для керування положенням сервопривода було реалізовано окремий повзунок, який задає кут повороту виконавчого механізму.

Керування реле було реалізовано за допомогою кнопкового віджета (Button), який працює у двопозиційному режимі (увімкнено/вимкнено). Це дозволяє дистанційно керувати підключеним електричним пристроєм. Для відображення поточного стану системи додатково використовуються індикаторні елементи, такі

					КС КРБ 123.196.00.00 ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

як Label або Value Display, що відображають значення параметрів, які передаються з мікроконтролера.

Особливістю налаштування є підтримка двостороннього обміну даними. Значення параметрів можуть змінюватися як із мобільного додатку, так і внаслідок розпізнавання жестів. У другому випадку мікроконтролер передає оновлені значення до платформи, що забезпечує синхронізацію станів між фізичним пристроєм і віддаленим інтерфейсом. Таким чином, незалежно від джерела керування користувач отримує актуальну інформацію.

Для реалізації аварійного сповіщення було використано механізм подій (Events) платформи Blynk. Було створено подію, яка активується при виконанні відповідного жесту, після чого користувач отримує push-сповіщення на мобільний пристрій. Це забезпечує можливість оперативного реагування у разі потреби в допомозі.

Під час налаштування також було враховано особливість роботи системи в автономному режимі. У разі відсутності підключення до мережі Wi-Fi мікроконтролер продовжує виконувати всі локальні функції керування, однак обмін даними з платформою тимчасово припиняється. Після відновлення з'єднання синхронізація параметрів відновлюється автоматично.

У результаті виконаних налаштувань було сформовано зручний користувацький інтерфейс, який дозволяє здійснювати повноцінне керування системою, контролювати її стан у реальному часі та отримувати повідомлення про події. Це забезпечує інтеграцію локального безконтактного керування з можливостями віддаленого доступу, що підвищує функціональність і зручність використання розробленої системи.

### 3.4 Результати моделювання та тестування системи

Перевірка працездатності розробленої комп'ютерної системи безконтактного керування IoT-пристроями здійснювалася у два етапи: спочатку було виконано моделювання в середовищі Circuit Designer, після чого проведено

					КС КРБ 123.196.00.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

експериментальне дослідження фізичного прототипу. Такий підхід дозволив поетапно виявити та усунути можливі помилки як на рівні електричної схеми, так і на рівні програмного забезпечення.

На етапі моделювання було створено віртуальну модель системи відповідно до розробленої електричної принципової схеми. До складу моделі було включено мікроконтролерний модуль, датчик жестів, світлодіодний модуль, сервопривід, реле, п'єзодинамік та елементи індикації (рис. 3.11).

