

Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр

Освітній рівень

Мікроконтролерна система керування роботом з використанням модуля ESP8266

Назва теми

КвРКІ. 2102100.19.02.37 ПЗ

Шифр

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»

Шифр, назва

Освітня програма «Комп'ютерна інженерія та програмування»

Назва

Виконав: студент IV курсу, група КІ2-19-2

[Підпис]
Підпис

П.П. Матушак

Ініціали, прізвище

Керівник

[Підпис]
Підпис, дата

О.С. Засорнов

Ініціали, прізвище

Нормоконтролер

[Підпис]
Підпис, дата

С.М. Лисенко

Ініціали, прізвище

До захисту допускаю:

Зав. кафедри комп'ютерної
інженерії та інформаційних
систем

[Підпис]
Підпис

Т.О. Говорушенко

Ініціали, прізвище

« 6 » червня 2023 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорущенко

“ 11 ” 01 2023 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Матушаку Павлу Петровичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Мікроконтролерна система керування роботом з використанням модуля ESP8266

Керівник проекту (роботи) Засорнов О.С., к.т.н., доц.

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 1.03.2023 р. № 5

2. Строк подання студентом проекту (роботи) на кафедру 01.06.2023 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Дослідження предметної області та постановка задачі

Проектування програмно-технічного засобу

Програмно-апаратна реалізація пристрою

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)


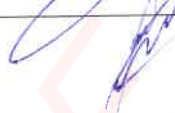

Схема електрична структурна

Схема електрична функційна

Схема електрична принципова

Алгоритм роботи

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Лисенко С.М., професор кафедри КІС		
Антиплагіат	Нічепорук А.О., доцент кафедри КІС		

7. Дата видачі завдання « 01 » 03 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітки
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи з керівником	20.02.2022	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.03.2023	виконано
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	10.03.2023	виконано
4	Робота над розділом 2 – проектування програмно-технічного засобу	20.04.2023	виконано
5	Робота над розділом 3 – програмно-апаратна реалізація пристрою	30.04.2023	виконано
6	Оформлення пояснювальної записки згідно вимог	20.05.2023	виконано
7	Попередній захист ВКР	25.05.2023	виконано
8	Захист ВКР на засіданні ЕК	Червень 2023 року	

Студент


Підпис

П.П. Матушак
Ініціали, прізвище

Керівник проекту (роботи)


Підпис

О.С. Засорнов
Ініціали, прізвище

№ рядка	Формат	Позначення	Найменування	Кількість	№ екземпляру	Примітка
			Текстові документи			
1	A4	КвРКІ. 2102100.19.02.37 ПЗ	Пояснювальна записка	64		
			Графічні матеріали	1		
2	A2	КвРКІ. 2102100.19.02.37 Е1	Схема електрична структурна	1		
3	A2	КвРКІ. 2102100.19.02.37 Е2	Схема електрична функційна	1		
4	A2	КвРКІ. 2102100.19.02.37 Е3	Схема електрична принципова	1		
5	A2	КвРКІ. 2102100.19.02.37 Е8	Алгоритм роботи	1		
6	A4	КвРКІ. 180246.18.02.18 ПЕ	Перелік елементів	1		

ийняв

имітка

конано

иконано

иконано

иконано

зиконано

виконано

виконано

вище

В
вище

КвРКІ. 2102100.19.02.37 ВП				
Зм.	Аркуш	№ докум.	Підп.	Дата
Розроб.		Матушак П.П.	<i>[Signature]</i>	
Перевір.		Засорнов О.С.	<i>[Signature]</i>	
Н. Контр.		Лисенко С.М.	<i>[Signature]</i>	
Затверд.		Говорушенко Т.О.	<i>[Signature]</i>	06.08
Мікроконтролерна система керування роботом з використанням модуля ESP8266 Відомість проекту			Літера	Аркуш
			Н	1
			ХНУ, КІ2-19-2	

АНОТАЦІЯ

Тема кваліфікаційної роботи: Мікроконтролерна система керування роботом з використанням модуля ESP8266

Автор роботи: Матушак Павло Петрович

Керівник роботи: к.т.н., доц. Засоронов Олександр Сергійович

Пояснювальна записка: 62 с., 26 рис., 3 дод., 46 джерел.

Графічна частина: 4 плакати.

ПРОГРАМНО-АПАРАТНИЙ ПРИСТРІЙ, РОБОТОТЕХНІКА, ARDUINO

Метою кваліфікаційної роботи є розробка системи дистанційного управління роботом, яка працює на базі контролера Arduino та керується за допомогою Android-пристрою через бездротову лінію зв'язку.

Для розробки був використаний новий модуль Wi-Fi - ESP8266, який забезпечує всі функції Wi-Fi та має низьку вартість. На основі цього модуля було створено точку доступу Wi-Fi та веб-сервер, до якого може підключатись клієнт та надсилати дані. ESP8266 отримує ці дані та передає їх на плату Arduino через послідовний порт. Arduino обробляє отримані дані та виконує відповідні команди.

Даний пристрій має широкий спектр застосувань, як у побуті та господарстві для дистанційного керування побутовими пристроями, так і в промисловості для керування виробничими процесами та електричним устаткуванням.


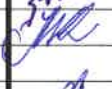

Дата 06.06.23



Підпис студента

ЗМІСТ

ВСТУП	3
1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ	5
1.1 Аналіз предметної області і виявлення наявних проблем	5
1.2 Огляд існуючих рішень	8
1.3 Постановка задачі	13
2 ПРОЄКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ	15
2.1 Обґрунтування вибору мікроконтролера	15
2.2 Організація доступу до середовища передачі даних	22
2.3 Система керування електродвигунами	27
2.4 Опис основних структурних блоків	30
2.5 Висновок	34
3 ПРОГРАМНО-АПАРАТНА РЕАЛІЗАЦІЯ ПРИСТРОЮ	36
3.1 Проєктування схеми електричної функціональної системи керування роботом	36
3.2 Аналіз схеми електричної принципової.....	38
3.3 Опис алгоритму роботи	49
3.3 Розробка програмного забезпечення	52
3.5 Розрахунок електричних параметрів	54
3.6 Висновок	56
ВИСНОВКИ	58
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	59
ДОДАТОК А Код програми для Arduino	63
ДОДАТОК Б Код програми для ESP8266	70
ДОДАТОК В Копія графічної частини	72

КВРКІ. 2102100.19.02.37 ПЗ									
№	Док.	№ док.	Підпис	Дата	Мікроконтролерна система керування роботом з використанням модуля ESP8266 Пояснювальна записка	Літера	Аркш	Аркушів	
Виконав		Матушак П.П.						2	62
Перевір.		Засорнов О.С.							
Н. контр. Патвер.		Лисенко С.М. Говорущенко Т.П.		6.06					
						ХНУ, КІ2-19-2			

ВСТУП

Мікроконтролерні системи є не менш важливою частиною нашого життя, ніж стандартні комп'ютери, та використовуються у звичайних речах, таких як пральні машини, автомобілі, холодильники та інші пристрої. Вони також широко використовуються в інноваційних системах, таких як "Розумний дім", а також у робототехніці.

Історія Wi-Fi починається в середині 1990-х років, коли технологія передачі інформації по радіоканалу була розроблена для використання у локальних мережах великих корпорацій та компаній у Кремнієвій долині США. Спочатку ця технологія використовувалася для забезпечення зв'язку між співробітниками компаній, обладнаними ноутбуками з бездротовим мережним адаптером, та "точками доступу", підключеними до кабельної інфраструктури компаній. Згодом термін Wi-Fi став використовуватися для позначення інших бездротових технологій локальних мереж, таких як IEEE 802.11a і 802.11g, які мають велику швидкість передачі інформації і використовують різні частотні діапазони [1].

Системи дистанційного керування є невід'ємною частиною сучасного життя і широко використовуються для збору та обробки інформації під час роботи обладнання, дистанційного моніторингу процесів, його захисту [2, 3]. На жаль, більшість існуючих систем зв'язку з цими пристроями можуть використовуватися безпосередньо людьми, що обмежує їх функціональність. У цій кваліфікаційній роботі пропонується модель дистанційного керування мікроконтролерними системами, розроблена на основі архітектури клієнт-сервер, яка надає багато нових можливостей для керування пристроями.

Дослідження проводилися на базі Arduino UNO, на основі якої був побудований колісний робот з дистанційним керуванням за допомогою пристрою Android. У роботі розглянуто кілька режимів роботи, які демонструють можливості цієї моделі управління. Для забезпечення доступу до середовища передачі даних використовується спеціальний модуль ESP8266,

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						3
Зм..	Арк.	№докум.	Підпис	Дата		

який забезпечує зв'язок і передачу даних по мережі Wi-Fi. Ця робота є актуальною, оскільки надає нові можливості дистанційного керування пристроями, що важливо для багатьох галузей, включаючи наукові дослідження, індустрію та розваги.

Отже, метою написання кваліфікаційної бакалаврської роботи є вирішення конкретної практичної задачі в галузі обчислювальної техніки та отримання відповідного прикладного результату у вигляді апаратного забезпечення, яке буде функціонально ефективним.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						4
Зм..	Арк.	№докум.	Підпис	Дата		

1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз предметної області і виявлення наявних проблем

Створення системи дистанційного керування роботами має кілька важливих переваг. По-перше, це дозволяє дистанційно керувати роботом, що дуже зручно для деяких застосувань, наприклад у важкодоступних місцях або в умовах підвищеного ризику. По-друге, це дозволяє дистанційно керувати кількома роботами одночасно, значно підвищуючи ефективність і продуктивність робіт у відповідних програмах.

Крім того, системи дистанційного керування дозволяють істотно скоротити витрати на допоміжний персонал, оскільки для управління роботами достатньо однієї людини в центрі управління. Крім того, дистанційне керування роботом забезпечує більш точне та швидке керування, оскільки немає необхідності у фізичному спілкуванні між оператором і роботом, а команди передаються миттєво.

При створенні системи дистанційного керування роботом використовуються різні технології та архітектури, такі як клієнт-серверна архітектура, мережі Wi-Fi та інші. Для успішного впровадження системи дистанційного керування роботом необхідні знання та навички в області програмування, мікроконтролерів, а також розуміння принципів робототехніки.

Існує безліч прикладів систем віддаленого керування, оскільки вони застосовуються в багатьох галузях, таких як [4]:

1. Промисловість, куди входять системи дистанційного керування використовуються для моніторингу та контролю виробничого обладнання для забезпечення безпеки та ефективності роботи.

