

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

перший (бакалаврський)

(рівень вищої освіти)

Розроблення технічного засобу візуалізації стану робота-промоутера

(тема)

Виконав:

здобувач 3 року навчання,
групи АКТАКІТу 22-1

Ярослав БЄЛИЙ

(власне ім'я, прізвище)

Спеціальність 151 Автоматизація та
комп'ютерно інтегровані технології

(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Автоматизація,
комп'ютерно-інтегровані технології та
робототехніка

(повна назва освітньої програми)

Керівник доцент Оксана СИЧОВА

(посада, власне ім'я, прізвище)

Допускається до захисту
Зав. кафедри КІТАР

(підпис)

Ігор НЕВЛЮДОВ

(власне ім'я, прізвище)

2025 р.

Я , Бєлий Ярослав Володимирович, як здобувач вищої освіти ХНУРЕ, розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Я не використовував штучний інтелект для підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

"7" Липня 2025 р.



Ярослав БЄЛИЙ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет _____ АКТ _____
Кафедра _____ КІТАР _____
Рівень вищої освіти _____ перший (бакалаврський) _____
Спеціальність _____ 151 Автоматизація та комп'ютерно-інтегровані технології _____
(код і повна назва)
Тип програми _____ Освітньо-професійна _____
Освітня програма _____ Автоматизація, комп'ютерно-інтегровані технології _____
та робототехніка _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри КІТАР _____
(підпис)

« 19 » травня 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Здобувачеві _____ Белому Ярославу Володимировичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Розроблення технічного засобу візуалізації стану робота-промоутера _____

Затверджена наказом по університету від _____ 21.05.2025 № 405 Ст _____

2. Термін подання здобувачем роботи до екзаменаційної комісії _____ 15.07.2025 _____

3. Вихідні дані до роботи _____

3.1 Мікроконтролер Raspberry Pi Zero 2 W; _____

3.2 Застосування сервоприводів для руху голови, очей та роту; _____

3.3 Контролер PCA9658 для керування сервоприводами по інтерфейсу I2C; _____

3.4 Плавне керування сервоприводами за допомогою П-регулятора; _____

3.5 Голосове керування з використанням MEMS мікрофону. _____

4. Перелік питань, що потрібно опрацювати в роботі _____

4.1 Вступ; _____

4.2 Аналіз сучасного стану технічних засобів для візуалізації роботів; _____

4.3 Розробка структури робота-промоутера; _____

4.4 Складання апаратної частини робота-промоутера; _____

4.5 Розробка програмного забезпечення та налаштування середовища виконання; _____

4.6 Дослідження стійкості системи автоматичного регулювання; _____

4.7 Висновки. _____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри)

Графічний матеріал у вигляді презентації – 5 арк. ф.А4

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз сучасного стану технічних засобів для автоматичної візуалізації стану роботів	05.06 – 10.06.25	виконано
2	Розробка структури робота-промоутера	12.06 – 18.06.25	виконано
3	Складання апаратної частини робота-промоутера	20.06 – 22.06.25	виконано
4	Розробка програмного забезпечення та налаштування середовища виконання	23.06 – 24.06.25	виконано
5	Дослідження стійкості системи автоматичного регулювання	30.06 – 2.07.25	виконано
6	Подання роботи на перевірку Інтернет-сервісом StrikePlagiarism	08.07 – 10.08.25	виконано
7	Оформлення пояснювальної записки	10.08 – 11.08.25	виконано
8	Подання роботи на рецензію	11.08 – 12.08.25	виконано
9	Подання роботи на підпис зав. кафедри	12.08 – 13.08.25	виконано
10	Подання кваліфікаційної роботи в ЕК	13.08 – 15.08.25	виконано

Дата видачі завдання 19.05.2025 р.

Здобувач _____
(підпис)

Ярослав БЄЛИЙ _____
(власне ім'я, прізвище)

Керівник роботи _____
(підпис)

доц. Сичова О.В. _____
(власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 52 с., 2 табл., 27 рис., 2 додатків, 13 джерел.

RASPBERRY PI ZERO 2W, РОБОТ ПРОМОУТЕР, СЕРВОПРИВОДИ, РОЗПІЗНАВАННЯ ОБЛИЧЬ, ОБРОБКА ЗОБРАЖЕНЬ (OPENCV), П-РЕГУЛЯТОР, РОЗПІЗНАВАННЯ МОВЛЕННЯ, ГОЛОСОВЕ КЕРУВАННЯ.

Об'єкт розробки – процес візуального інформування користувача про поточний стан системи керування роботом-промоутером.

Предмет розробки – технічні та програмні засоби автоматичної візуалізації станів роботизовано системи в режимі реального часу.

Мета роботи – візуалізація стану робота-промоутера під час його взаємодії з користувачем шляхом розроблення технічного засобу.

У межах роботи розроблено прототип робота-промоутера на базі Raspberry Pi, здатного до базової взаємодії з користувачем за допомогою зору, голосу та рухів. Визначено вимоги до системи, спроектовано конструкцію пристрою, підібрано необхідні апаратні компоненти та реалізовано апаратну частину. Розроблено програмне забезпечення для обробки зображень, розпізнавання мовлення, керування сервоприводами робота. Система доповнена веб-інтерфейсом для налаштування і керування. Проведено базове дослідження стійкості системи керування рухами з використанням П-регулятора. Результатом стала функціональна демонстраційна платформа, придатна для подальшого розвитку в галузі сервісної робототехніки.

Проект сприяє реалізації цілей сталого розвитку України:

- ціль 4 – якісна освіта (практична підготовка ІТ-фахівців);
- ціль 9 – індустріалізація та інновації (розробка інтелектуальної системи);
- ціль 12 – раціональне виробництво (енергоефективність, модульність).

ABSTRACT

Explanatory note: 52 pages, 27 pictures, 13 sources.

RASPBERRY PI ZERO 2W, PROMOTIONAL ROBOT, SERVOMOTORS, FACE DETECTION, IMAGE PROCESSING (OPENCV), P CONTROLLER, SPEECH RECOGNITION, VOICE CONTROL

Object of development – the process of visual informing the user about the current state of the control system of a promotional robot.

Subject of development – technical and software tools for automatic real-time visualization of robotic system states.

Objective of the work – visualization of the promoter robot's state during its interaction with the user through the development of a technical device.

The project involves the development of a prototype promotional robot based on Raspberry Pi, capable of basic user interaction through vision, speech, and motion. System requirements were defined, the device was designed, necessary hardware modules were selected, and the hardware part was implemented. Software modules were developed for image processing, speech recognition, servo control, and real-time state visualization. A web interface was also created for configuration and control. Basic stability analysis of the motion control system using a proportional (P) controller was performed. The result is a functional demonstration platform suitable for further research in the field of service robotics.

The project contributes to the implementation of Ukraine's Sustainable Development Goals:

- goal 4 – quality education (practical training of IT specialists);
- goal 9 – industry, Innovation and Infrastructure (development of intelligent systems);
- goal 12 – responsible consumption and production (energy efficiency and modularity).

ЗМІСТ

Перелік скорочень	5
Вступ.....	6
1 Аналіз сучасного стану технічних засобів для автоматичної візуалізації стану роботів	7
1.1 Аналіз наявних рішень	7
1.2 Постановка задачі.....	12
2 Розробка структури робота-промоутера.....	13
2.1 Розробка загальної структури схеми робота-промоутера.....	13
2.2 Модуль розпізнавання обличчя	15
2.3 Модуль керування відтворенням аудіофайлів з DFPlayer	16
2.4 Модуль керування сервоприводами.....	18
2.5 Модуль розпізнавання мовлення VOSK.....	19
2.6 Модуль розпізнавання мовлення Google STT.....	20
2.7 Скрипт створення анімацій мовлення робота	21
2.8 Модуль виявлення мовленнєвої активності	23
3 Складання апаратної частини робота-промоутера	25
3.1 Вибір апаратних модулів.....	25
3.2 Створення електричної схеми підключення	32
3.3 Інструкція з охорони праці при лудінні і паянні	37
4 Розробка програмного забезпечення та налаштування середовища виконання	39
4.1 Налаштування середовища виконання	39
4.2 Програмний засіб для налаштувань та керування роботом.....	43
5 Дослідження стійкості системи автоматичного регулювання	47
Висновки	50
Додаток А Апробація результатів роботи	53
Додаток Б Код програмного засобу	58
Додаток В Демонстраційний матеріал.....	97

ПЕРЕЛІК СКОРОЧЕНЬ

АМР – Автономний мобільний робот;

БД – база даних;

НДР – науково-дослідна робота;

ОС – операційна система;

ПЗ – програмне забезпечення;

РП – роботизований пристрій;

СК – система керування.

ВСТУП

Сучасні тенденції розвитку робототехніки передбачають не лише автоматизацію фізичних дій, а й інтерактивну взаємодію з людиною. Роботи-промоутери – це приклад соціально орієнтованих технічних систем, основне завдання яких полягає в презентації інформації, приверненні уваги та комунікації з користувачами. Візуалізація стану таких роботів відіграє ключову роль у формуванні інтуїтивного та ефективного інтерфейсу між машиною та людиною.

На практиці ефективність роботи робота-промоутера значною мірою залежить від спроможності технічного засобу демонструвати свої поточні дії, наміри та реакцію. Це потребує синхронної роботи апаратних компонентів, модулів та алгоритмів управління.

Метою цієї кваліфікаційної роботи є візуалізація стану робота-промоутера під час його взаємодії з користувачем шляхом розроблення технічного засобу. Для досягнення поставленої задачі виконано аналіз сучасних технічних рішень, реалізовано програмні модулі для розпізнавання обличчя, мовлення та керування сервоприводами, сформована пояснювальна записка відповідно стандартам ДСТУ [1, 2, 3].

Актуальність теми полягає в зростаючому попиті на інтелектуальні системи взаємодії у сфері обслуговування, рекламі та освіті, де використання таких роботів підвищує ефективність комунікації. Також актуальність підтверджується попередніми напрацюваннями в галузі розпізнавання мовлення для робототехнічних систем, зокрема результатами, опублікованими у матеріалах всеукраїнської науково-практичної конференції [10].

Практична реалізація проекту демонструє можливість створення доступного й адаптивного рішення з використанням доступних компонентів та матеріалів, а також можливістю подальшої інтеграції в різноманітні робототехнічні системи.

1 АНАЛІЗ СУЧАСНОГО СТАНУ ТЕХНІЧНИХ ЗАСОБІВ ДЛЯ АВТОМАТИЧНОЇ ВІЗУАЛІЗАЦІЇ СТАНУ РОБОТІВ

1.1 Аналіз наявних рішень

У сучасних робототехнічних системах зростає потреба у зрозумілому та адаптивному інтерфейсі взаємодії між користувачем та системою.

Автоматизована візуалізація стану робота відіграє ключову роль у підвищенні ефективності взаємодії та довіри до роботизованих платформ. Вона дозволяє користувачеві оперативно оцінювати наміри та дії системи, що важливо у соціальному застосуванні.

Технічні засоби візуалізації стану робота можна умовно поділити за рівнем складності, типом взаємодії та функціональним призначенням. Найпростіші з них це звичайні світлодіоди, які сигналізують про базові стани системи, наприклад, низький заряд батареї, помилка в системі, підключення до інтернету. Завдяки кольору оператор може легко орієнтуватися в поточному стані системи. RGB – світлодіоди додають можливість створювати динамічні світлові ефекти зміни кольору або пульсації, які можуть бути ще більш інформативними. Наприклад, зміна кольору від червоного до зеленого може сигналізувати про статус заряду акумулятора, а пульсація червоного світлодіода свідчатиме про важливу помилку в системі.

Для більш точного відображення даних застосовують сегментні індикатори – вони показують цифри або окремі символи. Якщо потрібно відобразити просте зображення або графіку, використовуються світлодіодні матриці. Їх часто можна побачити у вигляді очей робота, що змінюють вираз або напрямок погляду.

До більш складних елементів візуалізації відносять OLED та LCD дисплеї. OLED дисплеї компактні та добре підходять для виводу тексту, графіки чи емоцій робота. LCD або TFT дисплеї забезпечують вищу роздільну

здатність і кольорову передачу, тож здатні відтворювати відео або деталізоване зображення. Їх часто використовують для створення обличчя робота.

Сенсорні екрани поєднують вивід інформації з керуванням. Користувач може взаємодіяти з інтерфейсом, обирати режими або виконувати прості дії. У деяких системах застосовуються прості проекційні засоби – лазерні або світлодіодні проектори, які дають змогу виводити зображення на навколишні поверхні.

Окремим видом візуалізації є рухомі елементи. За допомогою компактних сервомоторів або актуаторів можна змінювати положення брів, повік, рота та інших частин обличчя. Такі системи забезпечують більш глибоку експресивність, дозволяючи відтворювати складні вирази обличчя.

Ключова перевага таких систем – фізична присутність у просторі. В таких системах немає проблем з яскравістю дисплеїв або кутом огляду. На основі таких систем можлива побудова реалістичних обличь, які здатні відтворювати не лише базові рухи, але й емоції, мімічні нюанси людини: асиметрію посмішки, мікрорухи брів, імітацію дихання.

Одним з прикладів систем із розвиненою візуалізацією є робот платформа Misty II [4], який зображений на рисунку 1.1.



Рисунок 1.1 – Робот-платформа Misty II

Misty II – це програмований робот, розроблений компанією Misty Robotics. Проект позиціонує себе для потреб розробників, дослідників та ентузіастів робототехніки. Платформа орієнтована на створення інтерактивних додатків з використанням штучного інтелекту та сенсорної взаємодією з користувачем. Робот підтримує розпізнавання обличь, емоцій, мовлення, а також має функцію синтезу мовлення.