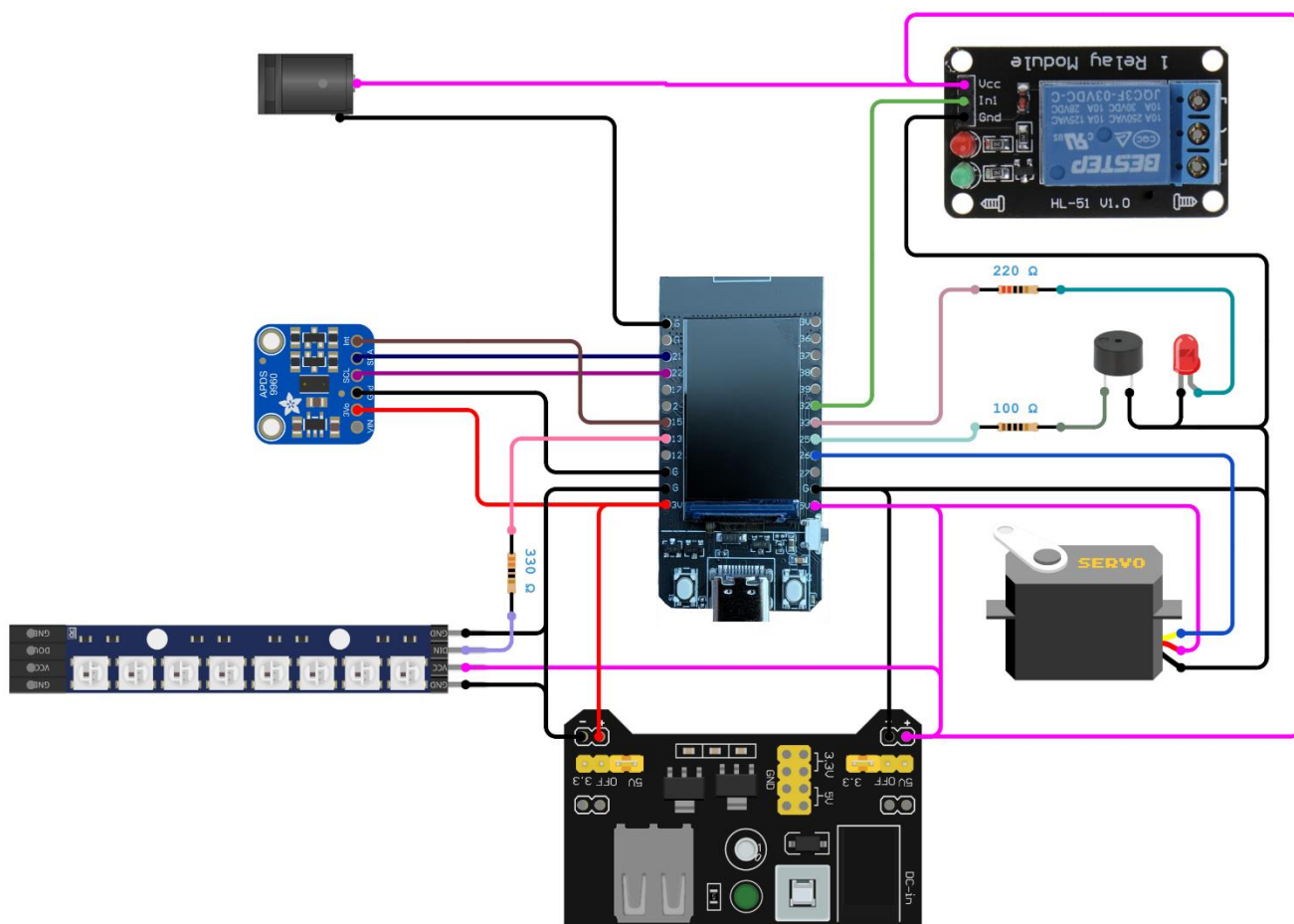


Рисунок 3.11 – Модель системи безконтактного керування IoT-пристроями

У процесі моделювання було перевірено коректність підключення інтерфейсу I2C для обміну даними з датчиком жестів, а також працездатність керуючих сигналів для виконавчих пристроїв. Особлива увага приділялася сигналам керування сервоприводом та світлодіодним модулем, оскільки вони використовують різні типи сигналів. Також було перевірено роботу релейного модуля в умовах комутації навантаження.

Моделювання дозволило виявити потенційні проблеми, пов'язані з живленням компонентів, зокрема необхідність розділення шин 5 В і 3,3 В. Крім того, було підтверджено правильність логіки керування виконавчими пристроями відповідно до умов алгоритму. Важливим результатом цього етапу стало підтвердження коректності структурної та електричної організації системи до її фізичної реалізації.

Після завершення моделювання було здійснено складання фізичного прототипу системи на основі макетної плати. На першому етапі виконано монтаж мікроконтролерного модуля та підключення джерела живлення. Далі було підключено давач жестів.

Наступним етапом було підключення виконавчих пристроїв: світлодіодного модуля, сервопривода та реле. Було перевірено коректність подачі сигналів керування та реакцію пристроїв на зміну цих сигналів. Окремо виконано підключення п'єзодинаміка та індикаторного світлодіода для реалізації функцій сигналізації.

Після завершення апаратного монтажу було виконано завантаження програмного забезпечення до мікроконтролера та проведено поетапне тестування функціональних можливостей системи. На початковому етапі перевірено процес ініціалізації системи, зокрема відображення службових повідомлень на дисплеї та встановлення з'єднання з бездротовою мережею. Було підтверджено коректну роботу як у режимі підключення до мережі, так і в автономному режимі.

Далі проведено тестування підсистеми розпізнавання жестів. Було перевірено коректність визначення жестів та їх інтерпретацію відповідно до заданого алгоритму. Важливим аспектом стало тестування механізму фільтрації жестів, який забезпечив усунення помилкових спрацювань та підвищення стабільності роботи системи.

Під час тестування виконавчих пристроїв було підтверджено правильність реалізації керування освітленням, включаючи зміну яскравості світлодіодів, а також керування положенням сервопривода з обмеженням діапазону кутів. Окремо перевірено функціонування релейного модуля, що забезпечує вмикання та вимикання підключеного навантаження.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

Особлива увага приділялася тестуванню аварійного режиму. Було підтверджено, що відповідний жест активує звуковий сигнал, індикаторний світлодіод та формує повідомлення на хмарну платформу. Це підтверджує можливість використання системи як засобу виклику допомоги.

На завершальному етапі було перевірено інтеграцію з хмарною платформою, включаючи передачу даних та дистанційне керування параметрами системи. Було встановлено, що зміни, виконані з мобільного додатку, коректно впливають на стан виконавчих пристроїв, а дані, сформовані системою, відображаються у віддаленому інтерфейсі.

					<i>КС КРБ 123.196.00.00 ПЗ</i>	<i>Арк.</i>
						55
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

## РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

### 4.1 Ергономічні проблеми безпеки життєдіяльності

Основним завданням ергономіки є забезпечення ефективної взаємодії людини і техніки, щоб перейти від техніки безпеки до безпечної техніки, яку ми використовуємо як у виробничій, так і побутовій сферах. Це один з основних напрямків ергономіки.

Основним об'єктом ергономіки є «людино-машинні» системи. Тому основне завдання ергономіки – вивчення зв'язків між елементами «людиномашинних» систем, розробка методів сумісності основного її компонента – людини з іншими середовищами та машинами, технікою.

Завдання ергономіки:

- вивчення закономірностей взаємодії людини з технічними системами та навколишнім середовищем;
- формування принципів побудови системи «людино-машинних» систем та алгоритмів дії у них людини-оператора;
- розробка основ проектування діяльності людини-оператора з врахуванням специфіки експлуатації технічних систем та факторів навколишнього середовища;
- розробка перспективних форм праці людини і пов'язаних з нею технічних систем, факторів навколишнього середовища;
- розробка методів дослідження, проектування та експлуатації «людиномашинних» систем, які забезпечують безпеку людини, ефективність праці.

Проблемами взаємодії людини та машини займається також інженерна психологія, яка вивчає закономірності процесів інформаційної взаємодії людини у системі «людино-машинних» систем.

					<i>КС КРБ 123.196.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ткачук С.А.</i>			<i>Безпека життєдіяльності, основи охорони праці</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Паламар А.М.</i>					56	6
<i>Консульт.</i>		<i>Сенчишин В.С.</i>				<i>ТНТУ, каф. КС, гр. СІ-42</i>		
<i>Н. Контр.</i>		<i>Тиш Е.В.</i>						
<i>Зав. каф.</i>		<i>Осужівська Г.М.</i>						

Проблемами взаємодії людини та машини займається також інженерна психологія, яка вивчає закономірності процесів інформаційної взаємодії людини у системі «людино-машинних» систем [32].