2. Медицина, наприклад для системи дистанційного керування дозволяють лікарям дистанційно контролювати та контролювати медичні пристрої, такі як системи дистанційного керування хірургічними роботами.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						5
Зм.	Арк.	№докум.	Підпис	Дата		

3. Транспортна галузь використовує системи дистанційного керування в авіаційній, залізничній та автомобільній промисловості для моніторингу та керування транспортними засобами.

4. Для сільського господарства системи дистанційного керування використовуються для керування різними пристроями в сільському господарстві, такими як трактори, комбайни тощо.

5. В робототехніці системи дистанційного керування є необхідним елементом керування роботами в різних сферах,

6. Для смарт будинків системи дистанційного керування використовуються для управління різними побутовими приладами, наприклад, кондиціонерами, освітленням, побутовою технікою [5 - 8].

Це лише декілька прикладів, але дійсно існує безліч варіантів використання систем віддаленого керування, які знаходять застосування в різних галузях.

Arduino - це мікроконтролерна платформа з відкритим кодом, яка використовується для створення різних електронних пристроїв [9, 10]. Вона дозволяє легко програмувати і контролювати різні фізичні пристрої та датчики за допомогою простого інтерфейсу.

Arduino широко використовується в системах дистанційного керування, оскільки це досить проста у використанні платформа та досить дешева. Ось кілька прикладів використання Arduino в системах дистанційного керування:

1. При дистанційному керуванні роботами можна використовувати для створення дистанційно керованих роботів. За допомогою спеціальних модулів Wi-Fi або Bluetooth можна підключити Arduino до мобільного пристрою та керувати роботом за допомогою програми на смартфоні чи планшеті.

2. Для систем керування кріпленням камери Arduino можна використовувати для створення систем керування кріпленням камери. Така система дозволяє віддалено змінювати положення камери через Wi-Fi або Bluetooth.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		6

3. Для автоматизованих систем поливу рослин модулі можна використовувати для створення автоматизованих систем поливу рослин.

4. Для систем домашньої автоматизації мікроконтролер можна використовувати для створення систем домашньої автоматизації. За допомогою датчиків і спеціальних пристроїв Arduino може контролювати температуру, освітлення та інші параметри будинку і змінювати їх за допомогою мобільного додатку [11, 12].

Окрім згаданих вище прикладів, Arduino знайшов застосування в багатьох інших областях. Наприклад, його можна використовувати для створення систем автоматизації типу «розумний дім», де він використовується для управління освітленням, системами безпеки, системами опалення та кондиціонування. Arduino також дозволяє створювати різні інтерактивні проекти, такі як інтерактивні художні інсталяції, роботів, які реагують на звук або рух, і багато іншого. Розробка систем дистанційного керування на основі мікроконтролерів, таких як Arduino, може спричинити деякі проблеми та труднощі. Деякі з них:

– обмежені ресурси, адже мікроконтролери мають обмежений обсяг пам'яті та обчислювальної потужності, що може обмежити можливості розвитку дистанційного керування [13];

– системи віддаленого керування можуть бути атаковані з мережі, тому важливо забезпечити відповідний рівень безпеки, наприклад безпечне спілкування;

– для реалізації багатьох функцій дистанційного керування необхідно розробити та реалізувати складні алгоритми керування, що може бути складним завданням;

– для розробки дистанційного керування вимагає розуміння протоколів зв'язку, таких як Wi-Fi або Bluetooth, і роботи зі спеціалізованими модулями для їх реалізації [14];

– якщо на продуктах використовується система дистанційного керування, необхідно забезпечити відповідну підтримку та можливість змінити систему, якщо це необхідно.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						7
Зм..	Арк.	№докум.	Підпис	Дата		

Ці труднощі можуть бути подолані завдяки належному плануванню, розробці та тестуванню систем віддаленого керування на базі мікроконтролерів.

1.2 Огляд існуючих рішень

Нижче наведено опис пристроїв, які виконують функцію дистанційного управління та мають додаткові можливості, як промислові, так і радіолюбительські.

Пристрій для керування кроковим двигуном за допомогою ESP8266 і ATtiny85 використовує для руху крокового двигуна спеціальні сигнали, які повинні бути згенеровані мікроконтролером (рис 1.1) [15, 16].

ESP8266 відповідає за збір і передачу команд керування, а ATtiny85 відповідає за генерацію необхідних сигналів керування рухом крокового двигуна.

Пристрій складається з ESP8266, ATtiny85, драйвера крокового двигуна, а також додаткових компонентів, таких як резистори, конденсатори та інші.

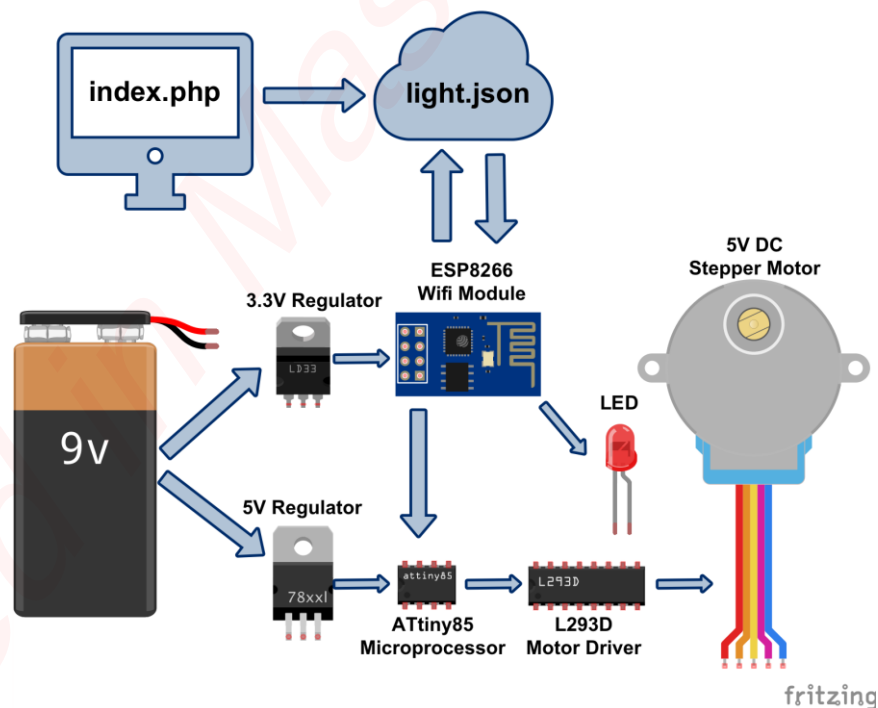


Рисунок 1.1 – Схема пристрою для управління кроковим двигуном за допомогою ESP8266

Зм.	Арк.	№докум.	Підпис	Дата

Даний пристрій здійснює керування через Інтернет і базується на мікроконтролері ATtiny85, який можна прошивати з використанням Arduino та Arduino IDE, а також включає в себе L293D Motor Driver та ESP8266-01. Його головною перевагою є компактність, що дозволяє зменшити розміри робота, проте його недоліком є те, що він підтримує тільки один мотор та не має окремого живлення, що може вплинути на стабільність роботи мікроконтролеру ATtiny85.

Пристрій для керування кроковим двигуном за допомогою ESP8266 і ATtiny85 може бути використаний у різних галузях, наприклад, у робототехніці, автоматизації виробничих процесів та інших сферах, де потрібен точний та надійний рух механізмів.

Wi-Fi робот на Arduino - це робот, який може бути керований за допомогою бездротової мережі Wi-Fi. Для цього використовується мікроконтролер Arduino з модулем Wi-Fi ESP8266 [17, 18]. Такий робот може бути використаний в багатьох галузях, де потрібне безпосереднє керування пристроями на відстані (рис. 1.2).

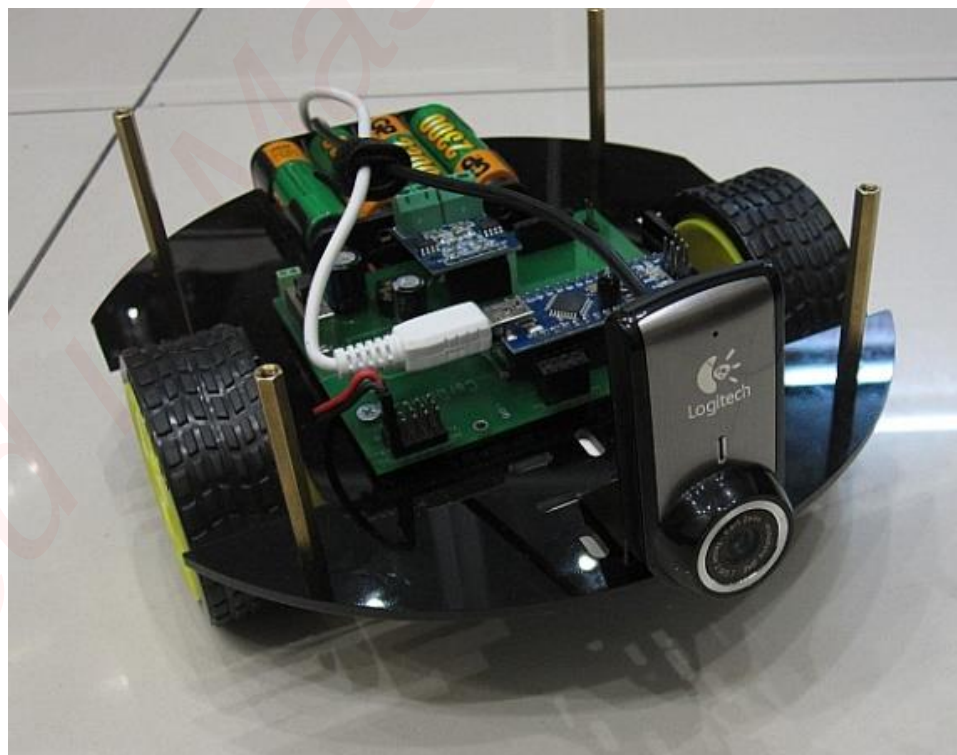


Рисунок 1.2 – Wi-Fi робот

Зм.	Арк.	№докум.	Підпис	Дата

Для створення цього робота були використані наступні компоненти: монтажний набір робота, плата контролера робота, Arduino Nano v.7, контролер двигуна, мініатюрний концентратор USB 2.0 і маршрутизатор WT3020H [19]. Роботом можна керувати з комп'ютера за допомогою роутера, прошитого CyberWrt. Особливістю даного пристрою є наявність веб-камери, яка дозволяє транслювати відео на екран комп'ютера, звідки ним керують. Однак недоліком є велика кількість споживаної енергії роутера, що може обмежити тривалість автономної роботи.