Конструкція Misty II має антропоморфні риси: рухому голову та руки з сенсорами дотику. Руки здебільшого використовуються для простих жестів і визначення фізичної взаємодії з користувачем. Ходова частина базується на мобільній платформі з колесами.

У технічному оснащенні Misty II передбачено:

- фронтальну 4К-камеру (яка розташована над дисплеєм);
- глибинну камеру для побудови тривимірних карт місцевості;
- три масиви мікрофонів для визначення напрямку джерела звуку;
- сенсори дотику на голові, спині та руках;
- лідари для навігації та уникнення перешкод;
- акселерометр, гіроскоп та датчик освітленості.

Програмна архітектура Misty II базується на Misty SDK, який підтримує JavaScript, Python і C# через REST API та WebSocket.

Робот підтримує інтеграцію з хмарними ШІ-сервісами Microsoft Azure Cognitive Services, Google AI та Amazon Alexa, що дозволяє облегшити реалізацію функцій розпізнавання обличь, мовлення та емоцій.

Misty II є відкритою платформою з можливістю встановлення додаткових апаратних модулів.

Реалізація візуального зворотного зв'язку відбувається через дисплей, вбудований у голову робота. Екран відображає набори піксельних анімацій, що відповідають запрограмованим емоціям. Система прив'язана до подій у середовищі (виявлення обличчя, розпізнавання мови, жести) на які робот реагує зміною виразу обличчя. Ці дані генеруються за допомогою програмної логіки Misty SDK, де події датчиків або команди розробника запускають

відповідні анімації або фрейми на дисплеї.

Водночас Misty II має низку недоліків. Одним із основних обмежень є висока вартість, що на момент написання становить близько 120 000 гривень, що ускладнює доступність пристрою для індивідуального використання або малих навчальних закладів. Для користувачів без попереднього досвіду в галузі програмування чи робототехніки первинне налаштування та розробка можуть бути складними, незважаючи на наявність спільноти. Фізичні можливості пристрою також обмежені – конструкція робота не передбачає виконання складних маніпуляцій або фізичних дій, а його руки мають скоріше декоративну, ніж функціональну роль. Додатковим обмеженням є залежність Misty II від постійного доступу до інтернету для використання хмарних сервісів, що створює труднощі в автономному режимі.

Гуманоїдного соціального робота Pepper [5] зображено на рисунку 1.2.



Рисунок 1.2 – Гуманоїдний робот Pepper

Pepper – це один із найрозвиненіших гуманоїдних роботів у світі, розроблений компанією SoftBank Robotics (тепер Aldebaran Robotics). Його створено для взаємодії з людьми, розпізнавання емоцій, голосових команд та виконання функцій асистента в бізнесі, освіті, сфері обслуговування та домашньому використанні.

Pepper оснащений 3D-камерами та сенсорами для розпізнавання обличь, виявлення емоцій та ідентифікації людей. На основі отриманих даних він може аналізувати міміку та тон голосу співрозмовника, визначаючи рівень його емоцій.

Голосове управління реалізоване за допомогою чотирьох вбудованих масивів мікрофонів, що дозволяють чути команди з різних напрямків. Його мовний модуль підтримує одночасно кілька мов, англійську, французьку, японську, що дає змогу використовувати його в міжнародних проектах. Pepper не просто реагує на запити користувачів, а й адаптується до них. Це один з небагатьох роботів такого типу що робить його одним із найрозвиненіших у світі.

Рухається Pepper за допомогою колісної бази та моторів. Він здатен плавно пересуватися у різних напрямках уникаючи перешкод навіть у натовпі. Для цього в його арсеналі є вбудовані сенсори: ультразвукові, інфрачервоні та лазерні.

Візуальний зворотній зв'язок забезпечується за рахунок дисплея, який розташовано на грудях, де виводиться текст, іконки, анімації або графічні елементи, що ілюструють емоційний або функціональний стан системи робота. Крім дисплея використовується кінематичні жести рук і голови.

Основне призначення Pepper – спілкуватися з людьми. Він може працювати консультантом, помічником або навчати дітей в школі. Його часто використовують в банках, готелях, магазинах та в освітніх установах.

Станом на зараз ціна робота від 700 тисяч до півтора мільйона гривень, в залежності від комплектацій та ліцензії.

1.2 Постановка задачі

Аналіз сучасних робототехнічних платформ показує, що вартісні або комерційні рішення системи візуалізації стану робота зазвичай є дорогими, технічно складними та обмежено доступними для самостійного виготовлення чи навчального застосування. У недорогих або аматорських конструкціях візуальний зворотній зв'язок реалізується на базовому рівні або взагалі відсутній.

Основною задачею даної роботи є розробка функціонального, простого та доступного модуля автоматичної візуалізації стану робота, який буде:

- низьковартісним у реалізації – з використанням легкодоступних компонентів та матеріалів;
- модульним за архітектурою – з можливістю розділення на окремі блоки для спрощення адаптації та масштабування;
- з відкритим програмним кодом – для забезпечення прозорості, повторного використання та подальшого розвитку проекту;
- легкомодифікованим – завдяки гнучкому налаштуванню логіки через конфігураційні файли або зручний інтерфейс;
- пристосованим до використання в умовах обмежених ресурсів – як апаратних (контролер з низькою продуктивністю), так і програмних (мінімальна залежність від сторонніх сервісів та підключення до інтернету).

Результатом має стати доступне технічне рішення, яке забезпечує базову візуальну комунікацію між роботом і користувачем та може бути легко адаптоване для демонстрацій, навчальних проектів або прототипів.

2 РОЗРОБКА СТРУКТУРИ РОБОТА-ПРОМОУТЕРА

2.1 Розробка загальної структури схеми робота-промоутера

Робот-промоутер призначений для взаємодії з користувачем за допомогою голосових команд та систем візуального стеження.

Розроблена контекстна діаграма IDEF0 зображена на рисунку 2.1.

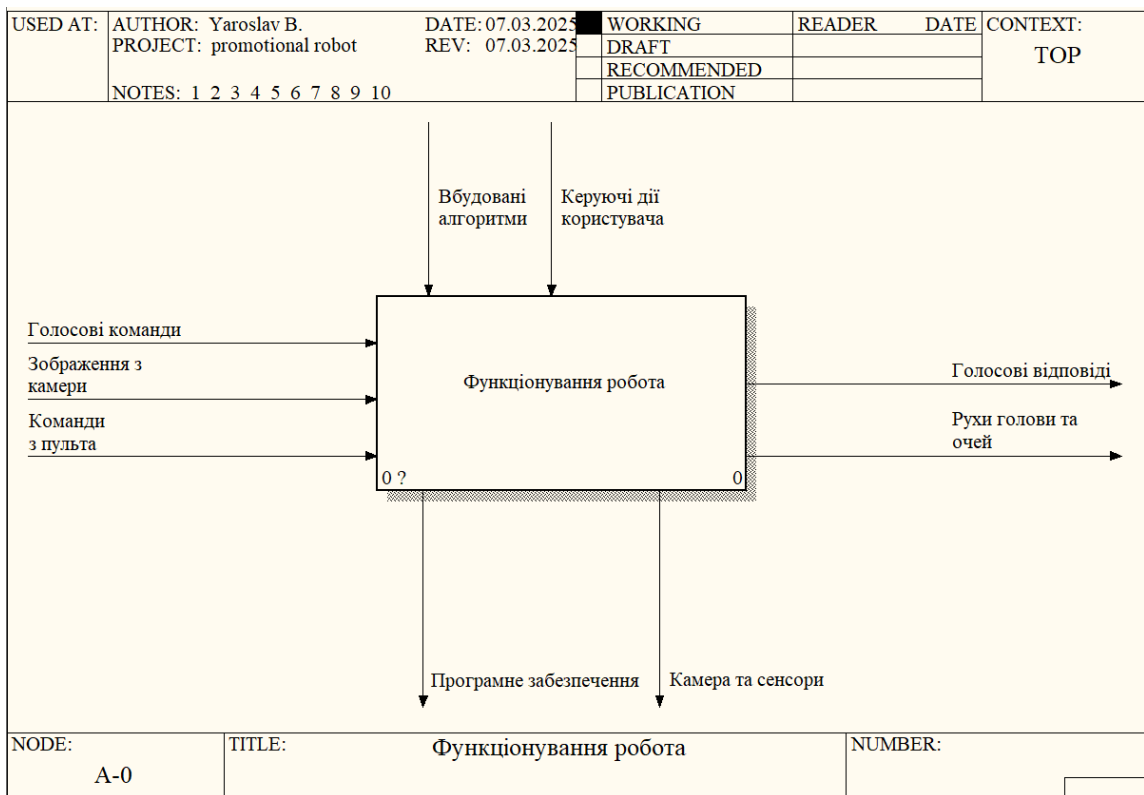


Рисунок 2.1 – Контекстна діаграма IDEF0 робота-промоутера

Декомпозиція контекстної діаграми зображена на рисунку 2.2.

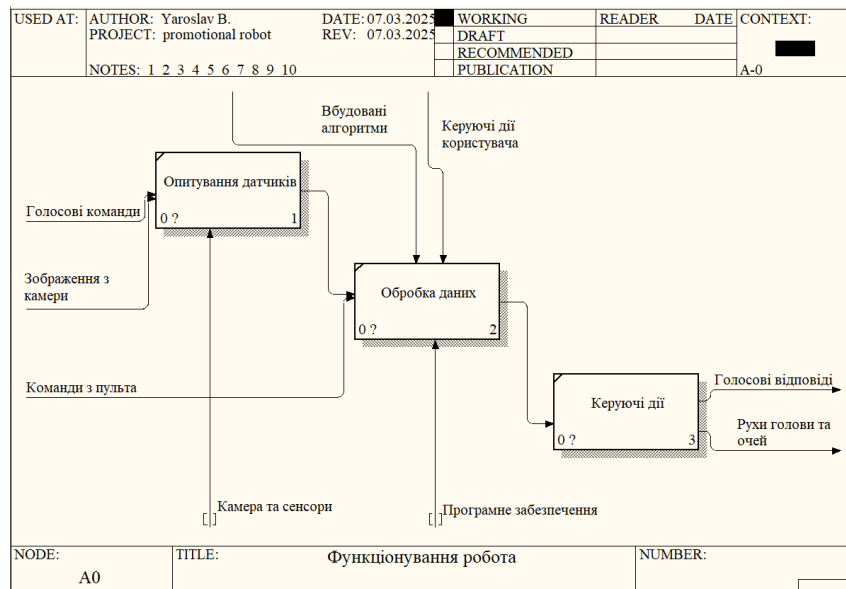


Рисунок 2.2 – Декомпозиція контекстної діаграми

Голосові команди можуть розпізнаватись як онлайн, за допомогою Google STT (Speech to text) сервісу, так і офлайн з використанням бібліотеки VOSK. Мовлення користувача спочатку оброблюється за допомогою бібліотеки webrtcvad, яка визначає наявність голосової активності мікрофона, після чого відправляє фрагмент аудіо до модуля розпізнавання голосу. Модуль розпізнавання голосу повертає рядок тексту, який звіряється зі списком команд та відтворює голосову відповідь та набір попередньо записаної анімації роту в разі співпадінні з відомою командою. Голосові відповіді попередньо записані у форматі аудіофайлу. Анімації роту представляють собою список кутів повороту сервоприводу, які відтворюються через однаковий проміжок часу.

Візуальне стеження реалізоване через камеру, яка передає зображення у бібліотеку cv2. Алгоритм визначає координати обличчя у кадрі та використовує їх для керування механічними компонентами робота. Очі рухаються за допомогою сервоприводів, що дозволяє роботу імітувати природне слідкування поглядом за користувачем. Голова також повертається у відповідному до користувача напрямку, забезпечуючи більш реалістичну взаємодію та розширює кут погляду робота. Рот відкривається та синхронно відтворює анімацію з програванням аудіо голосової відповіді.

2.2 Модуль розпізнавання обличчя

Алгоритм розпізнавання обличчя зображено на рисунку 2.3, базується на каскадному класифікаторі Хаара (Haar Cascade), що реалізований у бібліотеці OpenCV [7]. Спершу отриманий з камери кадр конвертується у відтінки сірого для спрощення обробки, оскільки алгоритм Хаара працює ефективніше з чорно-білими зображеннями. Далі виконується еквалізація гістограми отриманого сірого зображення для покращення контрасту та кращої видимості рис обличчя, що допомагає підвищити точність детекції.

Після підготовки зображення виконується пошук обличчя на кадрі за допомогою функції `detectMultiScale`. Ця функція аналізує зображення на різних масштабах, щоб виявити області, які відповідають ознакам обличчя. Вона повертає список координат та розмірів прямокутників, які вказують на знайдені обличчя. Для кожного знайденого обличчя алгоритм отримує позицію (координати верхнього лівого кута прямокутника) та розміри (ширина та висота).

Далі отримані координати використовуються для керування сервоприводами, що відповідають за рух очей та голови. Це дозволяє роботу “слідкувати” за обличчями в полі зору камери. На зображенні для візуалізації результатів навколо кожного знайденого обличчя малюється зелена рамка, яка допомагає користувачеві побачити, що саме система розпізнала.

Обробка кадрів виконується у циклі, що безперервно захоплює нові кадри з камери, аналізує їх та оновлює інформацію про обличчя у реальному часі. Для забезпечення безперервної роботи та уникнення блокування інтерфейсу обробка кадрів відбувається у окремому потоці, що дозволяє системі одночасно приймати голосові команди та виконувати інші завдання.