У системі «людино-машинних» систем завжди є 3 елементи:

- предмет праці;
- засоби праці;
- суб'єкт праці.

Найменшою цільною одиницею, де наявні вказані елементи, є місце праці. Місце праці – це зона, де відбувається трудова діяльність людини. Місце праці обладнане засобами відображення інформації, органами керування та допоміжним обладнанням.

Організацією місця праці називається проведення системи заходів щодо його обладнання засобами та предметами праці і їх розташуванням у визначеному порядку з метою досягнення [32]:

- оптимізації умов трудової діяльності;
- безпеки праці;
- максимальної ефективності;
- комфортності роботи людини.

До робочого місця ставляться такі вимоги:

- достатній робочий простір, який дає змогу працюючій людині здійснювати необхідні рухи та переміщення;
- достатні фізичні, зорові та слухові зв'язки між людиною та обладнанням, а також між людьми під час виконання спільного трудового завдання;
- необхідний рівень освітлення;
- наявність необхідних засобів захисту;
- оптимальне розташування робочих місць, а також безпечні та достатні проходи для працюючих людей.

При організації робочого місця враховують основні антропометричні дані людини. Найважливішою характеристикою робочого місця є зона досягнення

					КС КРБ 123.196.00.00 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

моторного поля. Моторне поле – це простір робочого місця, в якому розміщені органи керування та інші технічні засоби, в якому людина здійснює рухові дії для виконання робочого завдання.

Ергономіка виробила конкретні вимоги до антропометричних показників обладнання [32].

Характеристика пульта:

- загальна висота: «сидячи»– 1650 мм, «стоячи» – не більше ніж 1300 мм;
- висота розміщення органів керування для положення «сидячи» 530 - 1040 мм, стоячи - 1000 - 1500 мм.

Характеристики крісла:

- форма сидіння-квадратна;
- форма спинки - прямокутна вгнута;
- розмір сидіння - 400x400 мм, спинки - 300x120 мм;
- кут нахилу сидіння назад - 50 - 60°;
- кут нахилу спинки - 50 - 100°.

Розміри вільного місця для ніг:

- висота -не менше 600 мм;
- ширина - не менше 500 мм;
- глибина - не менше 400 мм.

Досягнення органів керування по горизонталі – півколо радіусом 600 мм.

Встановлені також відстань між органами керування, їх розміри, зусилля переміщення, величина переміщення, напрямок переміщення.

Для операторів, які працюють з екранами дисплеїв та інших індикаторів, можуть бути рекомендовані такі режими праці та відпочинку. Тривалість безперервної праці не повинна перевищувати 4-6 год. В іншому випадку працездатність через втому зору раптово знижується.

Оператор, який стежить за екраном індикатора, найуважніше працює протягом перших 30 хв чергування. А далі, внаслідок втоми зорового аналізатора, кількість помилок зростає май же в два рази та залишається незмінною до кінця

					КС КРБ 123.196.00.00 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

другої години. Потім спостерігається нове зростання кількості помилок через загальну втому оператора. Тому для підтримки високої ефективності праці може бути рекомендований 30-хвилинний період чергування з наступною 30-хвилинною перервою.

#### 4.2 Соціальне значення охорони праці

Закон України «Про охорону праці» визначає «Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних та лікувально профілактичних заходів і засобів, спрямованих на збереження здоров'я та працездатності людини в процесі праці».

Завдання охорони праці – забезпечення безпечних, нешкідливих і сприятливих умов праці через вирішення багатьох складних завдань [33].

Соціальне значення охорони праці включає в себе різні аспекти, пов'язані з безпекою та благополуччям працівників на робочому місці. Основні аспекти соціального значення охорони праці включають: здоров'я та безпеку працівників, покращення умов праці, зниження виробничого травматизму і профзахворювань, збільшення продуктивності, захист прав працівників, збереження людського капіталу, підвищення задоволеності працівників, зміцнення соціального діалогу, що – забезпечення сталого розвитку суспільства.

Охорона праці спрямована на запобігання травмам, хворобам та іншим негативним наслідкам, пов'язаним з виконанням трудових обов'язків. Це допомагає зберегти фізичне та психологічне здоров'я працівників, покращує їх якість життя та сприяє загальному благополуччю [33].

Охорона праці сприяє створенню безпечних та здорових умов праці. Це може включати належне оснащення робочого місця, забезпечення належної вентиляції, освітлення та ергономіки, а також зменшення шкідливих чинників, таких як шум, вібрація та отруйні речовини. Здорові та безпечні робочі умови сприяють збільшенню продуктивності працівників, тобто, коли працівники мають

					КС КРБ 123.196.00.00 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

можливість працювати в безпечному та комфортному середовищі, вони здатні бути більш ефективними та зосередженими на своїх обов'язках.

Заходи з охорони праці сприяють зменшенню відсутності працівників на робочому місці через травми або хвороби, що в свою чергу забезпечує більшу стабільність та ефективність роботи. Охорона праці сприяє захисту прав працівників, зокрема щодо безпеки, здоров'я, працевлаштування та рівного ставлення на робочому місці. Вона включає в себе нормативно-правові акти, які гарантують права працівників і встановлюють відповідальність роботодавців за забезпечення безпеки й охорони праці. Забезпечення безпеки та здоров'я працівників допомагає уникнути втрати кваліфікованої робочої сили, зменшує витрати на підготовку та навчання нових працівників, а безпечне та здорове робоче середовище сприяє покращенню задоволеності працівників [34].