Одним із способів керування роботом є використання пристрою Android як пристрою керування [20]. Для цього часто використовують спеціальні мобільні додатки, які дозволяють контролювати рух робота. Особливістю такої системи полягає в тому, що вона дозволяє користувачам управляти комфортним для них способом, використовуючи сенсорний екран свого мобільного пристрою.

Як правило, для зв'язку між пристроєм Android і роботом використовуються бездротові технології, такі як Wi-Fi або Bluetooth. Деякі роботи можуть мати вбудовані модулі Wi-Fi або Bluetooth, що дозволяє спілкуватися без додаткового обладнання. Іншим роботам може знадобитися додаткове обладнання, наприклад модулі Bluetooth або Wi-Fi, які потрібно буде підключити до робота.

Для забезпечення безпеки управління роботом через Android-пристрій, додатки зазвичай мають різні заходи захисту, такі як автентифікацію користувача, пароль або інші методи. Недоліком такої системи може бути те, що вона залежить від живлення Android-пристрою, яке може бути обмеженим в часі або проблемним в залежності від умов використання (рис. 1.3.).

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						10
Зм..	Арк.	№докум.	Підпис	Дата		

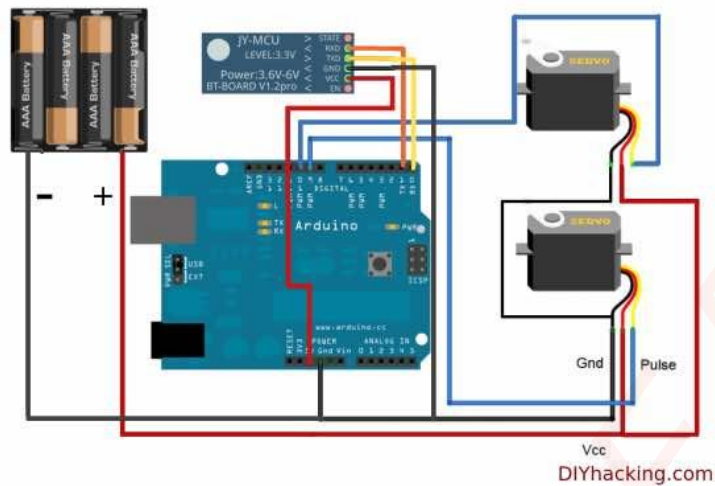


Рисунок 1.3 – Схема пристрою

Wi-Fi робот з відеотрансляцією та виявленням перешкод [21, 22] - це робот, який може бути керований за допомогою Wi-Fi зв'язку та має можливість передачі відео в реальному часі на додаток на смартфоні чи іншому пристрої (рис 1.4).

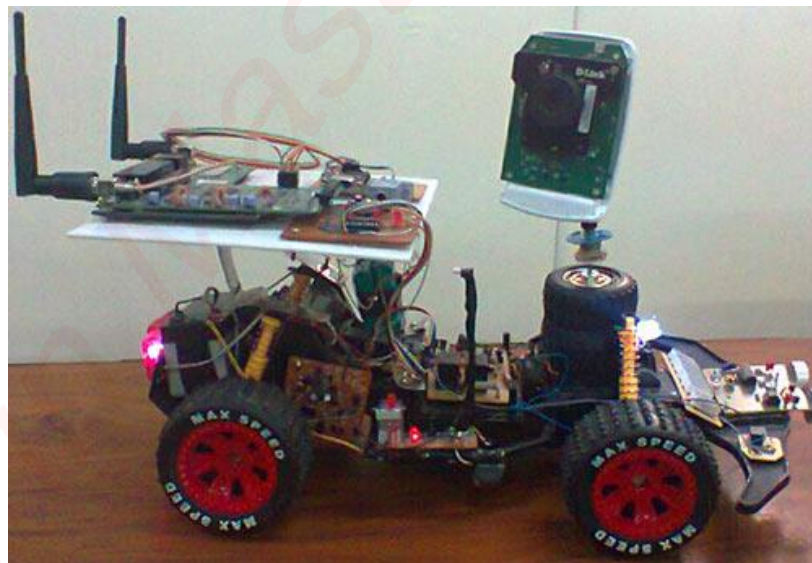


Рисунок 1.4 – Wi-Fi робот з відеотрансляцією та виявленням перешкод

Цей пристрій має вагомую перевагу високої швидкості передачі даних і відтворення відео в реальному часі. Однак, недоліком є те, що кожен компонент

пристрою потребує свого власного джерела живлення, що призводить до збільшення ваги пристрою.

Розглянуті пристрої мають свої переваги та недоліки.

Пристрій для керування кроковим двигуном за допомогою ESP8266 і ATtiny85 має компактний розмір, але може мати проблеми зі стабільністю роботи через використання одного мотора без окремого живлення.

Робот Wi-Fi на Arduino забезпечує керування роботом Wi-Fi, а також потокове відео в реальному часі, але він споживає багато енергії через використання маршрутизатора для передачі сигналу.

Робот Wi-Fi з передачею відео в реальному часі та виявленням перешкод має можливість автоматично виявляти та уникати перешкод, а також дозволяє віддалено керувати роботом через Wi-Fi, але він досить великий і потребує окремого джерела живлення . для кожного елемента пристрою.

Для програмування мікроконтролера можна використовувати середовище Arduino IDE. Для передачі відео на комп'ютер або мобільний пристрій можна використовувати бібліотеки OpenCV або MJPEG-Streamer [23, 24]. Для управління роботом з мобільного пристрою можна створити додаток на платформі Android або iOS.

Підключивши датчики відстані до мікроконтролера, робот зможе автоматично уникати перешкод. Для цього можна написати програму, яка буде отримувати дані від датчиків відстані та вибирати оптимальний шлях для переміщення робота.

Кожен пристрій має свої переваги і недоліки, тому при виборі потрібно враховувати ці фактори і визначити, які можливості пристрою найбільш важливі для конкретної ситуації. Оскільки вартість багатьох пристроїв може бути досить високою, багато користувачів віддають перевагу більш доступним аналогам. Тому метою розробки буде підвищення енергоефективності, за рахунок високого енергоспоживання, що може призвести до скорочення часу роботи на одному заряді та зниження вартості, оскільки багато користувачів віддають перевагу більш доступним аналогам.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						12
Зм..	Арк.	№докум.	Підпис	Дата		

1.3 Постановка задачі

Цей проект спрямований на створення робота, який працюватиме на контролері Arduino та керуватиметься з пристроєм Android за допомогою бездротової лінії зв'язку. Головними перевагами цього проекту є висока маневреність і можливість використання в різних сферах, включаючи наукові дослідження, промисловість і розваги. Однак для досягнення мети проекту необхідно вирішити проблему енергоспоживання та забезпечити стабільну роботу всіх компонентів системи.

Створення робота на основі контролера Arduino, керованого пристроєм Android через бездротове з'єднання, має кілька переваг.

По-перше, контролер Arduino відкритий і доступний кожному, що робить його зручним інструментом для розробки різноманітних проектів. Крім того, Arduino має велику спільноту користувачів, що дозволяє отримати підтримку та поради від досвідчених розробників.

По-друге, керування роботом з Android-пристроєм через бездротову лінію зв'язку є зручним і мобільним способом управління роботом з віддаленого місця. Це забезпечує зручність використання робота в різних умовах, коли з ним важко або неможливо працювати безпосередньо.

По-третє, використання бездротової лінії зв'язку дозволяє передавати відео в режимі реального часу з камери робота на Android-пристрій, що дозволяє контролювати стан робота та використовувати його для дистанційного спостереження.

Прилад повинен відповідати наступним вимогам:

1. Основна вимога до приладу – його доступність і вартість. Необхідно, щоб він був максимально простим і не вимагав великих монтажних і матеріальних витрат.

2. Вимога до системи управління полягає в тому, щоб вона була легко зрозумілою і доступною для будь-якого користувача без спеціальних знань і навичок.

					КВРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		13

3. Пристрій має бути універсальним і не мати особливих вимог до конструкції, щоб його можна було використовувати в різних умовах і середовищах.

4. Вимога надійності та стабільної роботи є життєво важливою для пристрою. Він повинен працювати безперебійно і надійно протягом тривалого часу.

Призначення та технічні характеристики пристрою:

- Керування електромоторами з віддалення.
- Забезпечення автоматичного режиму.
- Можливість вибору різних режимів керування.
- Універсальність застосування для використання в різних сферах діяльності. Опис технічних характеристик системи:

- Напруга живлення - від 7 до 12 В зі струмом до 1 Або 12В зі струмом до 2Або 3,3 В зі струмом до 800 мА.

- Діапазон відстані вимірювання ультразвукового датчика - від 2 до 400 см.

- Дистанція зв'язку Wi-Fi - до 100 метрів або більше з антеною.

- Вихідна потужність Wi-Fi - до + 25 дБм.

- Максимальна швидкість руху - до 15 км/год.

Ці характеристики забезпечать ефективну та надійну роботу системи на великій відстані та у різних умовах застосування.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм..	Арк.	№докум.	Підпис	Дата		14

2 ПРОЄКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ

2.1 Обґрунтування вибору мікроконтролера

Загалом, Arduino є досить універсальною платформою, яку можна використовувати для різноманітних додатків, починаючи від простих електронних проєктів і закінчуючи складними системами керування та автоматизації [8]. Його висока доступність і відкритість дозволяють використовувати його широкому колу людей для різних цілей, включаючи студентів, ентузіастів і професіоналів.

Робот Wi-Fi на Arduino — це робот, яким можна керувати за допомогою бездротової мережі Wi-Fi. Для цього використовується мікроконтролер Arduino з модулем Wi-Fi ESP8266. Такого робота можна використовувати в багатьох галузях промисловості, де потрібне пряме керування віддаленими пристроями.

Одним із прикладів використання робота Wi-Fi в Arduino є його використання в промисловості. Наприклад, його можна використовувати для дистанційного керування таким обладнанням, як кілька насосів або конвеєрів. Wi-Fi-робота також можна використовувати для дистанційного керування виробничими процесами.

Іншим прикладом використання робота Wi-Fi в Arduino є його використання в освіті. Його можна використовувати для вивчення основ програмування та робототехніки, оскільки він дозволяє дистанційно керувати та програмувати робота.

Щоб створити Wi-Fi робота на Arduino, потрібно використовувати Arduino UNO або будь-який інший сумісний мікроконтролер, модуль Wi-Fi ESP8266, драйвер двигуна L293D і різні датчики. Після складання його необхідно запрограмувати за допомогою Arduino IDE та встановити програми для керування мережею Wi-Fi [9, 19].