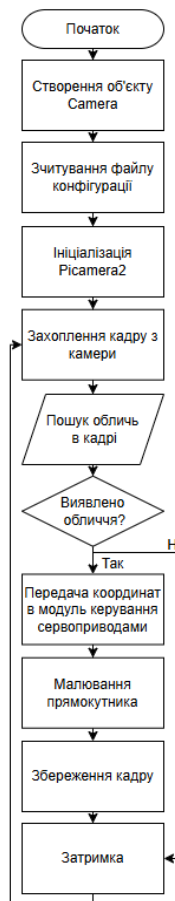


Рисунок 2.3 – Алгоритм роботи модуля розпізнавання обличчя

2.3 Модуль керування відтворенням аудіофайлів з DFPlayer

Блок схема алгоритму керування відтворенням аудіофайлів з DFPlayer зображено на рисунку 2.4.

Цей модуль реалізує клас DFPlayer для керування MP3-плеєром DFPlayer через послідовний порт UART.

Перш за все, модуль завантажує конфігурацію з файлу config/config.json, де зберігаються налаштування гучності відтворення.

При створенні об'єкту класу DFPlayer відкривається серійний порт з вказаним параметром порту /dev/ttyS0 і швидкістю 9600 бод. Після короткої паузи (2 секунди) виконується завантаження конфігурації, і одразу встановлюється гучність, яка задана у конфігурації.

Основний метод send_command формує команду у вигляді байтового масиву, який відповідає протоколу DFPlayer. Команда починається зі

стартового байту, включає версію, довжину, код команди, два параметри, та двобайтову контрольну суму, а завершується стоп-байтом.

Метод `play` приймає номер треку і відправляє команду плеєру для початку відтворення відповідного треку. Номер треку розбивається на два байти – старший та молодший.

Метод `reload_volume` перезавантажує налаштування гучності з конфігураційного файлу і застосовує нове значення.

В кінці модуля створюється екземпляр класу `DFPlayer` готовий до керування плеєром.

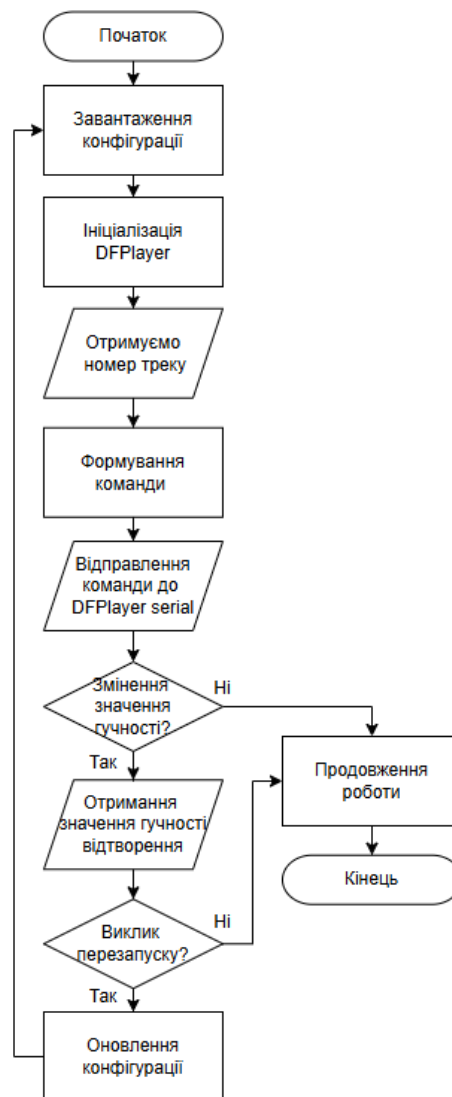


Рисунок 2.4 – Блок схема алгоритму керування відтворенням аудіофайлів з DFPlayer

2.4 Модуль керування сервоприводами

Алгоритм роботи модуля керування сервоприводами наведено на рисунку 2.5.

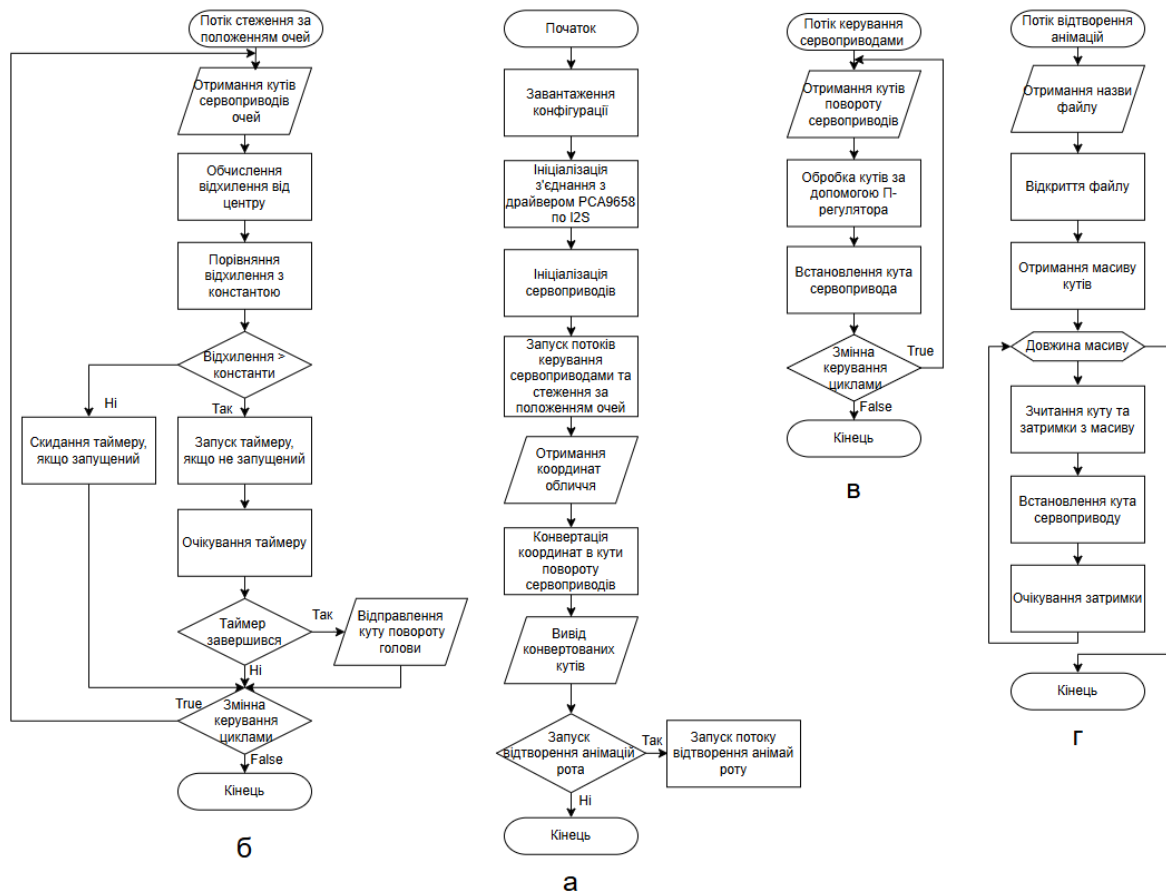
Модуль керує кількома сервоприводами, які відповідають за рух очей, рух голови та рух рота робота. Спочатку модуль завантажує конфігураційний файл, у якому визначені параметри підключення сервоприводів, їхні канали, межі кута обертання, інверсії осей і характеристики руху. Після цього ініціалізується апаратний інтерфейс I2C для взаємодії з платою PCA9658, яка управляє подачею імпульсів на серводвигуни. Для кожного сервоприводу створюється об'єкт із заданими параметрами пульсу. Алгоритм першочергової ініціалізації модуля зображено на рисунку 2.5, а.

Щоб забезпечити плавний рух сервоприводів, у модулі запускаються окремі паралельні потоки. Один із них – це головний цикл оновлення позицій сервоприводів очей та голови (рисунок 2.5, в). Він постійно порівнює поточні кути положення з цільовими, поступово змінюючи їх з певною швидкістю, визначеною в конфігураційному файлі, і встановлює нові кути для кожного сервоприводу. Це дозволяє уникнути різких рухів і забезпечує природність анімацій.

Потік стеження за положенням очей, алгоритм якого зображено на рисунку 2.5, б, контролює відхилення очей від центру і, якщо воно перевищує певний поріг протягом заданого часу, відповідно коригує положення голови. Ця логіка дає змогу роботу “пристосувати” положення голови в залежності від руху очей.

Модуль надає функцію асинхронного відтворення анімацій руху рота (рисунок 2.5, г), завантажуючи кути рухів із JSON-файлів. Кожен кадр анімації задає кут сервоприводу рота, а між кадрами робиться затримка відповідно до заданої тривалості.

Для керування кутами руху передбачені функції, які перетворюють координати позиції обличчя в кути обертання сервоприводів.



а – алгоритм першочергової ініціалізації модуля,

б – алгоритм потоку стеження за положенням очей,

в – алгоритм потоку керування сервоприводами,

г – алгоритм потоку відтворення анімацій

Рисунок 2.5 – Алгоритм роботи модуля керування сервоприводами

2.5 Модуль розпізнавання мовлення VOSK

Алгоритм роботи модулю розпізнавання мовлення VOSK [6] наведено на рисунку 2.6.

Модуль призначений для перетворення звукових даних у текст. На вхід вона отримує звук у вигляді масиву чисел та частоту, з якою цей звук був записаний.

Спочатку дані перетворюються у формат, який можна зберегти як звуковий файл (WAV). Потім цей звук тимчасово зберігається в оперативній пам'яті, без запису на диск.

Далі створюється розпізнавач мовлення, який використовує попередньо завантажену україномовну модель Vosk (nano). Розпізнавач поступово “прослуховує” звук по частинах, поки не дійде до кінця. Після чого він повертає результат у форматі тексту. Отриманий текст і є розпізнаною мовою з аудіо.

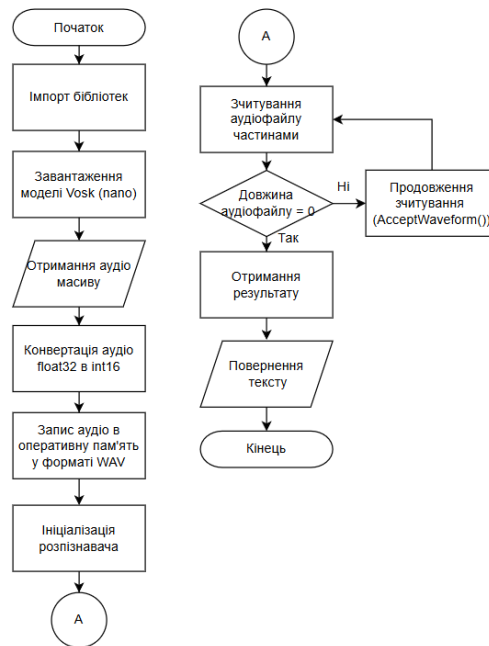


Рисунок 2.6 – Алгоритм роботи модулю розпізнавання мовлення VOSK

2.6 Модуль розпізнавання мовлення Google STT

Алгоритм роботи модуля розпізнавання мовлення Google STT зображено на рисунку 2.7.

Модуль виконує розпізнавання мовлення за допомогою хмарного сервісу від Google. Спочатку аудіо подається у вигляді масиву чисел із плаваючою комою (float) від -1 до 1. Ці значення перетворюються в 16-бітове цілочисельне значення (int16), оскільки Google STT очікує аудіо у форматі WAV з лінійним кодуванням. Після цього у пам'яті створюється WAV файл. Далі цей файл передається в Google API разом з параметрами частоти дискретизації та українським мовним кодом. API обробляє запит та повертає результат розпізнавання у вигляді тексту.

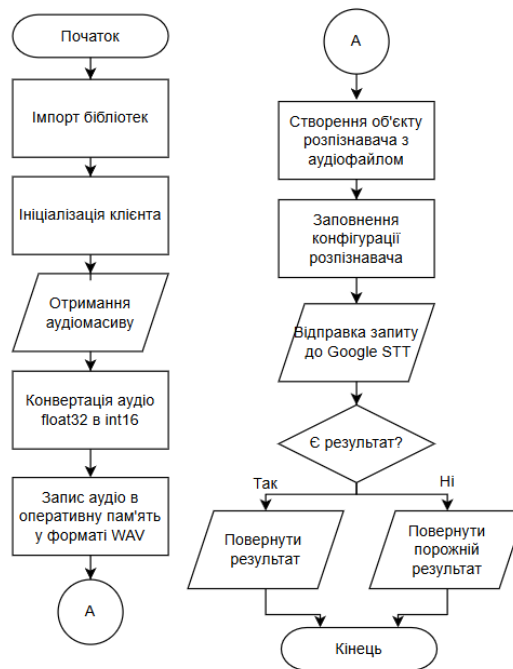


Рисунок 2.7 – Алгоритм роботи модуля розпізнавання мовлення Google STT

2.7 Скрипт створення анімацій мовлення робота

Блок схема алгоритму роботи скрипту створення анімацій мовлення робота на основі голосових відповідей наведено на рисунку 2.8.

Скрипт призначений для автоматичного генерування анімацій рота робота на основі звукового сигналу, що міститься у MP3-файлі голосової відповіді. Основна ідея полягає в тому, щоб перетворити амплітуду звуку у кути повороту сервоприводу, що імітує відкривання та закривання рота в ритм мовлення. Спочатку скрипт завантажує бібліотеки для роботи з аудіо, математичними операціями, json файлами та os для файлових операцій.