Охорона праці також є важливою складовою корпоративної відповідальності. Підприємства, які вкладають зусилля у створення безпечних умов праці, демонструють свою готовність нести відповідальність за добробут своїх працівників та сприяють створенню позитивного іміджу. Також охорона праці допомагає в розвитку соціального діалогу між роботодавцями та працівниками. Забезпечення безпеки й охорони праці вимагає співпраці всіх зацікавлених сторін, включаючи роботодавців, профспілки та урядові органи. Це сприяє створенню конструктивного діалогу, вирішенню проблем та покращенню умов праці.

Забезпечення безпеки й охорони праці має значний вплив на економіку та суспільство в цілому, оскільки сприяє зниженню соціального навантаження на суспільство. Заходи з охорони праці сприяють зниженню витрат на медичне обслуговування та страхування. Впровадження ефективних заходів з охорони праці може допомогти зменшити витрати на компенсацію травм та хвороб, пов'язаних з роботою, а також на відшкодування матеріальних збитків, пов'язаних з недбалістю або порушеннями у сфері безпеки. Це сприяє оптимізації виробничих процесів та підвищенню рентабельності [34].

					КС КРБ 123.196.00.00 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

Охорона праці є важливою складовою сталого розвитку. Заходи з безпеки й охорони праці спрямовані на забезпечення екологічної, соціальної та економічної сталості. Також охорона праці може стимулювати інновації та покращення робочих процесів. Коли компанії зосереджуються на створенні безпечного та здорового робочого середовища, вони сприяють розвитку нових технологій, методів роботи та інноваційних рішень, які покращують якість та продуктивність праці.

Загалом, соціальне значення охорони праці полягає в забезпеченні безпеки, здоров'я та благополуччя працівників, підвищенні якості життя та рівня задоволення від праці, сприянні сталому розвитку та підвищенню конкурентоспроможності підприємств та суспільства в цілому.

Працівники, які використовуватимуть систему безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів, повинні знати та дотримуватися вимог охорони праці.

					<i>КС КРБ 123.196.00.00 ПЗ</i>	<i>Арк.</i>
						61
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

## ВИСНОВКИ

У результаті виконання кваліфікаційної роботи було розроблено комп'ютерну систему безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів.

Проаналізовано технічне завдання, визначено основні вимоги до системи та досліджено сфери її застосування. Проведено огляд існуючих рішень у галузі безконтактного керування, що дозволило виявити їх недоліки та обґрунтувати доцільність розробки запропонованої системи.

Розроблено структуру системи, створено електричну принципову схему та обґрунтовано вибір елементної бази. Визначено склад апаратних компонентів, що забезпечують ефективну реалізацію функцій розпізнавання жестів, керування виконавчими пристроями та візуалізації стану системи.

Розроблено алгоритм роботи системи та реалізовано ПЗ для мікроконтролера. Забезпечено інтеграцію з хмарною IoT платформою, що дозволяє здійснювати моніторинг і керування в реальному часі, а також реалізовано функцію сповіщення про критичні події.

Виконано моделювання системи та тестування фізичного прототипу, що підтвердило працездатність розроблених рішень, коректність функціонування всіх компонентів і відповідність системи поставленим вимогам.

					<i>КС КРБ 123.196.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		62

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Міронов Н.О., Самчук Л.М. Дослідження характеристик та методології системи розпізнавання жестів для безконтактної взаємодії з комп'ютером з використанням технологій Computer Vision. Науковий журнал "Комп'ютерно-інтегровані технології: освіта, наука, виробництво", 2024. Вип. № 57. С. 111-119.
2. Małeckı K., Nowosielski A., Kowalicki M. Gesture-based user interface for vehicle on-board system: a questionnaire and research approach. Applied Sciences, 2020. 10(18), 6620.
3. Sideridis V., Zacharaki, A., Tzagkaraki, G., Papadopouli M. Gesturekeeper: Gesture recognition for controlling devices in IoT environments. In 2019 27th European Signal Processing Conference (EUSIPCO). 2019. P. 1-5.
4. Adsul S., Khobragade S., Kohad N., Khot P., Kumbhar S. Real-Time Vision-Based Hand Gesture Recognition System for Multi-Functional Contactless Control of Digital Applications. AIJR Proceedings, 2025. 7(6), P. 124-134.
5. Theodoridou E., Cinque L., Mignosi F., Placidi G., Polsinelli M., Tavares J.M.R., Spezialetti M. Hand tracking and gesture recognition by multiple contactless sensors: A survey. IEEE Transactions on Human-Machine Systems, 2022. 53(1). P. 35-43.
6. Mohyuddin H., Moosavi S.K.R., Zafar M.H., Sanfilippo F. A comprehensive framework for hand gesture recognition using hybrid-metaheuristic algorithms and deep learning models. Array, 2023 (19). P. 100317.
7. 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.
8. Лупенко С.А., Пасічник В.В., Тиш Є.В. Комп'ютерна логіка. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 354 с.
9. Буров Є., Митник М. Комп'ютерні мережі. (у 2-х томах). Львів, Магнолія, 2018. 740 с.
10. Stadnyk M., Palamar A. Project management features in the cybersecurity area. Scientific Journal of TNTU, Ternopil, Ukraine, 2022. Vol. 106, No 2. P. 54–62.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						63
Змн.	Арк.	№ докум.	Підпис	Дата		

11. Yatsyshyn V., Pastukh O., Kukharska V., Palamar A., Kulikov S. Method and tool of detecting software architecture patterns in the process of computer systems development. CEUR Workshop Proceedings, 4th International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP 2024), Ternopil, Ukraine, Opole, Poland, October 23-25, 2024. Vol. 3896. P. 12-24.

12. Palamar M., Yavorska M., Palamar A., Strembitskyi M. Modeling and Research of Satellite Antenna Adjustment Process for Earth Remote Sensing. 2022 IEEE 2nd Ukrainian Microwave Week (UkrMW), Kharkiv, Ukraine, November 14-18, 2022. P. 317-320.