Відповідно до технічного завдання на дипломну кваліфікаційну роботу, необхідно розробити робота на базі контролера Arduino, який буде управлятися

					КВРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		15

через бездротову лінію зв'язку з пристроєм Android. Щоб забезпечити цю функцію, ми розробимо систему мікроконтролера, яка обмінюватиметься інформацією з пристроєм Android через мережу Wi-Fi. Отримані з пристрою дані будуть використовуватися для управління рухом робота. Для цього ми використовуємо контролер двигуна постійного струму L298N, який забезпечує потужність і точне керування рухом [26-28].

За допомогою нашого проекту буде можливо контролювати робота на відстані та забезпечити його автоматичний режим. Крім того, проект має універсальне застосування та може бути використаний в різних галузях людської діяльності. Технічні характеристики нашої системи включають живлення від 7 до 12 вольт, діапазон дальності ультразвукового датчика відстані від 2 до 400 см, дистанцію зв'язку Wi-Fi до 100 метрів і більше з антеною, вихідну потужність Wi-Fi від 25 dBm та можливість руху зі швидкістю до 15 км/год [32].

Для керування роботом, що буде розроблено в рамках дипломного проектування, використовуватиметься плата Arduino як керуючий елемент системи. Arduino - це відкрите програмне та апаратне середовище, що дозволяє створювати різноманітні електронні проекти з використанням різних компонентів та сенсорів. Завдяки можливості завантажування програми на процесор Arduino, розробник може створити власний алгоритм управління роботом.

Плата Arduino має багато вхідних та вихідних контактів, які можна використовувати для з'єднання з різними компонентами, такими як мотори, датчики, світлодіоди тощо. Крім того, Arduino може використовуватися для створення інтерактивних систем з використанням різних датчиків та перемикачів.

Вибір плати Arduino для керування роботом в рамках дипломного проектування є обґрунтованим, оскільки вона є досить простою у використанні та програмуванні, має багатий функціонал та забезпечує зручні можливості для взаємодії з іншими пристроями, в тому числі з пристроями на базі операційної системи Android через бездротову лінію зв'язку Wi-Fi [9, 29].

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм..	Арк.	№докум.	Підпис	Дата		16

Вибір моделі Arduino залежить від функцій, які має виконувати пристрій. Наприклад, для пристрою, який має бути компактним, можна вибрати модель Arduino Mini або Arduino Micro, оскільки вони мають менший розмір та менше кількість контактів. Але якщо до пристрою має бути приєднано багато інших компонентів, бажано вибрати модель з більшою кількістю контактів, наприклад, Arduino Mega.

Arduino Uno є досить популярною моделлю, оскільки вона має оптимальне співвідношення розміру та функціональності. Вона має 14 цифрових входів/виходів, 6 аналогових входів, може працювати з частотою 16 МГц та має 32 кБ флеш-пам'яті для зберігання програм. Крім того, ця модель є однією з найбільш доступних та підтримується великою спільнотою розробників.

Однак, якщо ваш пристрій вимагає більшого розміру плати, наприклад, для використання в невеликому проекті, тоді Arduino Mini або Arduino Micro можуть бути кращим варіантом. На відміну від Arduino Uno, вони менші за розміром і мають менше контактів. Однак слід пам'ятати, що ці моделі мають обмежену функціональність порівняно з більш потужними моделями, такими як Arduino Mega. Отже, якщо ваш пристрій має велику кількість компонентів і потребує багато енергії, Arduino Mega може бути кращим варіантом.

Але ми зупинимо свій вибір на Arduino Uno і розглянемо її переваги. Це одна з найпопулярніших моделей мікроконтролерів сімейства Arduino (рис. 2.1). Основні переваги Arduino Uno наступні:

- простий і зрозумілий дизайн, що робить його ідеальним для початківців у електроніці та програмуванні і для розробки на Arduino Uno використовується мова програмування C++, яка досить проста і доступна більшості програмістів;
- має вбудований порт USB, який дозволяє завантажувати програми на мікроконтролер і читати дані швидко і легко;
- Arduino Uno має 14 цифрових входів-виходів, з яких 6 можна використовувати як аналогові входи, що дозволяє підключати багато датчиків і пристроїв до мікроконтролера;

- можна легко розширити додатковими модулями і датчиками, дозволяючи створювати складні системи зв'язку зі світом;
- є частиною відкритої спільноти, що означає, що документація, схеми, код та інші ресурси є у вільному доступі для використання та редагування. Це дозволяє розробникам та ентузіастам додавати та покращувати систему за потреби. Крім того, на Arduino Uno можна встановити багато програм і бібліотек, розроблених спільнотою користувачів.

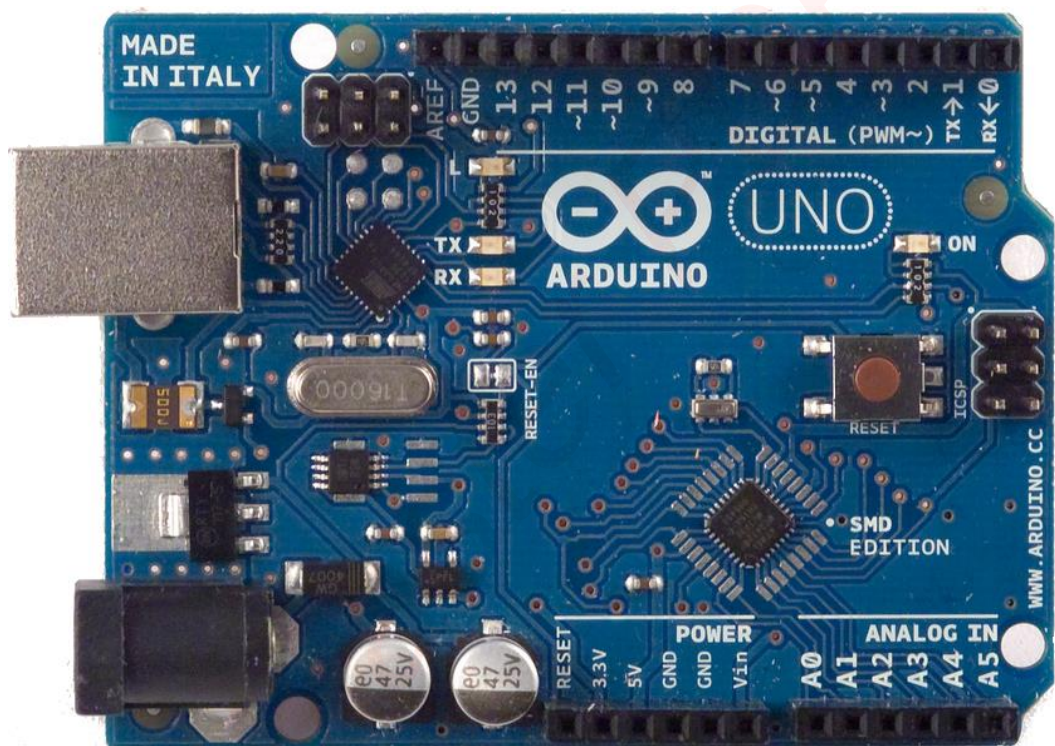


Рисунок 2.1 - Arduino UNO

Arduino Uno є пристроєм, який базується на відомому мікроконтролері ATmega328. Для того, щоб розпочати роботу з пристроєм, достатньо лише підключити живлення від AC / DC-адаптера чи батарейки. Крім того, є можливість підключення до комп'ютера за допомогою USB-кабелю.

Основні характеристики Arduino Uno:

- мікроконтролер: ATmega328P;
- вхідна напруга (рекомендована): 7-12 В;
- вхідна напруга (максимальна): 20 В;

- робоча напруга: 5 В;
- цифрові входи/виходи: 14 (включаючи 6 виходів PWM);
- струм, що постачається через вихідні піни: 20 мА;
- аналогові входи: 6;
- пам'ять Flash: 32 Кб (0,5 Кб використовуються для загрузчика);
- EEPROM: 1 Кб;
- оперативна пам'ять (RAM): 2 Кб;
- частота кварцового генератора: 16 МГц.

Arduino Uno має просту інтерфейсну панель з контактами, на які можна підключати різні сенсори, актуатори, індикатори та інші електронні пристрої. Перевагою Arduino Uno є те, що вона підтримує безліч додаткових модулів, що дозволяє розширювати її можливості. Також Arduino Uno має вбудований USB-порт для зручного програмування та завантаження коду на мікроконтролер. Arduino Uno є досить простою у використанні та зрозумілою для початківців, що робить її популярною у галузі макетування та розробки прототипів.

Замість використання мікросхеми FTDI, пристрій використовує мікроконтролер ATmega16U2 як перетворювач інтерфейсів USB-UART [30]. Цей мікроконтролер дозволяє забезпечити зручне підключення до комп'ютера через USB-порт і взаємодію з Arduino Uno за допомогою UART-протоколу. Це забезпечує зручний та швидкий спосіб зміни програмного забезпечення пристрою та забезпечує широку сумісність з багатьма операційними системами. Крім того, використання мікроконтролера ATmega16U2 (або ATmega8U2) дозволяє забезпечити додаткові функції, такі як можливість заблокувати запис до пам'яті мікроконтролера, що забезпечує більш високий рівень захисту від несанкціонованого доступу до програмного забезпечення пристрою.

Мова програмування для пристроїв Arduino базується на мові C/C++ [10]. Ця мова програмування вважається дуже доступною і легкою для вивчення навіть новачкам. Сьогодні Arduino є одним з найбільш зручних і широко використовуваних інструментів для програмування пристроїв на мікроконтролерах.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						19
Зм..	Арк.	№докум.	Підпис	Дата		

Для програмування пристроїв Arduino використовується USB-кабель з роз'ємами USB-A і USB-B, що забезпечує зв'язок між пристроєм і комп'ютером. Цей кабель дозволяє завантажувати програмне забезпечення на плату Arduino з комп'ютера, а також отримувати дані з пристрою. Завдяки використанню USB-кабелю програмування пристроїв Arduino стає ще простішим і зручнішим процесом [31].

Arduino IDE – інтегроване середовище розробки на платформі Arduino, що працює на базі Java [10]. Він складається з редактора коду, компілятора та модуля передачі прошивки на плату. Його основна мета — полегшити процес програмування для початківців, які ще не мають достатнього досвіду розробки програмного забезпечення. Arduino IDE базується на мові програмування Processing і має синтаксис, подібний до мови Wiring. Строго кажучи, це розширений C++, який включає деякі бібліотеки. Програми обробляються препроцесором і компілюються за допомогою AVR-GCC. Використання Arduino IDE полегшує та прискорює розробку програмного забезпечення для платформи Arduino без необхідності використання додаткових програм або знання мов програмування високого рівня.

В середовищі Arduino IDE потрібно обрати свою плату, для програмування.