У скрипті реалізовано функцію rms, яка обчислює середньоквадратичне значення (Root Mean Square, українською середньоквадратичне значення) для аудіофрейму. Це значення є показником гучності фрагменту звуку і саме воно використовується для визначення відповідного положення сервоприводу.

Функція `audio_to_servo_animation` відкриває зазначений MP3-файл, приводить його до монофонічного звучання та стандартної частоти дискретизації для подальшої обробки. Потім аудіо розбивається на короткі

відрізки фіксованої тривалості по 50 мілісекунд. Для кожного такого фрейму розраховується значення rms, яке відповідає амплітуді в цьому часовому проміжку. Усі значення rms записуються у список, паралельно зберігається найбільше з них для подальшого масштабування.

Після збору значень гучності скрипт перетворює їх у кути повороти сервоприводу в діапазоні від 0 до 180 градусів. Для цього використовується функція інтерполяції, яка забезпечує пропорційне перетворення гучності у відповідний кут. Отримані кути зберігаються у вигляді списку у структурі даних, яка включає тривалість кожного кадру. Ця структура форматується у формат JSON та зберігається поруч із оригінальним MP3-файлом, отримуючи таку ж назву з додаванням суфікса `_mouth.json`. У разі, якщо відповідний JSON вже існує, скрипт не генеруватиме його повторно.

В основному блоці програми перевіряється наявність папки `Mouth_anim`, у якій очікуються MP3-файли. Якщо така папка є, скрипт перебирає всі MP3-файли, що в ній містяться. Для кожного з них, якщо JSON-файл ще не створений, запускається процес обробки звуку для генерації анімації. Якщо файл вже існує, обробка пропускається.

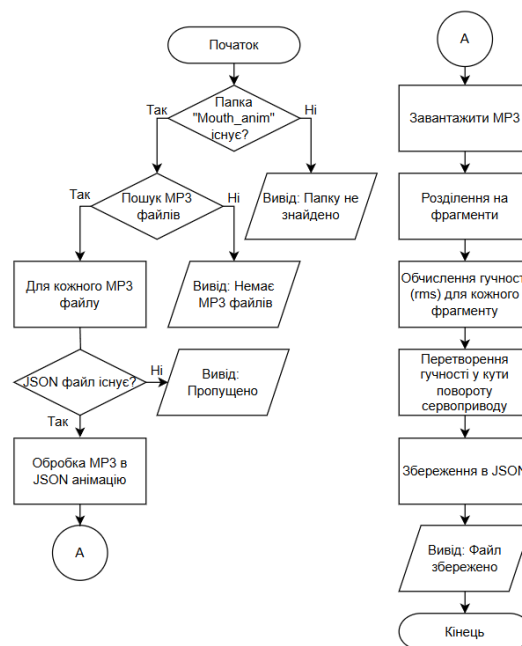


Рисунок 2.8 – Блок схема алгоритму роботи скрипту створення анімацій мовлення робота на основі голосових відповідей

2.8 Модуль виявлення мовленнєвої активності

Блок схема алгоритму роботи модуля виявлення мовленнєвої активності зображено на рисунку 2.9.

Модуль призначений для виявлення мовлення з сигналу мікрофона. Його основне завдання – розпізнати в який момент людина почне говорити та зберегти цей фрагмент для подальшої передачі до розпізнавання. Це дозволяє зменшити навантаження на модулі розпізнавання та на сам контролер.

На початку задаються параметри: частота дискретизації, тривалість одного аудіофрейму, рівень посилення мікрофону, чутливість детектора мовлення, та мінімальна довжина голосового фрагменту. Мікрофон підключається через аудіопотік, і кожен фрейм звуку передається в чергу на обробку.

Для спрацювання алгоритму VAD потрібен невеликий час, що означає що можна втратити частину голосу.

Щоб не втрачати ці перші мілісекунди мовлення, використовується буфер, який зберігає трохи звуку до початку голосу. Коли надходить новий фрейм, його спочатку підсилюють і перетворюють у формат, придатний для аналізу. Потім запускається алгоритм VAD, який перевіряє, чи є в цьому фрагменті мовлення.

Якщо мовлення виявлено, починається запис. Якщо в кількох кадрах підряд мовлення вже немає – запис зупиняється. Коли фрагмент завершено, перевіряється його довжина. Якщо фрагмент надто короткий, він не зберігається і не використовується. Якщо запис нормальний, він передається на розпізнавання. Для розпізнавання можна використовувати або локальний розпізнавач Vosk, або онлайн сервіс Google STT. Після розпізнавання тексту програма перевіряє, чи містить він одну з команд зі списку. Якщо така команда існує – виконується відповідна дія.

Для запобігання потрапляння голосової відповіді до аудіопотоку мікрофону, використовується вбудований контакт DFPlayer.



Рисунок 4.7 – Алгоритм роботи модуля виявлення мовленнєвої активності

3 СКЛАДАННЯ АПАРАТНОЇ ЧАСТИНИ РОБОТА-ПРОМОУТЕРА

3.1 Вибір апаратних модулів

Створення робота-промоутера – це комплексне завдання, яке включає апаратну та програмну частину, і вимагає балансування між обчислювальною потужністю, енергоспоживанням та функціональністю.

Для забезпечення коректної роботи робота-промоутера необхідно обрати контролер, який відповідатиме вимогам обчислювальної потужності, кількості периферійних інтерфейсів, підтримки підключення зовнішніх пристроїв, енергоефективності та компактності. Контролер має забезпечувати стабільну роботу розпізнавання голосу, обробку зображення, керування сервоприводами та взаємодію з іншими модулями системи.

Проведено аналіз можливих варіантів мікроконтролерів, що можуть бути використані в якості основного керуючого пристрою.

Порівняння мікроконтролерів для робота-промоутера наведено в таблиці 3.1.

Таблиця 3.1 – Порівняння контролерів

Параметр	Raspberry Pi Zero 2 W	Raspberry Pi 4 Model B	NVIDIA Jetson Nano	Banana Pi M2 Zero
Процесор	4x ARM Cortex-A53, 1GHz	4x Cortex-A72, до 1.5GHz	4x Cortex-A57, 1.43GHz	4x Cortex-A7, 1GHz
Оперативна пам'ять	512 МБ LPDDR2	2-8 ГБ LPDDR4	4 ГБ LPDDR4	512 МБ DDR3
ОС/ПЗ	Linux (Raspberry Pi OS)	Linux (Raspberry Pi OS)	Ubuntu-based	Linux
Камера	CSI (офіційна RPі камера)	CSI, USB	CSI	CSI

Продовження таблиці 3.1

Параметр	Raspberry Pi Zero 2 W	Raspberry Pi 4 Model B	NVIDIA Jetson Nano	Banana Pi M2 Zero
Машинний зір	Так, OpenCV можливо	Так, швидше	Так, має GPU (CUDA)	Дуже обмежено
Розпізнавання мовлення (Vosk)	Так, працює	Так, швидше	Так, з високою продуктивністю	Повільно
Живлення	5 В, ~0.5 А	5 В, 2-3 А	5 В, 2-4 А	5 В, ~0.5 А
Ціна (станом на 2025)	~\$20-30	\$50-100	\$70-130	\$20-30
Wi-Fi / Bluetooth	Wi-Fi + Bluetooth 4.2	Wi-Fi + Bluetooth 5.0	Немає Wi-Fi (треба USB)	Wi-Fi (без Bluetooth)
Периферія	I2C, UART, GPIO	I2C, UART, GPIO, USB 3.0	I2C, USB, GPIO	I2C, UART, GPIO

Raspberry Pi Zero 2 W (рис. 3.1) – це хороший вибір для компактного робота-промоутера. Його основна перевага полягає у хорошому балансі між розмірами, енергоспоживанням, функціональністю та ціною. При мініатюрному форм-факторі ця плата підтримує всі необхідні інтерфейси (GPIO, I2C, UART), що дозволяє підключати мікрофони, звукові модулі, камери та керувати сервомоторами. При цьому вона достатньо потужна щоб одночасно виконувати просте розпізнавання голосу та паралельно обробляти відеопотік з камери для реалізації стеження за обличчям. Цього рівня продуктивності достатньо для створення інтерактивної взаємодії, а енергоефективність дозволяє працювати навіть від акумулятора.



Рисунок 3.1 – Raspberry Pi Zero 2 W з модулем Pi Camera rev 1.3

При виборі мікрофона для робота, який має розпізнавати голосові команди, найважливішими критеріями є: якість звуку, надійність роботи в реальному часі, простота інтеграції з мікроконтролером, енергоефективність та розміри. У нашому випадку робот працює на базі Raspberry Pi Zero 2 W, тож мікрофон має відповідати вимогам цієї платформи.

Порівняння мікрофонів наведено в таблиці 3.2.

Таблиця 3.2 – Порівняння мікрофонів для розпізнавання голосових команд

Мікрофон	Тип	Інтерфейс	Вбудований ADC	Підключення	Якість запису	Ціна (грн)
INMP441	MEMS	I2S	Так	Пряме через I2S	Висока	80-160
MAX9814	Analog	Analog OUT	Ні	Через зовнішній ADC	Середня	120-160
MAX4466	Analog	Analog OUT	Ні	Через зовнішній ADC	Середня	80-120
USB-мікрофон	discrete	USB	Так	Через USB	Висока	200-600

Мікрофони діляться на аналогові та цифрові. Аналогові мікрофони, як MAX9814 або MAX4466, хоча й доступні за ціною, потребують зовнішнього аналогово-цифрового перетворювача або звукової карти для оцифрування сигналу. Це ускладнює схему, створює затримки, додає шумів, а також вимагає калібрування посилення. У результаті – знижується точність розпізнавання команд, особливо в умовах фонових шумів.

Інший варіант – USB-мікрофони, які мають вбудований аналогово-цифровий перетворювач і працюють напряму з Raspberry Pi. Але тут з'являється інша проблема: Raspberry Pi Zero 2W має лише один вільний USB-порт, який зазвичай вже зайнятий іншим обладнанням. До того ж, USB-мікрофони можуть створювати затримку при передаванні даних, що шкодить швидкості реакції робота.

У цій ситуації цифрові мікрофони з I2S-інтерфейсом, є хорошим вибором. INMP441 (рис. 3.2) має вбудований аналогово-цифровий перетворювач, тому одразу видає цифровий сигнал по I2S. Цей інтерфейс підтримується Raspberry Pi, не займає USB, дозволяє зчитувати звук з мінімальною затримкою, та забезпечує стабільну якість у будь-яких умовах.

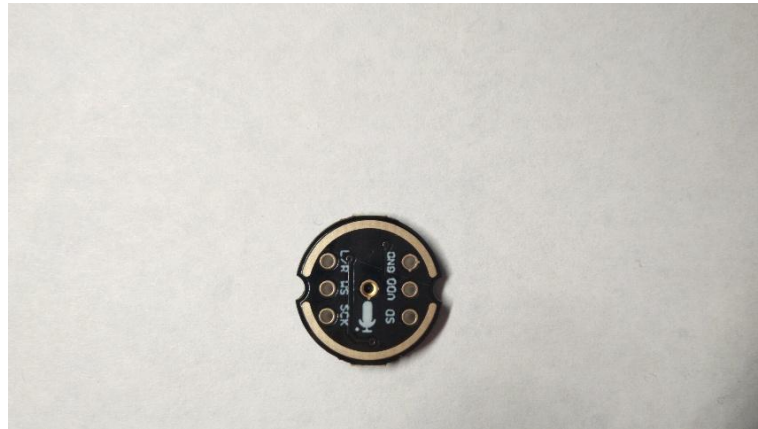


Рисунок 3.2 – MEMS мікрофон INMP441

Однією з головних переваг INMP441 є невелика ціна, яка коливається в межах 120 гривень. У порівнянні з аналоговими мікрофонами, які потребують додаткових модулів на кшталт MCP3008 або USB-звукової карти, це не лише дешевше, а й менш громіздке рішення.

Камера є одним з ключових сенсорів робота-промоутера, адже вона дозволяє роботу “бачити” людину, реагувати на її присутність, позицію обличчя. Тому при виборі камери важливо врахувати сумісність із Raspberry Pi Zero 2 W, якість зображення, частоту кадрів, габарити, а також ресурсоемність обробки відео.

Для Raspberry Pi доступні як USB-камери, так і офіційні CSI-камери, що підключаються через спеціальний роз’єм (Camera serial interface). З точки зору розпізнавання обличчя у реальному часі, потрібна камера, яка забезпечує стабільне відео мінімум у 15 кадрів за секунду, має роздільну здатність щонайменше 640x480, і при цьому не перевантажує процесор чи шину передачі даних.

USB-камери на Raspberry Pi Zero 2 W зазвичай підключаються через OTG-перехідник і споживають чимало ресурсу з USB-порту. Це створює конфлікт ресурсів із іншими пристроями. До того ж обробка відео з USB-камери навантажує CPU більше, ніж камера CSI, оскільки використовується загальна USB-шина, а не спеціалізований відеоінтерфейс. Тому USB варіант не є оптимальним, особливо якщо важлива швидка реакція на обличчя користувача.

Натомість офіційна 5-ти мегапіксельна камера для Raspberry Pi на чіпі OV5647 (рис. 3.1), є значно кращим варіантом. Вона підключається через спеціальний порт на платі Raspberry Pi, працює без навантаження на USB-шину, і підтримується напряму ядром Linux. Що дозволяє захоплювати відео на апаратному рівні, без зайвих драйверів, та виводити його напряму в OpenCV з мінімальною затримкою.