13. Palamar M., Nakonetchnyi Y., Palamar A., Strembitskyi M., Apostol Y. Modernization of the azimuth drive design for the antenna system. Scientific Journal of TNTU, Ternopil, Ukraine, 2025. Vol. 117, No 1, P. 54–61.

14. Strembitskyi M., Yavorska M., Palamar A., Kochan R., Yeromenko V. A comparative study of bug algorithms for robot navigation. CEUR Workshop Proceedings, 3rd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2025), Ternopil, Ukraine, June 11-12, 2025. Vol. 4057 P. 312-321.

15. Palamar A., Voloskyi V., Kramar O., Kramar T., Stankevych O., Yatsyshyn V. Information computer system with a virtual tour for cultural heritage preservation of the Zbarazh Castle Museum's exhibition hall. CEUR Workshop Proceedings, The 3rd International Workshop on Social Communication and Information Activity in Digital Humanities (SCIA 2024), Lviv, Ukraine, October 31, 2024. Vol. 3851.

16. Романов Д.В., Осухівська Г.М., Паламар А.М. Система управління зовнішнім освітленням на основі Інтернету речей. Актуальні задачі сучасних технологій : збірник тез доповідей X міжнародної науково-практичної конференції молодих учених та студентів (Тернопіль, 24-25 листопада 2021 року), Тернопіль: ТНТУ, 2021. С. 120.

17. Оконський М.В., Лупенко С.А., Паламар А.М. Інформаційно-вимірювальна система для контролю метеорологічних параметрів на основі Інтернету речей. Матеріали ІХ науково-технічної конференції "Інформаційні моделі, системи та технології" Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 8–9 грудня 2021 року), Тернопіль: ТНТУ, 2021. С. 118.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						64
Змн.	Арк.	№ докум.	Підпис	Дата		

18. Palamar A., Stadnyk M., Palamar M. Adaptive PID regulation method of uninterruptible power supply battery charge current based on artificial neural network. Scientific Journal of TNTU, Ternopil, Ukraine, 2022. Vol. 107, No 3. P. 5–13.

19. Palamar M., Horyn T., Palamar A., Batuk V. Method of calibration MEMS accelerometer and magnetometer for increasing the accuracy determination angular orientation of satellite antenna reflector. Scientific Journal of TNTU, Ternopil, Ukraine, 2022. Vol. 108, No 4. P. 79–88.

20. Palamar M., Pasternak Y., Palamar A., Poikhalo A. Precision tracking of the trajectory LEO satellite by antenna with induction motors in the control system. Proceedings of the 2017 IEEE 9th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2017), Bucharest, Romania, September 21–23, 2017. Vol. 2. P. 1051–1055.

21. Palamar A., Pettai E. Microgrid for the Department of Electrical Drives and Power Electronics. 8th International Symposium "Topical Problems in the Field of Electrical and Power Engineering" and "Doctoral School of Energy and Geotechnology II" (January 11-16, 2010), Pärnu, Estonia, 2010. P. 54-61.

22. Palamar A. Methods and means of increasing the reliability of computerized modular uninterruptible power supply system. Scientific Journal of TNTU, Ternopil, Ukraine, 2020. Vol. 99, No 3. P. 133–141.

23. Palamar A., Palamar M. Fire Safety Monitoring System Based on Internet of Things. CEUR Workshop Proceedings, 2023. 1st International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2023), Ternopil, Ukraine, June 14-16, 2023. Vol. 3468. P. 164-172.

24. Паламар М., Пастернак Ю., Паламар А. Дослідження динамічних похибок системи прецизійного керування антеною з асинхронним електроприводом. Вісник ТНТУ, Тернопіль: ТНТУ, 2014. Вип. 76, № 4. С. 164–173.

25. Voloshchuk A., Velychko D., Osukhivska H., Palamar A. Computer system for energy distribution in conditions of electricity shortage using artificial intelligence. CEUR Workshop Proceedings, 2nd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2024), Ternopil, Ukraine, June 12-14, 2024. Vol. 3742 P. 66-75.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						65
Змн.	Арк.	№ докум.	Підпис	Дата		

26. Palamar A., Karpinskyy M., Vodovozov V. Design and Implementation of a Digital Control and Monitoring System for an AC/DC UPS. 7th International Conference-Workshop «Compatibility and Power Electronics» (CPE 2011), June 1-3, 2011. P. 173–177.

27. Palamar A., Karpinskyy M. Control of an Uninterruptible Power Supply in a DC Microgrid System. 10th International Symposium "Topical Problems in the Field of Electrical and Power Engineering" and "Doctoral School of Energy and Geotechnology II" (January 10-15, 2011), Pärnu, Estonia, 2011. P. 80-84.

28. Погребенник В.Д., Клим Г.І., Бордун І.М., Пташник В.В., Паламар А.М. Системи оперативного контролю інтегральних параметрів водного середовища. Т. 2. Елементи комп'ютерних систем оперативного контролю: колективна монографія. Житомир: Видавничий дім «Бук-Друк», 2021. 180 с.

29. Voloskyi V., Leshchyshyn Y., Romanyshyn N., Palamar A., Tarasenko L. Method and algorithm for efficient cell balancing in the lithium-ion battery control system. CEUR Workshop Proceedings, The 1st International Workshop on Bioinformatics and Applied Information Technologies (BAIT 2024), Zboriv, Ukraine, October 02-04, 2024. Vol. 3842. P. 258-267.

30. Palamar A. Intelligent control and monitoring module for uninterruptible power supply system. II International Scientific and Practical Conference «Theoretical and Applied Aspects of Device Development on Microcontrollers and FPGAs» (MC&FPGA-2020), Kharkiv, Ukraine. 2020. P. 12-13.