В меню Tools > Serial Port необхідно обрати послідовний порт, до якого підключений ваш Arduino. Зазвичай це COM-порт з номером 3 (COM3) або вище, оскільки порти COM1 і COM2 зазвичай зарезервовані для апаратних портів.

Щоб дізнатися номер потрібного порту, можна тимчасово від'єднати Arduino та знову відкрити меню. Порт, який зник, буде асоційований з вашим Arduino. Після цього потрібно знову підключити пристрій до комп'ютера та вибрати необхідний порт з меню.

Для перевірки роботи Arduino можна скористатися програмою мигання світлодіода, який вже є на платі та підключений до виходу 13 (рис. 2.3).

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		20

Ця програма вмикає та вимикає світлодіод, підключений до виходу 13 плати Arduino, з частотою раз на секунду.

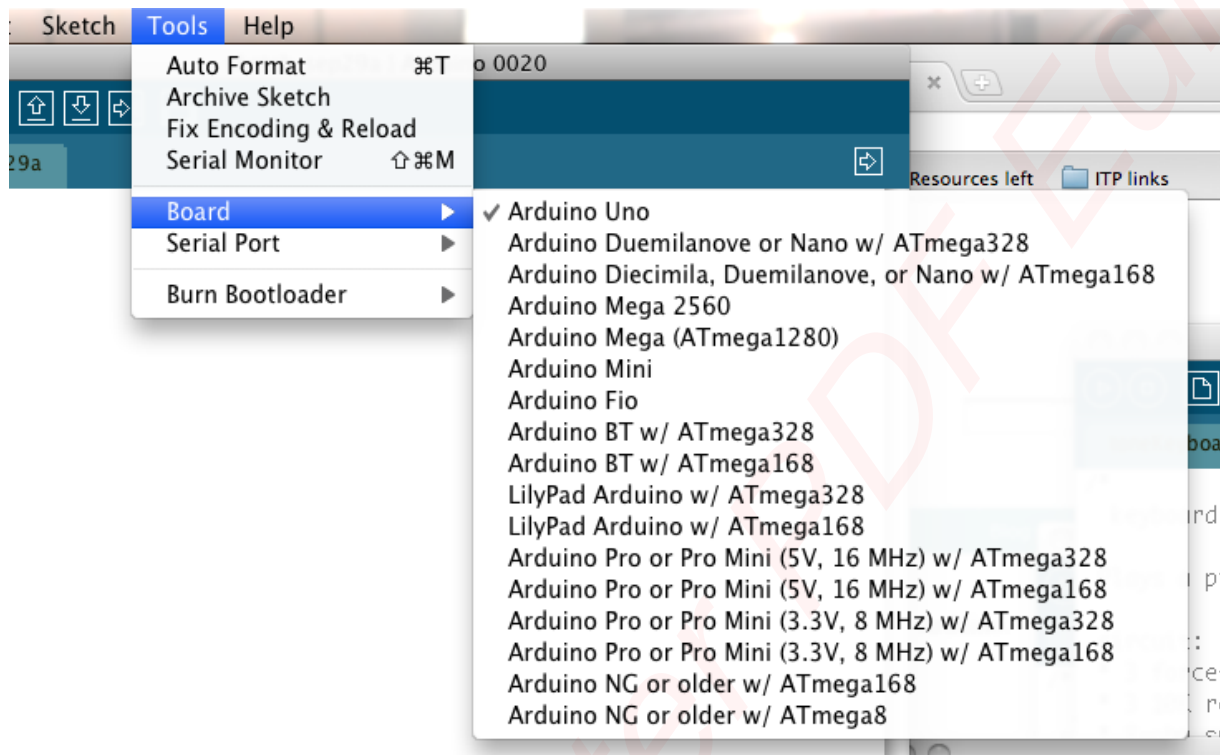


Рисунок 2.2 – Середовище Arduino IDE

Змінна `led` зберігає номер контакту, до якого підключено світлодіод. У цьому випадку їх 13.

У функції `setup()` режим вихідного виводу з номером світлодіода встановлюється функцією `pinMode()`. Це дозволяє вказати, що штифт буде використовуватися як вихід, тобто для керування світлодіодом.

У функції `loop()` за допомогою функції `digitalWrite()` на контакті з номером світлодіода встановлюється висока напруга, яка вмикає світлодіод. Після цього виконується функція `delay()`, що призводить до затримки в одну секунду. Потім функція `digitalWrite()` встановлює пін із номером світлодіода на низький рівень, вимикаючи світлодіод. Після цього функція `delay()` виконується знову, що викликає затримку.

```

int led = 13;

void setup() {
  pinMode(led, OUTPUT);
}

void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(500); // wait for half a second
  digitalWrite(led, LOW); // turn the LED off by making the voltage LOW
  delay(500); // wait for half a second
}

```

Рисунок 2.3 – Тестова програма

2.2 Організація доступу до середовища передачі даних

Для організації доступу до середовища передачі даних в бакалаврській роботі використано спеціальний модуль Wi-Fi ESP8266 [17, 18].

ESP8266 - це популярний модуль Wi-Fi, який забезпечує можливість підключення до бездротової мережі за допомогою протоколу Wi-Fi. Це автономний мікроконтролер із вбудованим Wi-Fi, що дозволяє йому взаємодіяти з іншими пристроями через Інтернет.

ESP8266 має вбудований стек TCP/IP, який дозволяє передавати дані через Інтернет за допомогою різних протоколів, таких як HTTP, MQTT і UDP. Ви можете використовувати ESP8266 для підключення до різних засобів масової інформації, таких як сервер на хмарній платформі або локальний сервер у локальній мережі.

Щоб отримати доступ до середовища передачі даних за допомогою ESP8266, ви повинні спочатку підключити модуль до пристрою або плати мікроконтролера. Потім ви можете використовувати таке програмне забезпечення, як Arduino IDE або MicroPython, щоб написати програму, яка керуватиме ESP8266 і передаватиме дані через Wi-Fi.

За допомогою ESP8266 ви можете підключатися до мережі Wi-Fi, надсилати та отримувати дані, контролювати вхід/вихід (GPIO) модуля, робити

HTTP-запити, створювати власні мережеві сервери та багато іншого. Це забезпечує велику гнучкість для розробки проектів, які вимагають бездротового зв'язку.

ESP8266 також підтримує різні режими роботи, включаючи режим клієнта, точку доступу (Access Point) і режим підключення між точками доступу (режим Station + SoftAP). Це дозволяє налаштувати роботу модуля в потрібному режимі відповідно до ваших потреб. Це дуже доступний модуль Wi-Fi порівняно з іншими бездротовими рішеннями. Він може бути легко доступним для широкого кола розробників та ентузіастів, що робить його популярним у спільноті IoT. Існує багато документації, прикладів і бібліотек, які полегшують розробку проектів за допомогою ESP8266.

Незважаючи на численні переваги, ESP8266 також має кілька недоліків:

- має обмежений обсяг оперативної пам'яті та місця для зберігання програмного забезпечення, що може бути обмеженням для складних програм або ресурсомістких проектів;

- не найдешевший варіант за енергоспоживанням, адже модуль може потребувати значної кількості енергії, особливо при використанні інтенсивного Wi-Fi, що може бути проблемою для проектів з обмеженим джерелом живлення або акумулятора;

- через низьку потужність передачі сигналу ESP8266 може бути вразливим до перешкод у бездротовій мережі, що може призвести до погіршення якості зв'язку або втрати зв'язку, особливо на великих відстанях або в середовищах із високим рівнем шуму;

- обмежений діапазон підтримуваних протоколів, не підтримує деякі спеціалізовані бездротові стандарти або протоколи;

- для стабільної роботи ESP8266 вимагає джерела живлення в діапазоні від 3,3 до 3,7 вольт і споживає до 215 мА, адже клои модуль не отримує достатньо струму, це може спричинити серйозні збої в його роботі.

- незважаючи на те, що Arduino та інші пристрої можуть взаємодіяти з ESP8266 при низьких логічних рівнях 5 вольт, які використовують більшість

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм..	Арк.	№докум.	Підпис	Дата		23

моделей Arduino, для підключення модуля потрібно знизити логічний рівень до 3,3 вольт, щоб уникнути його пошкодження.

– оскільки ESP8266 є популярним і широко використовуваним модулем, існує ризик програмних помилок або нестабільних функцій, тому необхідне регулярне оновлення прошивки та використання стабільних версій бібліотек.

Існує кілька доступних моделей модуля ESP8266, і з усіх них я вирішив обрати модель ESP-07 (рис. 2.4.) [31]. Ця модель має деякі особливості, які відповідають моїм конкретним потребам і вимогам проекту.

Основні причини, чому я вибрав ESP-07:

– має компактний розмір, що дозволяє легко помістити його в обмежений простір або на невелику панель приладів;

– ця модель із зовнішньою антеною, оскільки вона може забезпечити кращий прийом і передачу сигналу Wi-Fi порівняно з моделями з внутрішніми антенами;

– має достатню кількість контактів GPIO, що дозволяє підключати його до зовнішніх пристроїв, датчиків або модулів розширення, що робить його більш гнучким у використанні та розробці проектів;

– економічно вигідний і дозволяє заощадити гроші на вашому проекті без шкоди для якості чи функціональності.

Також важливо зазначити, що вибір моделі ESP-07 може мати свої обмеження. Наприклад, він не має вбудованого USB-порту, який може потребувати додаткового адаптера для програмування та взаємодії. Слід зазначити, що кожна модель ESP8266 має свої особливості та обмеження, тому вибір моделі слід робити відповідно до конкретних потреб і вимог проекту.

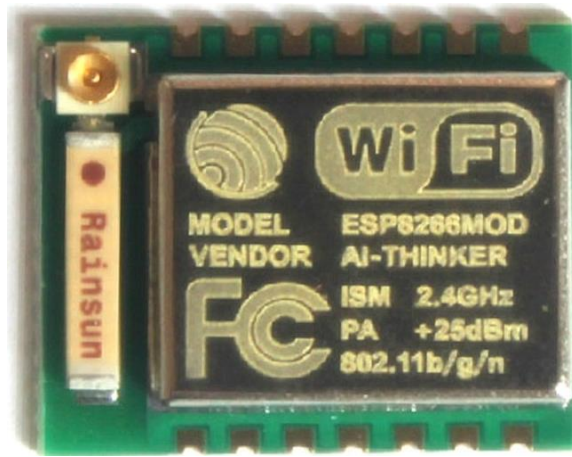


Рисунок 2.4 – ESP8266 ESP-07

Щоб використовувати веб-протокол, необхідно реалізувати на пристрої основні функції веб-сервера, такі як прийом і обробка клієнтських запитів і трансляція відповідного контенту. Для забезпечення роботи в Інтернеті пристрій також повинен мати власну унікальну MAC (Media Access Control) адресу та IP (Internet Protocol) адресу, що дозволить йому взаємодіяти з іншими пристроями в мережі [32, 33].