Для відтворення відповіді на голосові команди у нашому випадку важливо врахувати обмежені ресурси Raspberry Pi Zero 2 W, одночасну роботу з камерою, мікрофоном та сервоприводами, тому аудіопідсистема має бути максимально простою та незалежною від процесора.

Є два основні підходи для відтворення голосових повідомлень:

- вбудований синтез мовлення через програмне забезпечення (TTS);
- відтворення заздалегіть записаних аудіофайлів (MP3/WAV).

Програмний синтез мовлення через спеціалізовані бібліотеки або хмарні сервіси потребує постійного навантаження на процесор і використовує аудіоінтерфейс Raspberry Pi. У випадку Raspberry Pi Zero 2 W це означає помітну затримку, “роботизовану” якість звуку, а також конфлікти з іншими пристроями, які працюють паралельно (особливо при використанні I2S мікрофону або виводу зображення з камери). Через це синтез мовлення хоч і зручний, але не підходить для стабільної роботи в реальному часі на слабкому процесорі.

Натомість відтворення готових аудіофайлів, записаних заздалегіть, є значно ефективнішим. Воно дозволяє контролювати якість голосу, інтонацію,

додавати паузи, музику або ефекти – і все це без навантаження на процесор. Для цього існують спеціальні апаратні модулі, такі як DFPlayer Mini, які здатні відтворювати MP3-файли напряму з microSD-карти, без участі основного процесора. Керування модулем здійснюється по UART (послідовному порту) або простими командами через GPIO, і він має власний аудіовихід для підключення динаміка або підсилювача.

Модуль DFPlayer Mini (рис. 3.3) підтримує нумерацію треків, регулювання гучності, циклічне та одноразове відтворення, і дозволяє запускати репліки миттєво після розпізнавання команди. Це дає змогу повністю синхронізувати рухи рота та звукову відповідь. Ціна такого модуля приблизно 100 гривень.

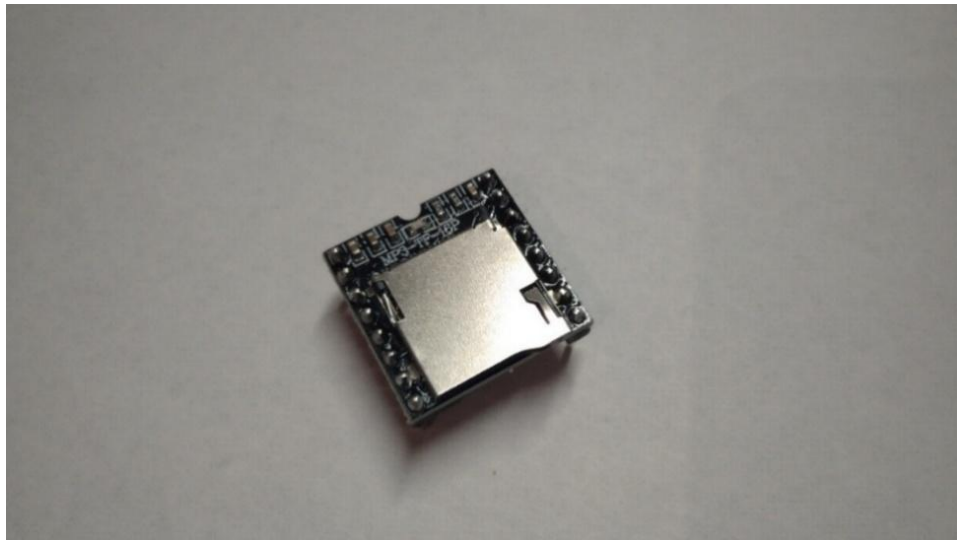


Рисунок 3.3 – Модуль відтворення звуку DFPlayer Mini

Функції руху робота реалізуються через малопотужні сервоприводи, що дозволяють точно контролювати кут обертання. Вибір способу керування цими сервоприводами напряму впливає на плавність рухів, надійність роботи та сумісність із іншими компонентами.

У системі використовується кілька сервоприводів: два для очей (погляд по вертикалі та по горизонталі), два для повороту голови (вертикаль та горизонталь), та один для відкриття рота.

Стандартні сервоприводи (типу SG90, MG90S або аналогічні) керуються широтно-імпульсною модуляцією (PWM) – вони реагують на тривалість імпульсу, що повторюється з частотою 50 Гц (20мс період), де сигнал у межах 0.5-2.5 мс відповідає різним кутам.

Теоретично Raspberry Pi може самостійно генерувати PWM сигнали через свої GPIO піни за допомогою спеціалізованих бібліотек. Однак на практиці програмно-генеровані PWM-сигнали нестабільні, особливо коли паралельно виконується обробка відео, голосу, I2C-пристроїв, тощо. Результат – тремтіння, вібрації, збої в положенні та непередбачувані рухи сервоприводів.

Тому доцільніше винести керування сервоприводами на зовнішній апаратний контролер. У нашому випадку був обраний 16-канальний 12-бітний PWM модуль PCA9685 (рис. 3.4), який спілкується з Raspberry Pi по I2C. Цей чіп забезпечує високу точність (до 4096 рівнів), стабільність, незалежність від навантаження системи та дозволяє керувати до 16 сервоприводами одночасно з ідеальною синхронізацією, без ривків і затримок.

До переваг PCA9685 можна додати підтримку популярних бібліотек на Python (Adafruit_PCA9685), які дозволяють значно спростити керування сервоприводами та працюють без додаткових драйверів.

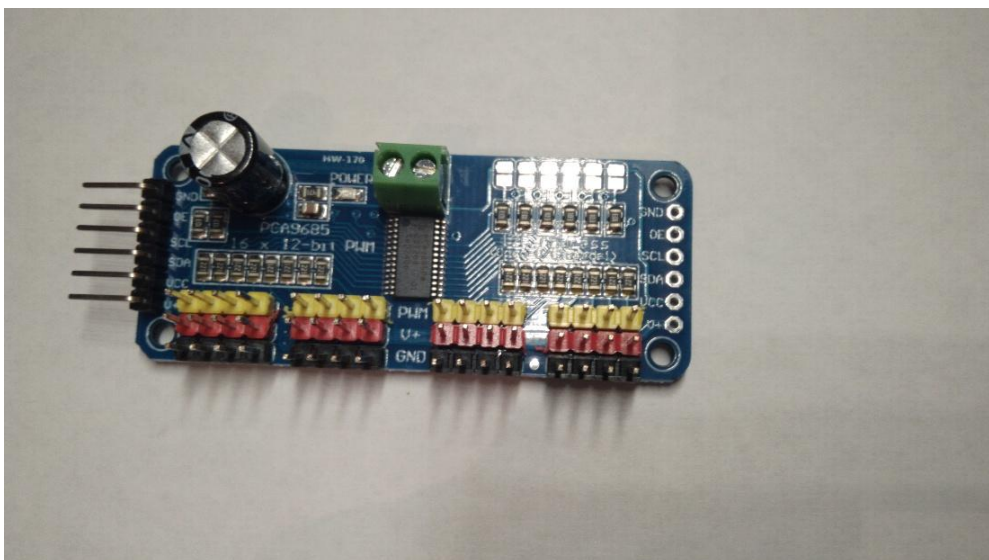


Рисунок 3.4 – Модуль керування сервоприводами PCA9685

Використання окремого PWM модуля дозволяє масштабувати систему: за потреби можна додати ще 11 сервоприводів, не змінюючи логіку керування. PCA9685 має окремі лагцюги живлення логічної частини та силової, що захищає контролер від перевантаження.

Модуль коштує приблизно 160 гривень, легко монтується, має мінімум пайки, і спрощує програмну частину.

Усі апаратні модулі були замовлені в інтернет магазині україни [8].

3.2 Створення електричної схеми підключення

Для створення схеми підключення макету робота промютера, яка розташована на рисунку 3.5, було використано застосунок EasyEDA [9].

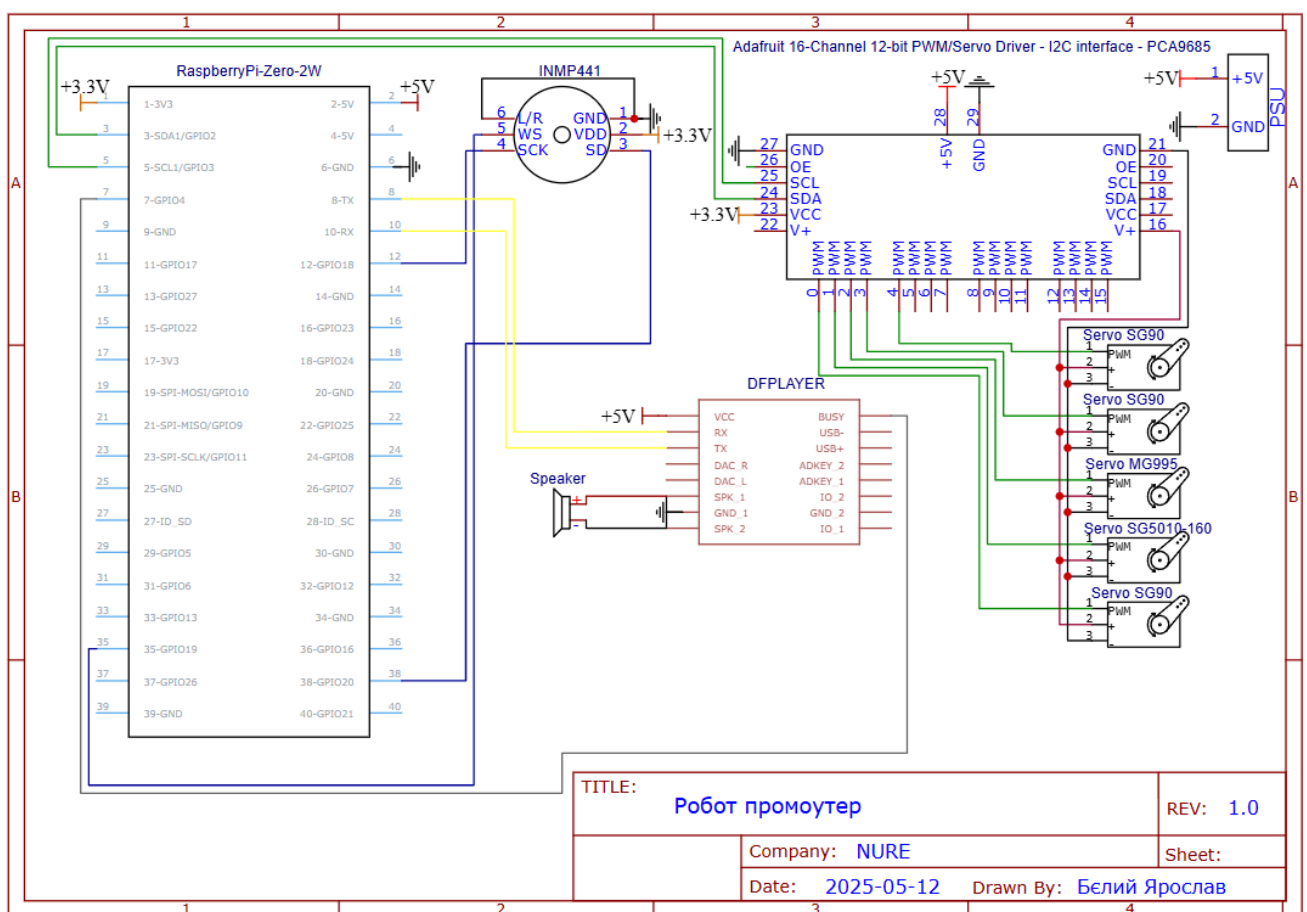


Рисунок 3.5 – Схема підключення макету робота промютера

Дана електрична схема реалізує систему керування роботом-промоутером, основою якого є мікроконтролер Raspberry Pi Zero 2 W. Система призначена для розпізнавання обличь з камери, голосових команд, відтворення звуків та керування кількома сервоприводами для імітації руху голови, очей та роту.

Мікроконтролер Raspberry Pi Zero 2 W відповідає за обробку сигналів від мікрофону, передавання команд на відтворення голосових відповідей плеєру, отримання та обробку зображення з камери та генерацію керуючих сигналів для сервоприводів. Мікроконтролер Raspberry Pi з'єднується з іншими модулями системи через стандартні інтерфейси GPIO, UART, I2C та I2S.

Для зчитування голосових команд використовується цифровий MEMS мікрофон INMP441, який передає аудіосигнал у цифровому форматі через інтерфейс I2S. Мікрофон підключений до GPIO виводів Raspberry Pi, які підтримують I2S: WS (Word Select) з'єднаний з GPIO19, SCK (Serial Clock) з GPIO18, а SD (Serial Data) з GPIO20. Живлення на мікрофон подається з лінії +3.3 V та GND.

Для розпізнавання обличь використовується офіційна камера Raspberry Pi, яка підключається безпосередньо до роз'єму CSI (Camera Serial Interface) на платі Raspberry Pi Zero 2 W. Камера передає відеопотік у реальному часі, який обробляється на борту для виявлення положення обличчя. Ця інформація використовується для орієнтації голови та очей робота у напрямку співрозмовника, забезпечуючи ефект зорового контакту.

Для відтворення звукових повідомлень використовується модуль DFPlayer Mini, який є автономним MP3-плеєром. Він взаємодіє з Raspberry Pi по UART інтерфейсу. DFPlayer має підключений динамік через виходи SPK1 і SPK2, що дозволяє відтворювати звуки безпосередньо без зовнішніх підсилювачів. Живлення на модуль подається від лінії +5 V, при цьому важливо дотримуватись спільної землі між усіма модулями.