31. Palamar A. Control system simulation by modular uninterruptible power supply unit with adaptive regulation function. Scientific Journal of TNTU, Ternopil, Ukraine, 2020. Vol. 98, No 2. P. 129–136.

32. Желібо Є. П. Заверуха Н.М., Зацарний В.В. Безпека життєдіяльності. Навчальний посібник. К.: Каравела, 2004. 328 с.

33. Скобло Ю.С., Соколовська Т.Б., Морозенко Д.І. та ін. Безпека життєдіяльності. Навчальний посібник для вищих навчальних закладів 3-4 рівнів акредитації. К.: Кондор, 2003. 424 с.

34. Жидецький В.Ц., Джигирей В.С., Мельников О.В. Основи охорони праці. Львів: Афіша, 2000. 350 с.

					КС КРБ 123.196.00.00 ПЗ	Арк.
						66
Змн.	Арк.	№ докум.	Підпис	Дата		

Додаток А  
Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Кафедра комп'ютерних систем та мереж

«ЗАТВЕРДЖУЮ»

Завідувач кафедру КС

\_\_\_\_\_ Осухівська Г.М.

“ 2 ” лютого 2026 р.

КОМП'ЮТЕРНА СИСТЕМА БЕЗКОНТАКТНОГО КЕРУВАННЯ ІОТ-  
ПРИСТРОЯМИ НА ОСНОВІ РОЗПІЗНАВАННЯ ТА ІНТЕРПРЕТАЦІЇ  
ЖЕСТІВ

ТЕХНІЧНЕ ЗАВДАННЯ

на  8  листках

Вид робіт: Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»

Керівник кваліфікаційної роботи

\_\_\_\_\_ к.т.н. Паламар А.М.

“ 2 ” лютого 2026 р.

«ВИКОНАВЕЦЬ»

Студент групи СІ-42

\_\_\_\_\_ Ткачук С.А.

“ 2 ” лютого 2026 р.

Тернопіль 2026

## 1 Загальні відомості

### 1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи бакалавра: «Комп'ютерна система безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.196.00.00.

### 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, Ткачук Сергій Андрійович.

### 1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи бакалавра є наказ по університету № 4/9-188 від «24» квітня 2026 року.

### 1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи бакалавра – 26.01.2026 р.

Плановий термін завершення виконання кваліфікаційної роботи бакалавра – 21.06.2026 р.

## 1.5 Порядок оформлення та пред'явлення результатів роботи

Оформлення технічної документації до кваліфікаційної роботи бакалавра здійснюється згідно діючих вимог вітчизняних та міжнародних стандартів. Технічна документація до кваліфікаційної роботи бакалавра включає в себе текст пояснювальної записки та креслення, які максимально інформативно та стисло відображають основні результати розробки комп'ютерної системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів. Основними регламентними документами при оформленні та пред'явленні результатів проектування є групи діючих стандартів ДСТУ, ISO, ЄСКД та ЕСПД. Пред'явлення результатів кваліфікаційної роботи бакалавра відбувається шляхом захисту роботи на відповідному засіданні ДЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

## 2 Призначення і цілі створення системи

### 2.1 Призначення системи

Комп'ютерна система безконтактного керування призначена для приймання, обробки та інтерпретації жестів користувача з подальшим формуванням керуючих команд для IoT-пристроїв. Система може застосовуватися в побутових умовах, у середовищах «розумного дому», навчальних і лабораторних установах, а також у системах автоматизації, де необхідне безконтактне або швидке керування обладнанням.

Основним призначенням системи є підвищення зручності та безпеки керування IoT-пристроями, зменшення залежності від фізичних елементів керування та розширення можливостей людино-машинної взаємодії.

## 2.2 Мета створення системи

Метою створення системи є розроблення функціонально завершеного апаратно-програмного комплексу, який забезпечує надійне розпізнавання жестів користувача та керування IoT-пристроями в реальному часі з використанням сучасних технологій Інтернету речей. Досягнення мети передбачає поєднання апаратних сенсорів, алгоритмів обробки даних і програмного забезпечення з можливістю інтеграції з IoT-платформами.

## 2.3 Характеристика об'єкту

Об'єктом керування в межах даної системи є IoT-пристрої, здатні приймати команди через мережеві або локальні інтерфейси зв'язку. Це можуть бути виконавчі модулі, освітлювальні прилади, реле, інтелектуальні розетки або інші пристрої автоматизації. Користувач взаємодіє з системою за допомогою жестів, які фіксуються сенсорними або оптичними модулями та перетворюються на керуючі сигнали.

## 3 Вимоги до системи

### 3.1 Вимоги до системи в цілому

Система повинна функціонувати як єдиний апаратно-програмний комплекс, що забезпечує стабільне та коректне безконтактне керування IoT-пристроями. Вона має бути зручною у використанні, надійною в роботі та адаптованою до умов тривалої експлуатації. Архітектура системи повинна забезпечувати можливість масштабування та модернізації без суттєвих змін базових компонентів.

### 3.1.1 Вимоги до структури та функціонування системи

Структура системи повинна включати сенсорний модуль для зчитування жестів, обчислювальний блок для обробки даних, модуль зв'язку та програмне забезпечення для інтерпретації жестів і формування команд. Функціонування системи має передбачати безперервний цикл зчитування даних, аналізу, прийняття рішень і передачі керуючих сигналів.

До основних функціональних блоків системи належать:

- модуль збору даних з сенсорів;
- блок обробки та розпізнавання жестів;
- модуль формування керуючих команд;
- інтерфейс взаємодії з IoT-платформою.