Щоб використовувати Arduino IDE як середовище розробки для ESP8266, необхідно встановити додатковий пакет, який забезпечує підтримку цього мікроконтролера. Нижче наведено наступні кроки:

- відкрити Arduino IDE;
- у верхньому меню обрати "File" (Файл), а потім "Preferences" (Налаштування);
- у полі "Additional Boards Manager URLs" (Додаткові URL-адреси менеджера плат) додати наступне посилання:
http://arduino.esp8266.com/stable/package_esp8266com_index.json
- натиснути "ОК", щоб зберегти налаштування.
- далі вибрати "Tools" (Інструменти) у верхньому меню, а потім "Board" (Плата);

– у пошуковому полі ввести "esp8266" і знайти пакет "esp8266 by ESP8266 Community" та натиснути кнопку "Install" (Встановити) поряд з ним (рис 2.5);

– після завершення встановлення виберіть "Tools" (Інструменти) у верхньому меню, а потім "Board" (Плата) та прокрутити вниз і знайти "Generic ESP8266 Module" (Загальний модуль ESP8266).

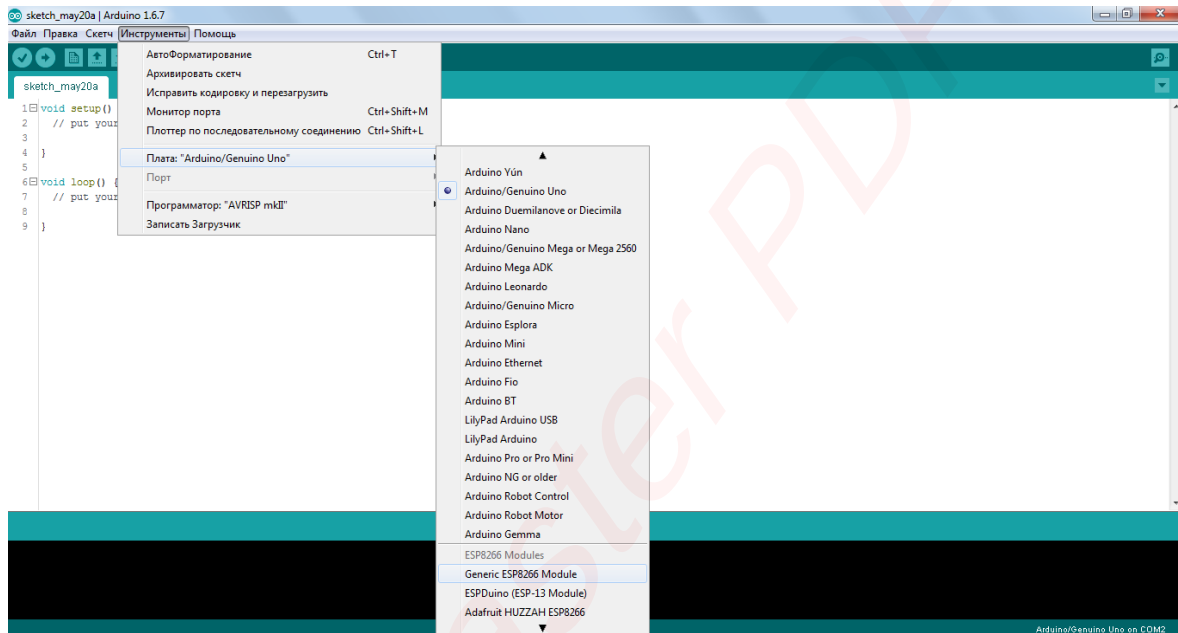


Рисунок 2.5 – Вибір зі списку плат модуля ESP8266

Щоб перепрошити ESP8266 за допомогою Arduino, потрібно виконати наступні кроки [34]:

- з'єднати вихідні піни RX і TX Arduino з відповідними вхідними пінами RX і TX на модулі ESP8266;
- приєднати піни RESET і GND на платі Arduino, щоб уникнути запуску мікроконтролера Arduino;
- з'єднайте масу (GND) на виході Arduino з пінами GND та GPIO15 на модулі ESP8266;
- подати живлення 3.3 В на вхід VCC модуля ESP8266;

– приєднати піни VCC і EN модуля через резистор для виходу модуля з режиму сну.

– приєднати Arduino до комп'ютера за допомогою USB-кабелю.

Ці кроки забезпечують необхідні з'єднання між Arduino та ESP8266 для прошивки модуля за допомогою Arduino IDE.

Для зручності використання мікроконтролера в типових проектах можна використовувати підготовлені двійкові файли, які можна завантажити безпосередньо в ПЗУ модулів, відомі як мікропрограми. Це спрощує процес початку роботи з пристроєм, що розробляється.

Для початку роботи з пристроєм необхідно підключитися до створеної точки доступу та завантажити розроблене програмне забезпечення або використовувати команди через веб-браузер для управління.

Після розробки та тестування пристрою можна проводити подальшу модернізацію та вдосконалення, впроваджуючи додаткові функції на основі існуючих.

2.3 Система керування електродвигунами

Arduino Motor Driver, яка зображена на рисунку 2.6, використовується для керування електродвигунами [14].

Arduino Motor Driver - це модуль або пристрій, призначений для керування рухом моторів з використанням Arduino-плати. Він забезпечує інтерфейс між моторами та Arduino, дозволяючи контролювати їх швидкість, напрямок обертання та інші параметри руху.

Цей модуль забезпечує зручність у підключенні та керуванні моторами. Він забезпечує інтерфейс, який дозволяє легко підключати мотори до плати Arduino і керувати ними за допомогою простого програмування.

Arduino Motor Driver може мати декілька каналів або виходів, що дозволяють керувати багатьма моторами одночасно. Зазвичай він підтримує різні

типи моторів, включаючи постійний струм (DC), кроковий (Stepper) та сервоприводи (Servo).

Модуль може мати вбудовані захисти, такі як захист від перевантаження, короткого замикання та захист від перегріву, що забезпечує безпеку для моторів і самої системи.

Керування моторами здійснюється за допомогою програмування Arduino, де ви можете встановлювати значення швидкості, напрямку руху та інші параметри залежно від вимог вашого проекту.

Зазвичай Arduino Motor Driver підтримує простий інтерфейс програмування, такий як використання певних функцій або бібліотек для керування моторами.

Цей модуль є корисним у багатьох проектах, включаючи робототехніку, автоматизацію, роботи з механізмами, моделювання руху та багато інших додатків, де потрібне точне і гнучке керування моторами.

Використання Arduino Motor Driver спрощує розробку та реалізацію таких проектів, дозволяючи легко і швидко керувати рухом моторів з використанням Arduino-плати.

Плата має можливість керувати двома електродвигунами постійного струму. На платі присутні три перемикача з позначеннями VLO, VLC і VM.

Перемикач VLO впливає на джерело живлення плати. Якщо перемикач увімкнений, для живлення плати використовується вхід VEX, який конвертується до напруги 5 В. Але якщо перемикач вимкнений, для живлення логіки плати використовується вхід 5V.

Перемикач VLC також впливає на джерело живлення плати. Якщо перемикач увімкнений, для живлення плати використовується вхід 5 В з плати Arduino. Якщо ж перемикач вимкнений, для живлення логіки плати використовується вхід 5V.

Перемикач VM визначає джерело живлення для двигунів. Якщо перемикач увімкнений, для живлення двигунів використовується 5 В з плати Arduino. Але

якщо перемикач вимкнений, для живлення двигунів використовується вхід VEX з напругою від 6 до 12 В.

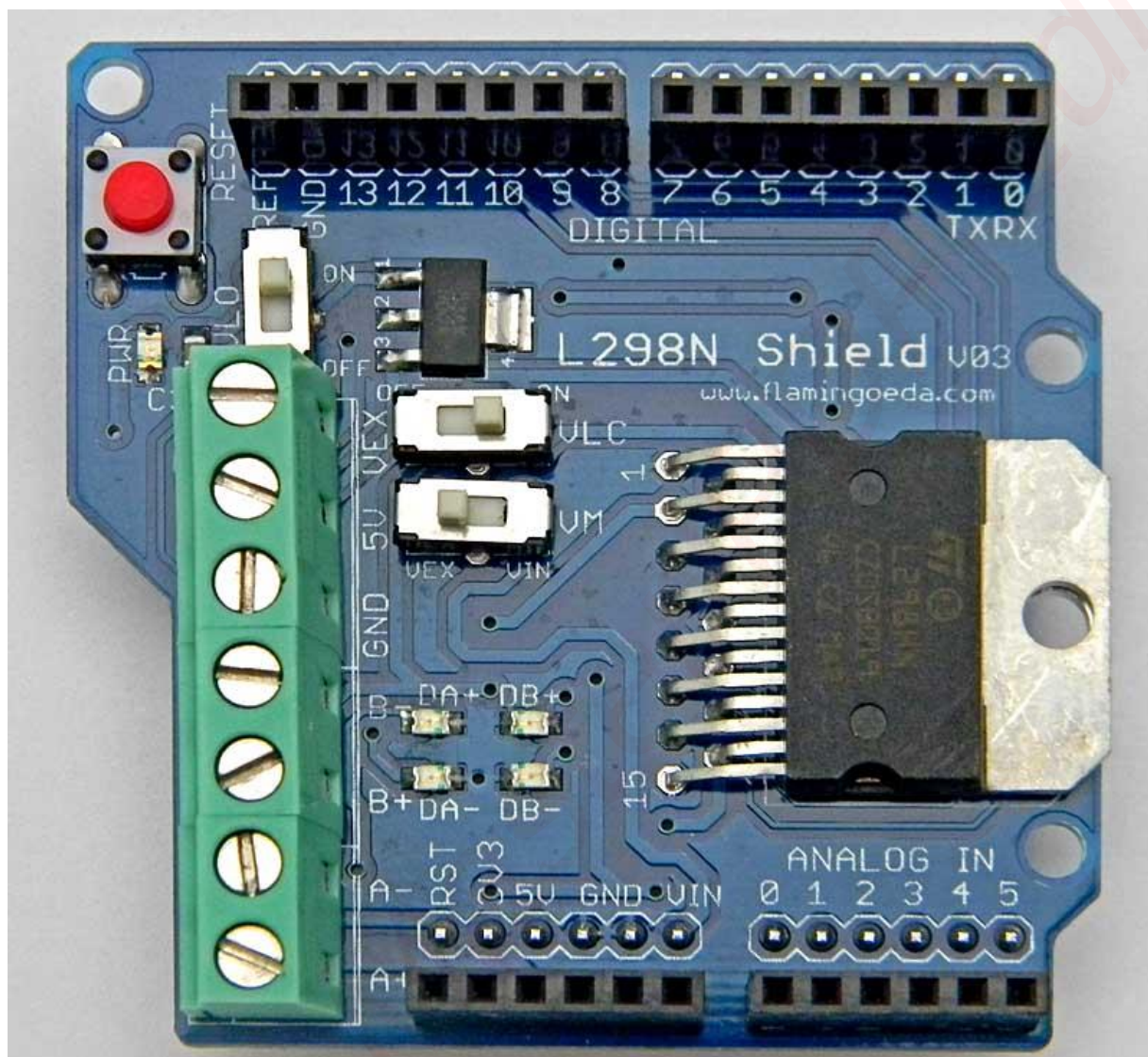


Рисунок 2.6 – Arduino Motor Driver

На платі також розташовані світлодіодні індикатори, які вказують напрямок і швидкість руху для кожного з каналів, а також показують статус живлення.