Керування сервомоторами реалізовано за допомогою драйвера PCA9685, що підтримує до 16 незалежних каналів PWM. Він підключений до Raspberry Pi через інтерфейс I2C: лінія SDA з'єднана з GPIO2, а SCL – з GPIO. Драйвер має окремі лінії живлення: логічна працює від VCC 3.3 V, а силова лінія для сервоприводів заживлюється через вивід V+ з зовнішнього джерела на 5 V. Всі лінії землі (GND) з'єднані разом, щоб забезпечити коректну роботу. Для стабілізації живлення сервоприводів на платі встановлено конденсатор на 1000 мкФ. При нестачі можна додати ще ємності паралельно живленню драйвера або окремого сервоприводу.

До виходів PCA9685 підключено п'ять сервоприводів різного типу. Використано три сервоприводи типу SG90 для дрібних рухів (очей та рота), один, потужніший сервопривод, MG995 для повороту голови по горизонталі, а також один SG5010-160, який забезпечує поворот голови по вертикалі.

Кожен сервопривід підключається до окремого PWM-виходу драйвера через три лінії: керуючу (PWM), живлення (+5V) та землю (GND).

Живлення Raspberry Pi здійснюється через USB, або через спеціальні лінії на роз'ємі GPIO. При цьому важливо не підключати безпосередньо сервомотори до живлення з Raspberry Pi, оскільки це може спричинити перевантаження та перезавантаження системи. Сервомотори повинні живитись від окремого блоку живлення, розрахованого на струм не менше 2-5 А залежно від навантаження. При цьому всі “землі” (GND) повинні бути поєднані між собою, щоб уникнути перепадів потенціалу.

Дана схема дозволяє реалізувати інтерактивну систему, де робот може реагувати на голосові команди користувача, озвучувати заздалегідь записані фрази або звуки, розпізнавати присутність та положення обличчя співрозмовника й повертати голові або очі відповідно до його розташування.

Готовий макет зображено на рисунку 3.6.

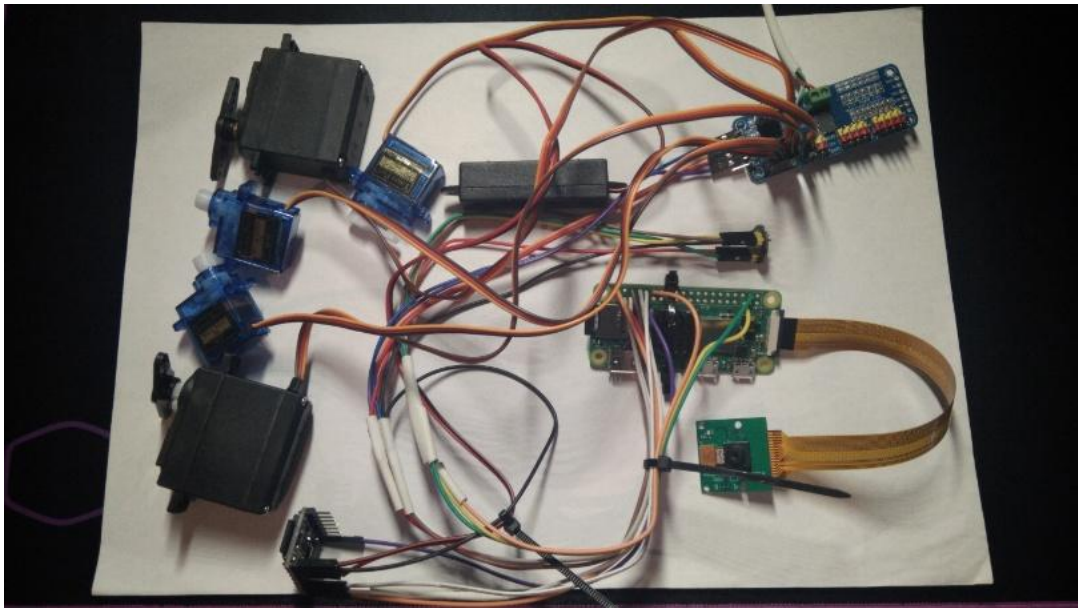


Рисунок 3.6 – Зібраний макет робота-промоутера

Збірка макету включає етап пайки кабелів та компонентів. Перелік використаного обладнання: паяльник 60 W ZD-735 з маленьким жалом; флюс-гель FVS-BGA-NC (150–300°C); припій (Sn60Pb40); гострогубці; термоусадка.

Додатково було встановлено радіатор на процесор Raspberry Pi для додаткового захисту від перегріву. Raspberry Pi не має кріплення для радіатора, тому було вирішено закріпити на дві пластикові стяжки. Між радіатором та процесором нанесено шар спеціальної термопровідної пасти для кращого контакту. Усі з'єднання, які не використовують роз'єми, були пропаяні та захищені шаром термоусадкової трубки для запобігання коротким замиканням та механічним пошкодженням.

Для прототипування робота-промоутера було виготовлено функціональний макет із картону, який зображено на рисунку 3.7. Модуль має габарити приблизно 25x15x20 см та слугує фізичною основою для розміщення електронних компонентів і тестування динамічної поведінки.

Конструкція складається з декількох картонних модулів, з'єднаних клеєм, скотчем та пластиковими стяжками. Всі активні вузли (камера, мікрофон, сервоприводи, електроніка) закріплені ззовні моделі для спрощення доступу під час розробки.

Механічна частина включає в себе п'ять сервоприводів:

- два сервоприводи для керування очима, розташовані безпосередньо біля очей приблизно посередині голови робота;
- один сервопривід для руху рота, встановлений безпосередньо під ним;
- два сервоприводи для повороту голови по вертикалі та по горизонталі, розміщені в середині розбірної конструкції.

На верхній частині голови встановлено камеру для захоплення зображення. Мікрофон тимчасово закріплено на передній частині голови поблизу рота. Звуковий модуль розташовано ззовні корпусу, як і решта електронних компонентів.

Слабкі місця конструкції посилено додатковим шаром товстого картону.

База, на якій розміщено голову, зроблена з товстого картону, додатково посилена дерев'яними елементами, на які кріпиться один із сервоприводів за допомогою комплектних гвинтів.

Макет є умовно розбірним – більшість з'єднань виконано фіксовано, однак розбірка можлива для подальшого демонтажу елементів.

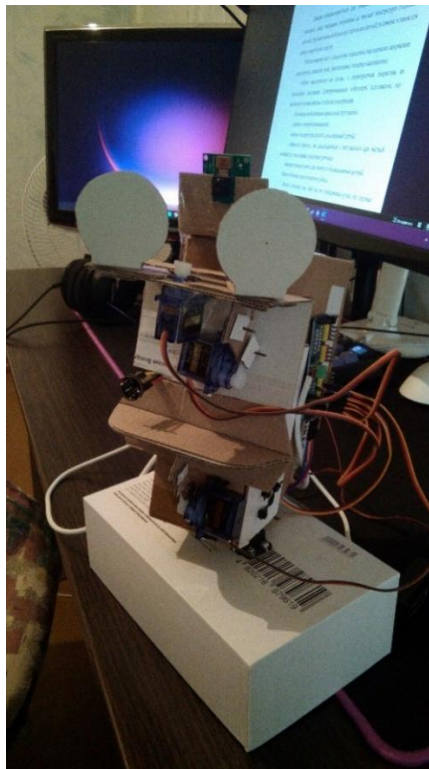


Рисунок 3.7 – Функціональний макет робота промоутера

3.3 Інструкція з охорони праці при лудінні і паянні

Паяння використовується для утворення нероз'ємного з'єднання у випадках, якщо небажане нагрівання до високої температури з'єднаних деталей. Лудіння застосовується при підготовці деталей до паяння, а також для захисту виробів від корозії.

Робота виконується з допомогою паяльників періодичного нагрівання, електричних, паяльних ламп, бензиновими, газовими паяльниками.

Роботи виконуються на столах з діелектричним покриттям на спеціальних підставках. Електропаяльники забезпечені підставками, що запобігають пошкодженню столів від перегрівання.

Основними небезпечними факторами при паянні є:

- робота з електропаяльниками;
- висока температура деталей, розплавлений припій;
- наявність флюсів, які розкладаються з виділенням шкідливих речовин;
- використання речовин для хімічного обезжирювання деталей.

Вимоги безпеки перед початком роботи:

– надіти спецодяг так, щоб він не ускладнював рухів, всі гудзики повинні бути застібнуті.

– уважно оглянути робоче місце і підготувати його до роботи: прибрати всі зайві предмети, інструменти, матеріали розмістити у зручному порядку, інструктивні та інструктивно-технологічні документи розмістити на підставках, перевірити справність витяжної вентиляції.

Вимоги безпеки під час роботи:

– підтримувати на робочому місці чистоту і порядок, працювати тільки при включеній вентиляції;

– при виконанні робіт бути уважним, не відволікатись сторонніми справами і розмовами;

– очищення паяльників від залишків припою виконувати серветкою, не допускати струшування.

- паяльник повинен лежати на підставці;
- під ногами повинен бути діелектричний килимок;
- вимкати паяльник з розетки потрібно за корпус вилки;
- у електричного паяльника ручка повинна бути сухою.

Вимоги безпеки після закінчення роботи:

- відключити паяльник, вимкнути рубильник;
- привести в порядок робоче місце;
- зняти спецодяг і покласти у відповідне для нього місце;

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА НАЛАШТУВАННЯ СЕРЕДОВИЩА ВИКОНАННЯ

4.1 Налаштування середовища виконання

Для налаштування та програмування контролеру використана література [11, 12, 13].

Для керування роботом-промоутером обрано мікроконтролер Raspberry Pi Zero 2 W із встановленою операційною системою Raspberry Pi OS Lite. Ця система забезпечує мінімальне навантаження на ресурси та роботу виключно в середовищі командного рядка. Образ системи записувався на microSD-карту об'ємом 16 ГБ за допомогою утиліти Raspberry Pi Imager. Вікно утиліти Raspberry Pi Imager зображено на рисунку 4.1.

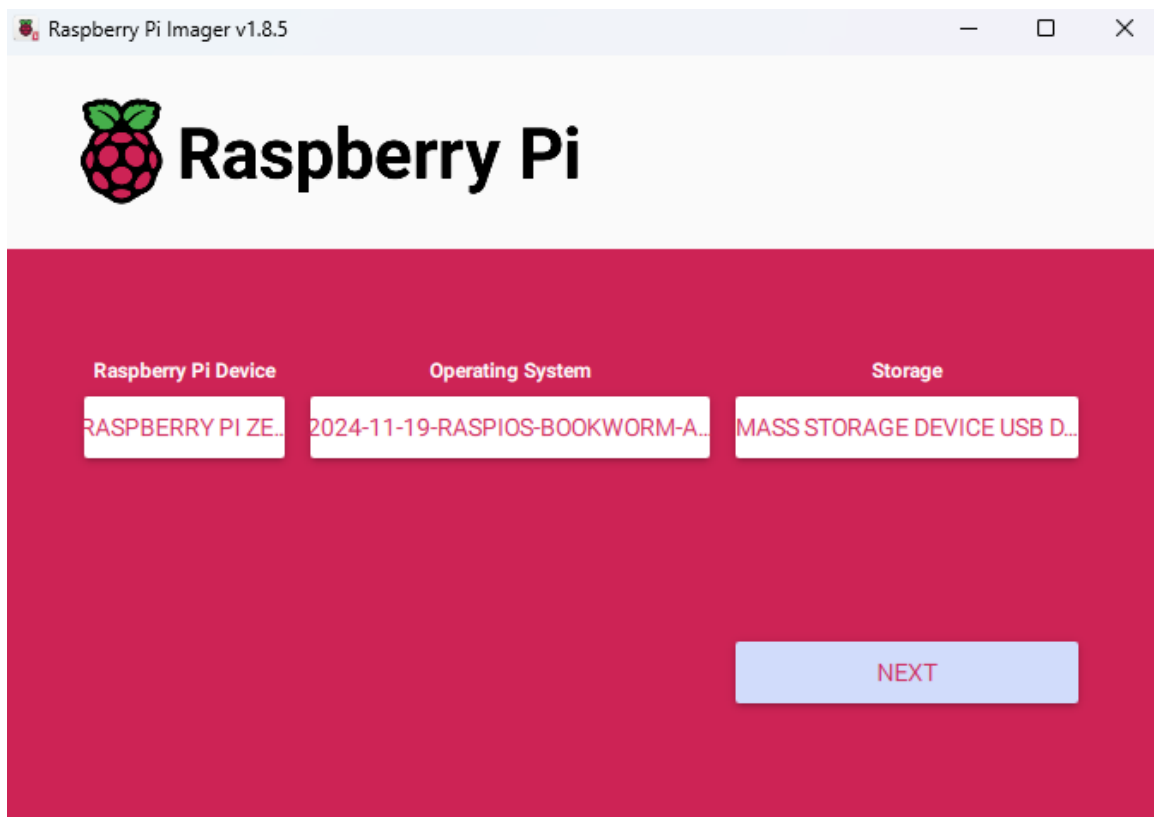


Рисунок 4.1 – Утиліта для встановлення операційної системи Raspberry Pi Imager

Перед записом операційної системи в налаштуваннях було активовано SSH-підключення, аби відразу після першого завантаження можна було заходити на плату по мережі без необхідності підключення монітора та клавіатури. Додатково в “Advance options” Pi Imager було вказано параметри Wi-Fi (SSID, пароль, регіон), що дозволить одразу вийти в інтернет для оновлення та встановлення додаткових пакетів. Вікно налаштувань Raspberry Pi Imager зображено на рисунку 4.2.