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

Обмін даними між компонентами системи повинен здійснюватися з використанням стандартних цифрових інтерфейсів і мережевих протоколів. Для локального зв'язку між мікроконтролером і сенсорами можуть використовуватися інтерфейси I<sup>2</sup>C, SPI або UART. Для передавання даних до IoT-платформи та керування віддаленими пристроями передбачається використання бездротових технологій зв'язку, зокрема Wi-Fi або Bluetooth.

Зв'язок має бути стабільним, захищеним від втрати даних та затримок, що можуть негативно впливати на швидкодію системи.

### 3.1.3 Вимоги до режимів функціонування системи

Система повинна підтримувати кілька режимів роботи, зокрема режим ініціалізації, основний режим керування та режим налаштування. В основному режимі система здійснює розпізнавання жестів у реальному часі та керування IoT-пристроями. У режимі налаштування користувач або оператор має

можливість змінювати параметри розпізнавання, чутливість сенсорів і відповідність жестів керуючим командам.

#### 3.1.4 Перспективи розвитку та модернізації системи

Архітектура системи повинна передбачати можливість подальшого розвитку, зокрема:

- розширення набору розпізнаваних жестів;
- підключення додаткових IoT-пристроїв;
- інтеграцію з іншими хмарними сервісами;
- використання алгоритмів машинного навчання для підвищення точності розпізнавання.

#### 3.1.5 Вимоги до надійності системи

Система повинна забезпечувати безперебійну роботу протягом тривалого часу без втрати функціональності. Вимоги до надійності включають стійкість до збоїв зв'язку, коректну обробку помилкових або некоректних жестів, а також відновлення роботи після перезапуску. Програмне забезпечення має бути захищене від критичних збоїв і помилок виконання.

Показники надійності системи безконтактного керування пристроями на основі розпізнавання та інтерпретації жестів повинні відповідати вимогам ДСТУ 50136-1. Ймовірність безвідмовної роботи системи повинна складати не менше 99,6 %.

#### 3.1.6 Вимоги до функцій та задач, які виконує система

Система повинна виконувати такі основні функції:

- зчитування даних з сенсорів жестів;
- обробка та фільтрація сигналів;

- розпізнавання та інтерпретація жестів;
- формування та передавання керуючих команд;
- обмін даними з IoT-платформою;
- моніторинг стану системи та підключених пристроїв.

### 3.1.7 Вимоги до апаратного забезпечення

Апаратне забезпечення системи має базуватися на сучасному мікроконтролері з достатніми обчислювальними ресурсами та підтримкою бездротового зв'язку. Сенсорні модулі повинні забезпечувати необхідну точність і стабільність вимірювань. Усі компоненти мають відповідати вимогам енергоефективності, компактності та електробезпеки.

Вимоги до мікроконтролера:

- мікроконтролер має підтримувати RISC архітектуру команд;
- мікроконтролер повинен містити необхідний набір вбудованих периферійних пристроїв (таймери, АЦП і т.п.) та потрібну кількість керованих портів введення /виведення.

## 4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ.

Комплект конструкторської документації повинен складатись з:

- пояснювальної записки;
- графічного матеріалу:
  1. структурна схема системи;
  2. схема електрична принципова;
  3. блок-схема алгоритму роботи;
  4. результати моделювання системи.

\*Примітка: В комплект конструкторської документації можуть вноситися зміни та доповнення в процесі розробки.

## 5 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання КРБ

№ етапу	Назва етапу виконання КРБ	Термін виконання
1.	Розробка технічного завдання	26.01 – 02.02
2.	Аналіз технічного завдання, вимог до комп'ютерної системи, та можливих рішень поставленого завдання	03.02 – 15.02
3.	Розроблення структури, вибір апаратного забезпечення, проектування комп'ютерної системи	20.04 – 25.04
4.	Реалізація алгоритму, написання програмного забезпечення, моделювання комп'ютерної системи	26.04 – 05.05
5.	Робота над четвертим розділом «Безпека життєдіяльності, основи охорони праці»	07.05 – 25.05
6.	Оформлення пояснювальної записки та графічного матеріалу	26.05 – 07.06
7.	Перевірка на академічний плагіат, перевірка керівником та консультантами	08.06 – 14.06
8.	Попередній захист кваліфікаційної роботи бакалавра	15.06 – 21.06
9.	Захист кваліфікаційної роботи бакалавра	22.06.2026