Зм.	Арк.	№докум.	Підпис	Дата

2.4 Опис основних структурних блоків

Електрична структурна схема була розроблена з урахуванням вимог до роботизованої системи, таких як функціональність, розміри, швидкість, точність тощо. Це допомагає візуалізувати, як система повинна працювати. На початковому етапі проектування роботизованої системи була розроблена узагальнена схема, яка характеризує основні функціональні компоненти системи та зв'язки між ними.

Структурна схема - це засіб опису загальної структури системи, її функціональних компонентів і їх взаємозв'язків. Використовується для ознайомлення з пристроєм і його призначенням. Діаграма надає візуальне представлення того, як система складається з різних компонентів і як вони взаємодіють один з одним для досягнення бажаних функцій. Завдяки структурній схемі можна краще зрозуміти систему, її логіку та залежності між її складовими елементами.

Компоненти, що входять до складу розробленого пристрою, представлені на структурній схемі за допомогою спрощених графічних позначень у вигляді прямокутників довільної форми. У внутрішній частині кожного прямокутника, що відповідає функціональному блоку системи, вказуються короткі назви, що описують призначення кожного блоку.

Вибір конкретних компонентів, які входять до структурної схеми, здійснювався з урахуванням сучасних, ефективних та сучасних компонентів мікроелектроніки, що відповідають вимогам, визначеним у технічному завданні на систему. Завдання, поставлені перед системою, враховуються для забезпечення її ефективної роботи.

Структурна схема призначена для відображення загальної структури пристрою, його основних блоків, вузлів, деталей та основних зв'язків між ними. Завдяки структурній схемі можна зрозуміти, як пристрій функціонує у своїх основних режимах і як його компоненти взаємодіють між собою. При викладенні структурної схеми електричної структури можна використовувати

					КВРКІ. 2102100.19.02.37 ПЗ	Арк.
						30
Зм.	Арк.	№докум.	Підпис	Дата		

довільні позначення для елементів, але слід дотримуватися загальноприйнятих правил виконання схеми [35, 36]. Це допомагає зрозуміти призначення пристрою і забезпечує зручну взаємодію з ним.

Структурна схема наведено на рисунку 2.7 та в Додатку В.

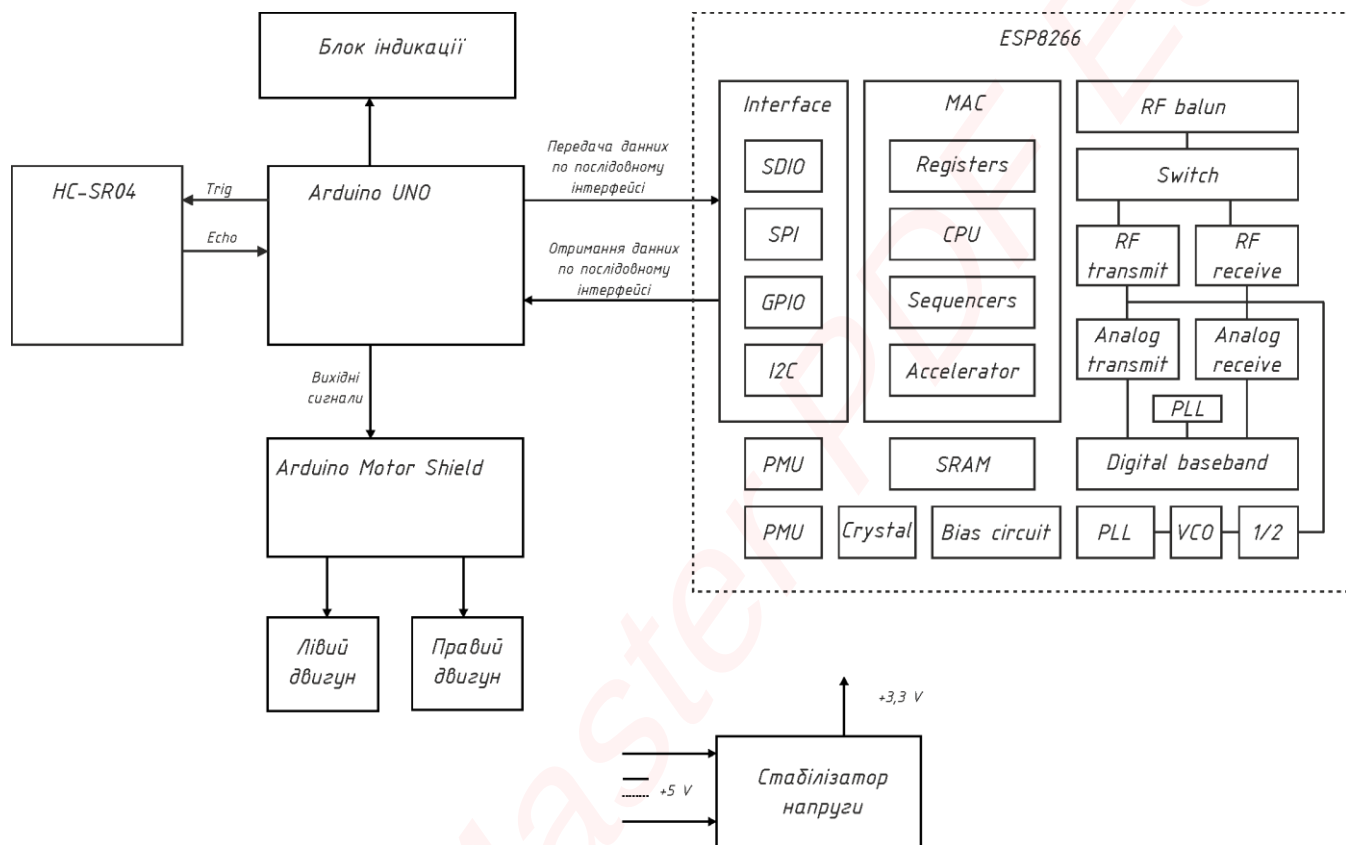


Рисунок 2.7 – Схема електрична структурна робота

На основі визначених функцій і завдань, які були встановлені в технічному завданні, можна виділити ключові модулі системи. З урахуванням цих завдань система розділена на окремі блоки, які виконують свої функції та взаємодіють між собою. Це дозволяє забезпечити ефективну роботу системи та виконання її функціональних вимог.

Поділ системи на модулі допомагає зрозуміти структуру та організацію системи, а також полегшує розробку, тестування та модифікацію окремих компонентів системи. Крім того, такий підхід забезпечує більшу гнучкість і масштабованість системи, якщо її необхідно розширити або модернізувати. Розглянемо основні складові електричної структурної схеми.

Зм.	Арк.	№докум.	Підпис	Дата

Arduino Uno виконує роль пристрою керування. Arduino Uno - платформа для прототипування та розробки електронних проектів. Має вбудований набір цифрових і аналогових входів і виходів, що дозволяє підключати різні датчики, датчики та інші пристрої. Arduino Uno має роз'єми для підключення до комп'ютера через USB-кабель, що дозволяє завантажувати програми на мікроконтролер. Також підтримується можливість розширення за допомогою щитів, що дозволяє ще більше розширити функціональність пристрою [11].

Arduino Uno є популярним вибором для початківців та досвідчених розробників, які бажають швидко прототипувати свої ідеї та створити робочі електронні пристрої. Він призначений для управління всім пристроєм, його основна функція – правильна взаємодія з усіма компонентами системи.

Arduino встановлює зв'язок з модулем Wi-Fi і передає дані через послідовний інтерфейс. Використовуючи цей інтерфейс, Arduino і модуль Wi-Fi взаємодіють для обміну інформацією. Arduino може надсилати команди та дані модулю Wi-Fi, а модуль передає відповіді та результати своєї роботи Arduino [10, 11].

Це забезпечує надійну передачу даних між Arduino і модулем Wi-Fi. Використання такого інтерфейсу спрощує зв'язок між цими пристроями та дозволяє їм обмінюватися інформацією, необхідною для виконання різних завдань, таких як передача даних через мережу Wi-Fi або керування підключеними пристроями.

Цей тип взаємодії дозволяє Arduino використовувати модуль Wi-Fi як засіб для підключення до Інтернету, отримання та надсилання даних з різних джерел і керування пристроями, підключеними до мережі. Послідовний інтерфейс — це спосіб безпечного зв'язку між Arduino та модулем Wi-Fi, що дозволяє їм працювати разом над спільним проектом.

Arduino отримує дані від ультразвукового датчика вимірювання відстані, і, аналізуючи ці дані, передає сигнали для керування дисплеєм і двигунами.

Крім того, Arduino також використовує ці дані для керування двигунами. Залежно від вимірної відстані та запрограмованих умов Arduino може

визначати, коли йти вперед, назад або зупинитися. Він генерує відповідні сигнали для керування двигунами, які виконують відповідні дії на основі вхідних даних.

Таким чином, Arduino використовує дані, отримані від ультразвукового датчика, для керування дисплеєм, який візуалізує виміряні відстані, а також для керування двигунами, які рухаються або зупиняються залежно від вимірних параметрів.

Стабілізатор напруги використовується для зниження напруги з 5 В до 3,3 В, що необхідно для нормальної роботи Wi-Fi модуля.

У цьому випадку для оптимальної роботи модуля Wi-Fi необхідно подати на нього стабільну напругу 3,3 В. Для цього використовується стабілізатор напруги, який приймає початкову напругу 5 В і знижує її до необхідних 3,3 В. Це забезпечує належне живлення модуля Wi-Fi і забезпечує його точну та надійну роботу.