The image shows a screenshot of the Raspberry Pi Imager's 'OS Customisation' window. The window has three tabs: 'GENERAL', 'SERVICES', and 'OPTIONS'. The 'GENERAL' tab is currently selected and highlighted with a pink underline. Below the tabs, there are several configuration options, each with a checked checkbox:

- Set hostname:** raspberrypi .local
- Set username and password:** Username: Zero2W, Password: [masked]
- Configure wireless LAN:** SSID: Tenda_355830, Password: [masked]. There are also checkboxes for 'Show password' and 'Hidden SSID', both of which are unchecked. The 'Wireless LAN country' is set to GB.
- Set locale settings:** Time zone: Europe/Kiev, Keyboard layout: US.

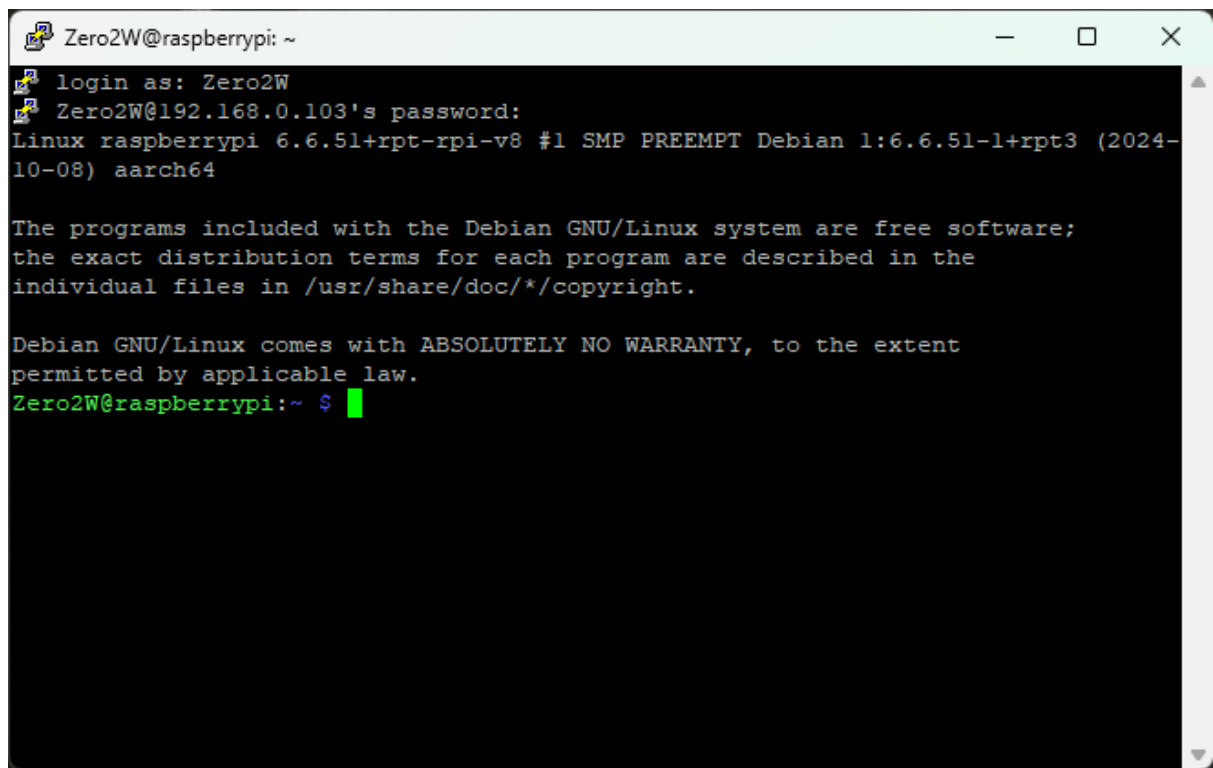
Below these settings is a blue 'SAVE' button. The 'SERVICES' tab is also visible and highlighted with a pink underline. It contains the following options:

- Enable SSH:** Checked.
- Use password authentication:** Selected with a radio button.

Рисунок 4.2 – Вікно налаштувань Raspberry Pi Imager

Для підключення до Raspberry Pi Zero 2 W через протокол SSH було використано застосунок PuTTY. PuTTY – це легкий і популярний клієнт для віддаленого доступу по протоколах SSH, Telnet, Rlogin та Serial, що працює під Windows. При запуску застосунку PuTTY відкривається вікно конфігурацій, де в полі “Host name (or IP address)” необхідно ввести IP-адресу плати, вказану при встановленні операційної системи, або отриману з роутера. Порт залишається стандартним 22, а тип з’єднання – SSH. Після натискання “Open” з’явиться чорне вікно терміналу з попередженням про невідомий ключ хоста; слід погодитись, щоб запам’ятати його в реєстрі PuTTY.

Далі з’явиться запит на логін: вводимо дані користувача “Zero2W”, створеного під час встановлення системи, а потім пароль “1111”. Після вдалого входу в консолі з’явиться рядок `Zero2W@raspberrypi:~$`, що означає готовність до роботи. Вікно активного з’єднання зображено на рисунку 4.3.



```
Zero2W@raspberrypi: ~
login as: Zero2W
Zero2W@192.168.0.103's password:
Linux raspberrypi 6.6.51+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.51-1+rpt3 (2024-10-08) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Zero2W@raspberrypi:~ $
```

Рисунок 4.3 – Вікно активного з’єднання через протокол SSH

Операційна система Raspberry OS Lite постачається з попередньо встановленим інтерпретатором Python 3, що дозволяє одразу розпочати розробку програм на Python. Втім, втручання у системне середовище Python може спричинити нестабільну роботу системи або конфлікти між пакетами. Тому рекомендується використовувати безпечні методи інсталяції залежностей – через системний менеджер пакетів apt або менеджер Python-залежностей pip у віртуальному середовищі.

Менеджер пакетів apt забезпечує встановлення попередньо скомпільованих, адаптованих під Raspberry Pi Os. Перевагами цього методу є:

- швидке встановлення завдяки готовим до використання пакетам;
- автоматичне встановлення усіх залежностей;
- безпечне видалення без пошкодження інших системних пакетів.

Встановлення бібліотек цим способом рекомендується для загальносистемного використання або коли бібліотека доступна лише у стандартних репозиторіях.

Менеджер pip дозволяє встановлювати пакети Python безпосередньо з Python Package Index (PyPI). Історично пакети можна було встановлювати безпосередньо в системну версію Python. Однак, починаючи з версії Raspberry Pi OS Bookworm, така практика заборонена – pip більше не може змінювати системне середовище Python напямую. Це пов'язано з потенційними конфліктами між apt і pip, які можуть призводити до порушення цілісності системи.

Рекомендуємий підхід – створення віртуального середовища Python за допомогою модуля venv. Таке середовище є ізольованим контейнером, де можна безпечно встановлювати сторонні пакети без впливу на системний Python.

Для створення власного віртуального середовища створимо нову папку проекту, в яке помістимо віртуальне середовище. Створення папки проекту зображено на рисунку 4.4.

```
Zero2W@raspberrypi:~ $ mkdir my_project
Zero2W@raspberrypi:~ $ ls
get-pip.py my_project
Zero2W@raspberrypi:~ $ cd my_project/
Zero2W@raspberrypi:~/my_project $
```

Рисунок 4.4 – Створення папки проекту

Далі для створення віртуального середовища скористаємося командою `python3 -venv venv --system-site-packages`, де параметр `--system-site-packages` дозволяє отримати доступ до системних бібліотек, залишаючи при цьому можливість додавати власні.

Для активації віртуального середовища скористаємося командою `source venv/bin/activate`. Після активації у командному рядку з'явиться префікс `(venv)`, що вказує на те, що зараз працює ізольоване середовище.

Після встановлення системи було одразу оновлено системну для отримання останніх виправлень безпеки та сумісності за допомогою команди `sudo apt update && sudo apt full-upgrade -y`.

4.2 Програмний засіб для налаштувань та керування роботом

Локальний сервер для налаштувань та керування роботом створений за допомогою вебфреймворку Flask, який забезпечує обробку HTTP-запитів, відображення вебсторінок, взаємодією з камерою, сервоприводами, аудіосистемою та конфігураційними файлами.

Головний код запуску серверу знаходиться у файлі `main.py`. Розроблений код програмного засобу наведено в додатку Б. При ініціалізації серверу до нього додаються два окремі модулі (блупрінти), які відповідають за налаштування та керування голосовими командами та сервоприводами. Сервер містить головну сторінку з можливістю навігації на сторінки відеопотоку, голосових команд та керуванням сервоприводами. Також головна сторінка показує поточне значення температури процесора та має дві кнопки для вимкнення та перезапуску Raspberry Pi. Зовнішній вигляд сторінки зображено на рисунку 4.5.

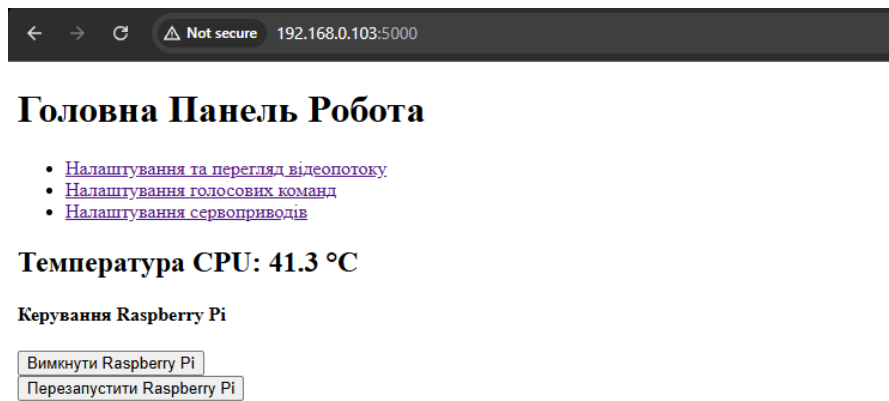


Рисунок 4.5 – Зовнішній вигляд головної сторінки

Сторінка налаштування та перегляду відеопотоку виводить оброблене зображення з класу Camera. Потік кадрів формується функцією generate, яка циклічно бере кадри з класу Camera, форматує їх у multipart формат і віддає у відповіді на запит клієнта. При завантаженні сторінки з відеопотоком клієнт отримує HTML із тегом img, що вказує на маршрут з відеопотоком. Браузер робить запит до нього і отримує послідовність JPEG кадрів, які оновлюються в реальному часі.

Сторінка налаштування голосових команд зображена на рисунку 4.6. Ця сторінка призначена для керування голосовими командами та анімаціями роту робота. Вона дозволяє завантажувати MP3-файли у спеціальну папку Mouth_anim, які потім використовуються для створення файлів анімації відкриття роту у форматі JSON. Сторінка відображає перелік вже завантажених файлів та перелік відповідних анімацій до цих файлів. В цьому списку можна видаляти файли за допомогою спеціальної кнопки. Нижче розташований повзунок регулювання гучності DFPlayer в діапазоні від 0 до 30, з можливістю збереження значення в конфігураційний файл.

Нижче розташовані перемикачі типу розпізнавання мовлення. Можна обрати між VOSK та Google STT.

Для додавання нових або редагування старих голосових команд на сторінці відображені відповідні поля.

Для створення анімацій після завантаження MP3 файлів голосових відповідей на сторінці розташована спеціальна кнопка, яка запускає скрипт створення анімацій роту на основі голосових відповідей.

Вся взаємодія реалізована через HTML-форми, які надсилають POST-запити до відповідних маршрутів Flask.

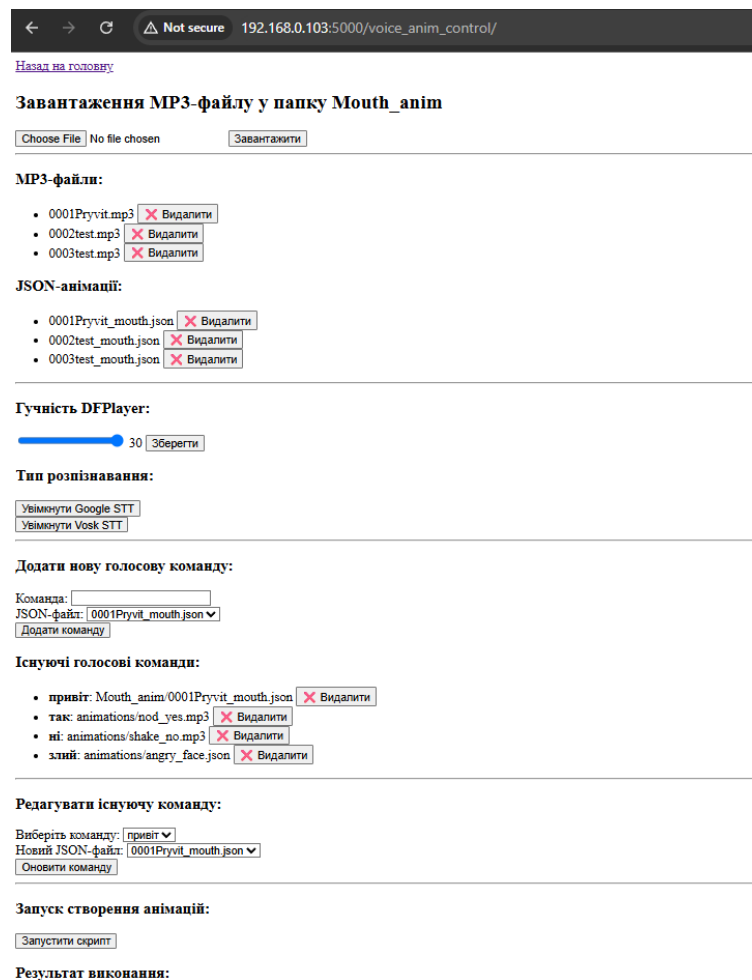


Рисунок 4.6 – Зовнішній вигляд сторінки налаштування голосових команд

Сторінка налаштування сервоприводів зображена на рисунку 4.7. Ця сторінка призначена для ручного керування сервоприводами та конфігураційним файлом. Вона дозволяє регулювати кути положення п'яти сервоприводів: горизонтальне та вертикальне положення очей та голови, та сервоприводом який відповідає за відкриття роту.