## 6 Додаткові умови виконання кваліфікаційної роботи бакалавра

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

Додаток Б  
Перелік елементів



## Додаток В

### Лістинг програми

Лістинг В.1 – Код програми мікроконтролера для реалізації системи безконтактного керування IoT-пристроями на основі розпізнавання та інтерпретації жестів.

```
#define BLYNK_PRINT Serial

#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <Wire.h>
#include <Adafruit_APDS9960.h>
#include <Adafruit_NeoPixel.h>
#include <ESP32Servo.h>
#include <TFT_eSPI.h>
#include "secret.h"

// ----- DISPLAY -----
TFT_eSPI tft = TFT_eSPI();
// ----- APDS -----
Adafruit_APDS9960 apds;
#define APDS_INT 15
volatile bool gestureFlag = false;
// ----- LED STRIP -----
#define LED_PIN 13
#define NUM_LEDS 8
Adafruit_NeoPixel strip(NUM_LEDS, LED_PIN, NEO_GRB + NEO_KHZ800);
// ----- SERVO -----
#define SERVO_PIN 26
Servo servo;
int servoPos = 90;
// ----- BUZZER -----
#define BUZZER_PIN 25
// ----- STATUS LED -----
#define STATUS_LED 33
// ----- RELAY -----
#define RELAY_PIN 32
bool relayState = false;
// ----- VARIABLES -----
int brightness = 100;
int mode = 0; // 0-light,1-servo,2-relay
bool isOnline = false;
// ----- GESTURE FILTER -----
unsigned long lastGestureTime = 0;
int lastGesture = -1;
const int GESTURE_DELAY = 400;
```

```

// ----- SETUP -----
void setup() {
  Serial.begin(115200);
  tft.init();
  tft.setRotation(1);
  showMessage("System Init...");
  pinMode(STATUS_LED, OUTPUT);
  pinMode(BUZZER_PIN, OUTPUT);
  pinMode(RELAY_PIN, OUTPUT);
  digitalWrite(RELAY_PIN, LOW);
  // INT
  pinMode(APDS_INT, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(APDS_INT), apdsISR,
FALLING);
  // Servo
  servo.attach(SERVO_PIN);
  servo.write(servoPos);
  // LEDs
  strip.begin();
  strip.setBrightness(brightness);
  strip.show();
  // I2C
  Wire.begin(21, 22);
  // APDS
  if (!apds.begin()) {
    showMessage("APDS ERROR");
    while (1);
  }
  apds.enableGesture(true);
  connectWiFi();
  showMessage("READY");
  delay(1000);
}

// ----- LOOP -----
void loop() {
  if (isOnline) {
    Blynk.run();
  }
  if (gestureFlag) {
    gestureFlag = false;
    while (apds.gestureAvailable()) {
      int gesture = apds.readGesture();
      processGesture(gesture);
    }
  }
  delay(10);
}

// ----- INTERRUPT -----
void IRAM_ATTR apdsISR() {
  gestureFlag = true;
}

```

```

}

// ----- DISPLAY -----
void showMessage(String msg) {
  tft.fillScreen(TFT_BLACK);
  tft.setTextSize(2);
  tft.setCursor(10, 50);
  tft.println(msg);
}

void drawUI(String gesture) {
  tft.fillScreen(TFT_BLACK);
  tft.setTextSize(2);
  tft.setCursor(0, 0);
  tft.println("Gesture Control");
  tft.setTextSize(1);
  tft.setCursor(0, 30);
  tft.printf("Mode: %d\n", mode);
  tft.printf("Gesture: %s\n", gesture.c_str());
  tft.printf("Bright: %d\n", brightness);
  tft.printf("Servo: %d\n", servoPos);
  tft.printf("Relay: %s\n", relayState ? "ON" : "OFF");
  tft.printf("WiFi: %s\n", isOnline ? "OK" : "OFF");
}

// ----- WIFI -----
void connectWiFi() {
  WiFi.begin(ssid, pass);
  showMessage("Connecting WiFi...");
  unsigned long start = millis();
  while (WiFi.status() != WL_CONNECTED && millis() - start < 8000) {
    delay(200);
  }

  if (WiFi.status() == WL_CONNECTED) {
    isOnline = true;
    showMessage("WiFi Connected");
    Blynk.config(auth);
    Blynk.connect();
  } else {
    isOnline = false;
    showMessage("Offline Mode");
  }
  delay(1000);
}

// ----- FUNCTIONS -----
void setAllLEDS(uint8_t r, uint8_t g, uint8_t b) {
  for (int i = 0; i < NUM_LEDS; i++) {
    strip.setPixelColor(i, strip.Color(r, g, b));
  }
  strip.show();
}

```

```

void alarmSignal() {
    digitalWrite(STATUS_LED, HIGH);
    tone(BUZZER_PIN, 2000, 500);
    if (isOnline) {
        Blynk.logEvent("alarm", "Потрібна допомога!");
    }
    delay(500);
    digitalWrite(STATUS_LED, LOW);
}

// ----- BLYNK -----
BLYNK_WRITE(V0) {
    brightness = param.asInt();
    strip.setBrightness(brightness);
    strip.show();
}

BLYNK_WRITE(V1) {
    servoPos = param.asInt();
    servo.write(servoPos);
}

BLYNK_WRITE(V2) {
    mode = param.asInt();
}

BLYNK_WRITE(V4) {
    relayState = param.asInt();
    digitalWrite(RELAY_PIN, relayState);
}

// ----- GESTURE -----
void processGesture(int gesture) {
    // фільтрація
    if (gesture == lastGesture && millis() - lastGestureTime <
GESTURE_DELAY) return;
    lastGesture = gesture;
    lastGestureTime = millis();
    String name = "";
    switch (gesture) {
        case APDS9960_UP:
            name = "UP";
            if (mode == 0) {
                brightness = min(255, brightness + 20);
                strip.setBrightness(brightness);
                strip.show();
            }
            if (mode == 2) {
                relayState = !relayState;
                digitalWrite(RELAY_PIN, relayState);
            }
            break;
    }
}

```

```

case APDS9960_DOWN:
  name = "DOWN";
  if (mode == 0) {
    brightness = max(0, brightness - 20);
    strip.setBrightness(brightness);
    strip.show();
  }
  break;
case APDS9960_LEFT:
  name = "LEFT";
  if (mode == 1) {
    servoPos = max(0, servoPos - 20);
    servo.write(servoPos);
  }
  break;
case APDS9960_RIGHT:
  name = "RIGHT";
  if (mode == 1) {
    servoPos = min(180, servoPos + 20);
    servo.write(servoPos);
  }
  break;
case APDS9960_NEAR:
  name = "MODE";
  mode++;
  if (mode > 2) mode = 0;
  break;

case APDS9960_FAR:
  name = "ALARM";
  alarmSignal();
  break;
}
drawUI(name);
if (isOnline) {
  Blynk.virtualWrite(V0, brightness);
  Blynk.virtualWrite(V1, servoPos);
  Blynk.virtualWrite(V2, mode);
  Blynk.virtualWrite(V4, relayState);
}
}

```