Інтерфейс містить п'ять повзунків, кожен відповідає окремому сервоприводу. Зміна положення повзунка миттєво надсилає запит на сервер з новим кутом сервоприводу. Сервер приймає цей запит, змінює кут через контролер та повертає статус виконання. Поряд з кожним повзунком відображається поточне числове значення кута у градусах.

Нижче розташовані кнопки для увімкнення або вимкнення автоматичного керування сервоприводами. При натисканні на кнопку дія виконується на сервері та зупиняє або починає роботу модуля керування сервоприводами.

Додатково на сторінці є форма для редагування конфігураційного файлу. Всі поточні значення підвантажуються з JSON файлу та виводяться як текстові поля для редагування. Після зміни та натискання кнопки збереження конфігурація зберігається та модуль керування сервоприводами перезапускається.

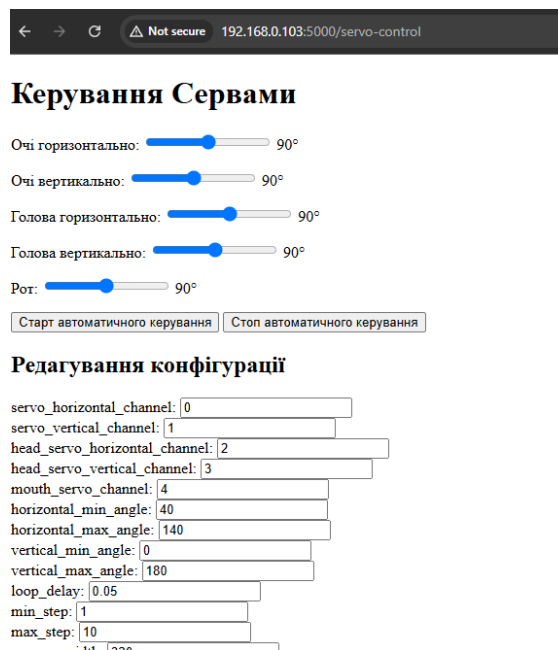


Рисунок 4.7 – Зовнішній вигляд сторінки налаштування сервоприводів

5 ДОСЛІДЖЕННЯ СТІЙКОСТІ СИСТЕМИ АВТОМАТИЧНОГО РЕГУЛЮВАННЯ

Перетворення П-регулятора у передавальну функцію.

У системі автоматичного керування положенням сервоприводу використовується пропорційний регулятор (П-регулятор), який працює у дискретному часі. Його математична модель має вигляд:

$$y(k + 1) = y(k) + K_p \cdot [u(k) - y(k)] \quad (5.1)$$

де $y(k)$ – вихід системи на k -му такті (поточне положення сервопривода);

$u(k)$ – вхідне керуюче значення (задане положення);

K_p – коефіцієнт пропорційності (посилення П-регулятора).

Розкриємо дужки:

$$y(k + 1) = (1 - K_p) \cdot y(k) + K_p \cdot u(k) \quad (5.2)$$

Застосуємо z -перетворення (припускаючи нульові початкові умови):

$$zY(z) = (1 - K_p) \cdot Y(z) + K_p \cdot U(z) \quad (5.3)$$

Виразимо передавальну функцію систему як відношення виходу до входу:

$$\frac{Y(z)}{U(z)} = \frac{K_p}{z - (1 - K_p)} \quad (5.4)$$

Передавальна функція дискретного П-регулятора має вигляд:

$$W(z) = \frac{K_p}{z - (1 - K_p)} \quad (5.5)$$

де K_p – коефіцієнт пропорційності;

$z - (1 - K_p)$ – характеризує динаміку системи та містить полюс у точці $z = 1 - K_p$.

Аналітичний розрахунок діапазону стабільних коефіцієнтів

Передавальна функція має вигляд:

$$W(z) = \frac{K_p}{z - (1 - K_p)} \quad (5.6)$$

Полюс системи:

$$z = 1 - K_p \quad (5.7)$$

Для стійкості дискретної система треба дотримуватися умови в якій усі полюси повинні знаходитись всередині одиничного кола на площині z :

$$|z| < 1 \quad (5.8)$$

тобто:

$$|1 - K_p| < 1 \quad (5.9)$$

Розкриваємо модуль:

$$-1 < 1 - K_p < 1 \quad (5.10)$$

Звідси:

$$-1 < 1 - K_p \rightarrow K_p < 2 \quad (5.11)$$

$$1 - K_p < 1 \rightarrow K_p > 0 \quad (5.12)$$

Отже, стабільний діапазон коефіцієнтів K_p :

$$0 < K_p < 2 \quad (5.13)$$

Графік стабільних діапазонів зображено на рисунку 5.1.

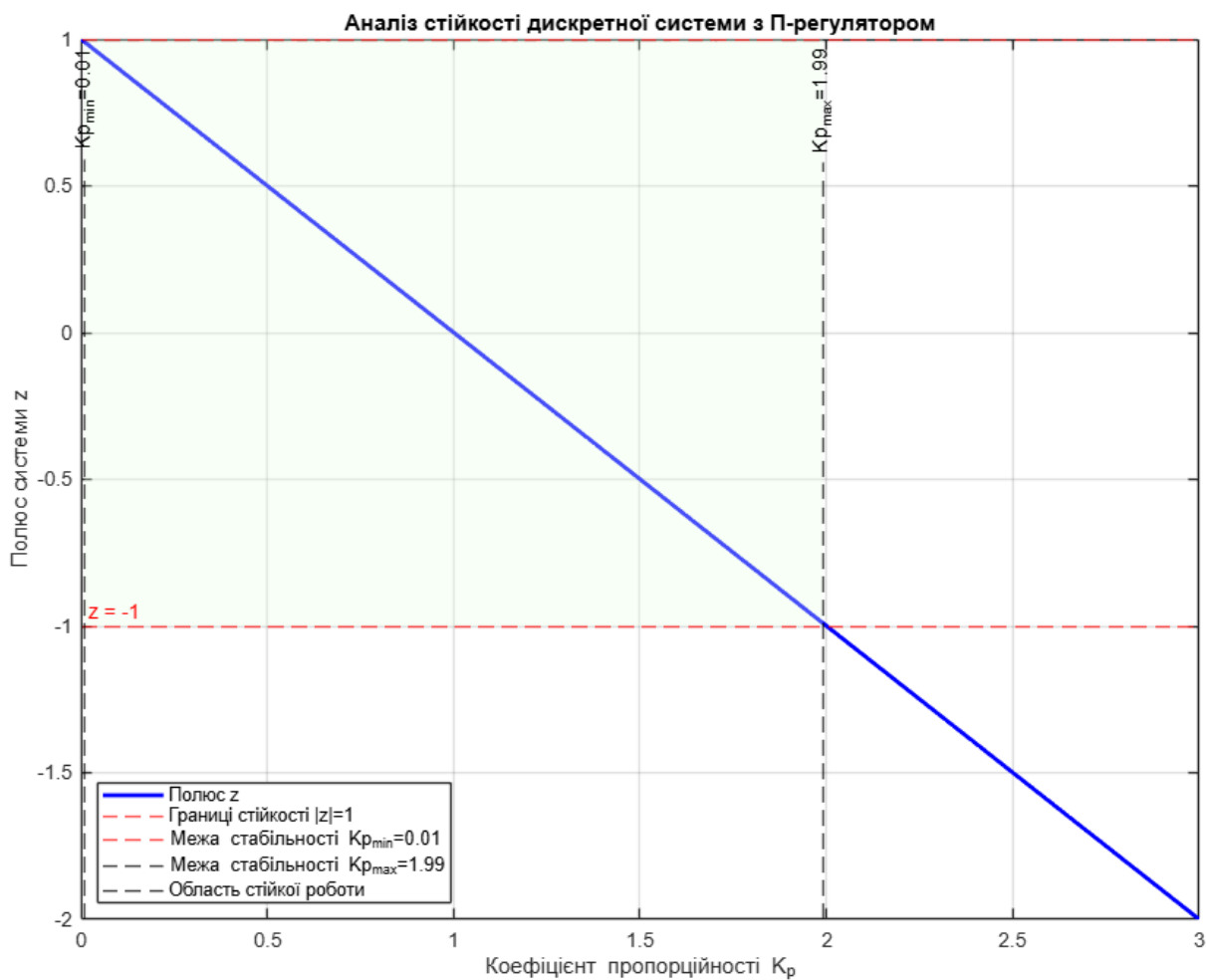


Рисунок 5.1 – Графік стабільних діапазонів коефіцієнту пропорційності

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи була створена система для автоматичної візуалізації стану робота-промоутера. Спершу був проведений аналіз існуючих технологій, що дозволив визначити ключові компоненти для інтеграції в систему, можливості голосового управління, розпізнавання обличчя та управління механізмами робота за допомогою сервоприводів. Це дало змогу вибрати апаратні компоненти для системи. Головним “мозком” системи став мікроконтролер Raspberry Pi Zero 2 W з офіційною камерою для обробки зображень, мікрофоном для голосового управління став INMP441, для відтворення голосових відповідей був використаний MP3 модуль DFPlayer, для контролю серводвигунів був обраний PWM/Servo модуль з I2C інтерфейсом на мікроконтролері PCA9685. Було розроблено структурну схему системи, підбрано необхідну елементну базу та змонтовано апаратну частину. Після підготовки середовища для програмування мікроконтролера реалізовано програмне забезпечення, яке включає модулі розпізнавання обличчя, голосових команд, керування сервоприводами та відтворенням звукових сигналів. Додатково створено скрипт для синхронізації голосових команд з анімацією руху рота.

Результатом виконаної роботи стала повноцінна функціональна система, яка демонструє базові можливості людино-орієнтованої взаємодії та може бути використана як прототип або основа для подальших досліджень і вдосконалень у галузі сервісної робототехніки.

Проект сприяє реалізації цілей сталого розвитку України:

Ціль 4 – якісна освіта (практична підготовка IT-фахівців);

Ціль 9 – індустріалізація та інновації (розробка інтелектуальної системи);

Ціль 12 – раціональне виробництво (енергоефективність, модульність).

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008-15. Документація. Звіти у сфері науки та техніки. Структура та правила оформлення. Введ. 2015-06-22. К. Держстандарт України, 2017. - 29 с.
2. Семенець В. В., Невлюдов І. Ш., Сінотін А. М., Сотник С. В. Оформлення технічної документації : навч. посіб. / М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. Харків : ХНУРЕ, 2021. 148 с. ISBN 978-966-659-302-6. DOI: 10.30837/978-966-659-302-6.
3. Невлюдов І. Ш., Филипенко О. І., Токарева О. В., Новоселов С. П., Сичова О. В. / Навчальний посібник з підготовки кваліфікаційної роботи бакалавра для здобувачів вищої освіти денної і заочної форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» : навч. посіб. Харків : Видавництво Іванченка І. С., 2023. 151 с.
4. Офіційний сайт Misty Robotics [Електронний ресурс] / Режим доступу: <https://www.mistyrobotics.com>
5. Офіційний сайт робота Pepper / Aldebaran Robotics [Електронний ресурс] / Режим доступу: <https://aldebaran.com/en/pepper/>
6. Офіційний сайт бібліотеки для розпізнавання мовлення Vosk Speech Recognition. Alpha Cephei [Електронний ресурс] / Режим доступу: <https://alphacephei.com/vosk>
7. Офіційна бібліотека OpenCV для Python [Електронний ресурс] / Режим доступу: <https://pypi.org/project/opencv-python>
8. Інтернет магазин Arduino в Україні [Електронний ресурс] / Режим доступу: <https://arduino.ua>
9. Онлайн-платформа для проектування електронних схем [Електронний ресурс] / Режим доступу: <https://u.easyeda.com>. (Дата звернення: 15.05.2025).

10. Бєлий Я.В. Розпізнавання голосу за допомогою офлайн-бібліотеки VOSK в робототехніці // Матеріали всеукраїнської науково-практичної конференції «Комп'ютерно-інтегровані технології, автоматизація та робототехніка (CITAR'25)», 16–17 травня 2025 р., м. Харків [Електронний ресурс] / Режим доступу: <https://tapr.nure.ua/dijalnist-kafedri/naukova-robota/computer-integrated-technologies-automation-and-robotics>. (Дата звернення: 05.07.2025).

11. Невлюдов І. Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Ч. 1 : підручник. Харків : ФОП Панов А. М., 2021. 604 с. ISBN 978-617-7947-67-6.

12. Невлюдов І. Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Ч. 2 : підручник. Кривий Ріг : Видавець Чернявський Д. О., 2022. 424 с. ISBN 978-617-8045-44-9.

13. Невлюдов І. Ш., Андрусевич А. О., Євсєєв В. В. Проектування мобільних роботів на базі одноплатних комп'ютерів (Raspberry Pi і мови Python 3.6) : підручник. Харків : 2020. 257 с.