Стабілізатор напруги дозволяє регулювати вихідну напругу до постійного рівня, незалежно від зміни напруги на вхідному джерелі. Він забезпечує стабільну і точну напругу 3,3 В, що є оптимальним для безперебійної роботи Wi-Fi модуля.

На основі сформульованих вимог у третьому розділі вибираються необхідні апаратні компоненти, такі як двигуни, датчики та інші. При цьому враховується сумісність з Arduino і можливості її інтеграції.

2.5 Висновок

Відповідно до завдання на кваліфікаційну роботу необхідно розробити мікроконтролерну систему керування роботом за допомогою модуля ESP8266.

Система управління базуватиметься на платі Arduino UNO.

Доцільність розробки проекту в Arduino з модулем ESP8266 і використанням Arduino Motor Driver обґрунтовується наступними факторами:

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		33

– драйвер двигуна Arduino дозволяє керувати двома двигунами постійного струму, що дає змогу створювати роботизовані системи з рухомими частинами, здатними виконувати різноманітні завдання, такі як переміщення, підйом або поворот;

– модуль ESP8266 забезпечує підключення до бездротової мережі Wi-Fi, що відкриває широкі можливості взаємодії з роботизованою системою через мережу, дистанційне керування, передачу даних, обмін інформацією тощо. Використання ESP8266 разом з Arduino дозволяє реалізувати функції, які вимагають доступу до Інтернету або мережевого зв'язку;

– має просту платформу розробки, зрозумілу багатьом програмістам, що спрощує розробку програмного забезпечення для керування роботизованою системою та інтеграція з модулем ESP8266 дозволяє використовувати широкий набір бібліотек і інструментів для роботи з Wi-Fi та Інтернетом;

– використання контролера двигуна Arduino в поєднанні з модулем ESP8266 дає можливість розширити функціональність системи за допомогою додаткових датчиків, актуаторів і модулів, та створювати більш складні роботизовані системи та пристрої з більш широким набором функцій.

Серед доступних моделей модулів я вибрав ESP-07. Такий вибір обумовлений наявністю роз'єму для зовнішньої антени, що дозволяє збільшити дальність управління. Крім того, модель має металевий екран, який забезпечує захист від пошкоджень, запобігає перешкодам і перегріву.

У другому розділі розроблено структурну електричну схему пристрою. Після детального аналізу основних вузлів схеми можна прийти до висновку, що всі ці модулі необхідні для коректної роботи пристрою та виконання поставлених завдань.

Вивчення структурної схеми дозволяє зрозуміти, як різні компоненти пристрою взаємодіють один з одним для забезпечення його працездатності. Кожен вузол у схемі має свою функцію та призначення, які разом сприяють досягненню цілей пристрою.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		34

Тому ретельний аналіз електричної структурної схеми підтверджує, що кожен вузол є важливим і необхідним для досягнення мети пристрою.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						35
Зм.	Арк.	№докум.	Підпис	Дата		

3 ПРОГРАМНО-АПАРАТНА РЕАЛІЗАЦІЯ ПРИСТРОЮ

3.1 Проектування схеми електричної функціональної системи керування роботом

Схема електрична функціональна пристрою є наступним етапом у розробці електричних схем для системи управління, що включає мікроконтролер [37 - 39]. Вона визначає функціональну роль кожного елемента, розкриваючи природу процесів, що відбуваються в пристрої в цілому та в його окремих функціональних частинах.

На схемі показані функціональні складові пристрою та їх взаємозв'язок. Фактичне розташування елементів і пристроїв на пристрої можна не враховувати. При побудові електричної функціональної принципової схеми слід керуватися послідовністю функціональних процесів, які зображені зліва направо і зверху вниз.

На функціональній схемі пристрою використовуються спеціальні графічні позначення, які описані в стандартах і відображені прямокутниками. Дана схема дозволяє зобразити елементи з'єднання і задати технічні характеристики функціональних частин, параметри в характерних точках і т.д.

Функціональні схеми пояснюють процеси, що відбуваються в окремих функціональних ланцюгах пристрою. Вони використовуються для вивчення принципів роботи виробу, а також під час його налагодження, контролю та ремонту. Оскільки функціональна схема більш детально розкриває функції окремих елементів і пристроїв, вона дає більш повну інформацію про їх роботу порівняно з принциповою схемою.

Електрична функціональна схема є важливим інструментом для розуміння та пояснення роботи пристрою, що дозволяє аналізувати функції окремих компонентів і зв'язки між ними. Це основа для подальшого розвитку та вдосконалення пристрою.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм..	Арк.	№докум.	Підпис	Дата		36

На основі схеми електричної структурної було розроблено схему електричну функціональну, яка наведена на рисунку 3.1 та у Додатку В (рис.3.1).

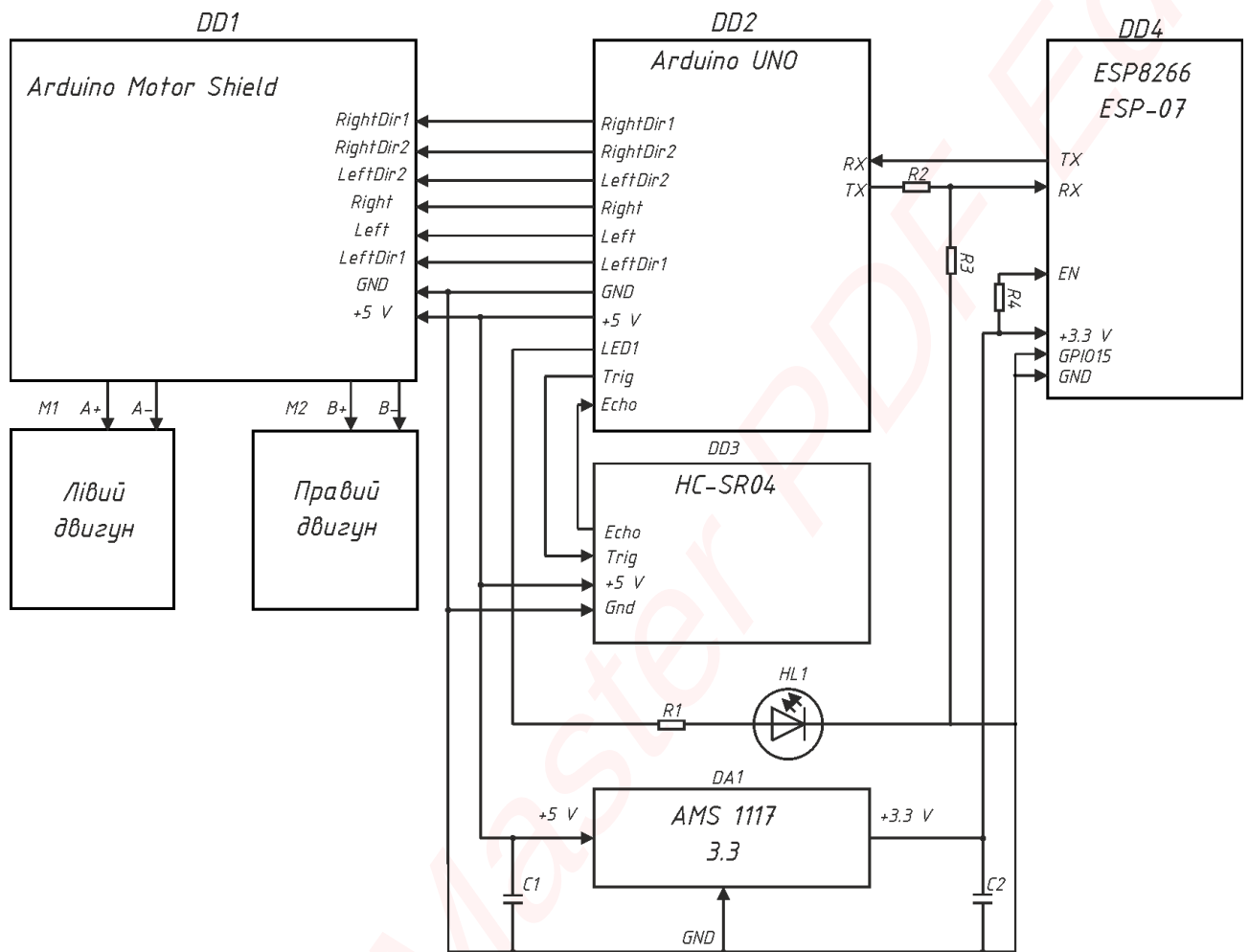


Рисунок 3.1 - Схема електрична функціональна

Розроблюваний пристрій працює згідно наступної схеми: після включення живлення, модуль DA1 виконує функцію перетворення напруги з 5 В, що надходить з виходу DD2 Arduino, на необхідних 3.3 В для модуля ESP8266 (позначений як DD4). Конденсатори C1 і C2 використовуються для стабілізації напруги та згладження перепадів. Резистор R4 застосовується для виведення модуля ESP8266 з режиму сну [40, 41].

ESP8266 створює точку доступу Wi-Fi і чекає підключення клієнта. Коли клієнт підключається до відповідної адреси, ESP8266 передає дані через послідовний порт на Arduino для подальшої обробки.

Зм.	Арк.	№докум.	Підпис	Дата

Ця схема дозволяє забезпечити бездротовий доступ до пристрою через Wi-Fi, забезпечуючи зв'язок між клієнтами і Arduino. Живлення, стабілізація напруги та передача даних між модулями здійснюється за допомогою відповідних компонентів та елементів схеми.

Arduino аналізує отримані дані та виконує відповідні дії, які були закодовані над цими даними. Наприклад:

- передає сигнали до модуля DD1 (Arduino motor shield), що змінює швидкість або напрямок руху двигунів;
- керує подачею або відключенням живлення на світлодіоді HL1. Резистор R1 приблизно 200 Ом перед входом до світлодіоду використовується для обмеження споживання енергії та запобігання перегріву [42];
- генерує імпульс тривалістю 10 мкс на пін Trig сенсора, а потім зчитує імпульс з пина Echo і повертає тривалість імпульсу у мікросекундах. Це дозволяє виміряти відстань до найближчого об'єкта;
- активує автоматичний режим об'їзду перешкод;
- передає дані, наприклад, відстань до найближчого об'єкта назад на модуль ESP8266. Для захисту ESP8266 від 5-вольтової логіки використовуються резистори R2 і R3, які знижують напругу [43].

Розроблена функціональна схема задовольняє поставлені цілі та вимоги проекту. Вона включає необхідні елементи та забезпечує потрібні можливості для досягнення поставлених завдань. Вона враховує особливості та вимоги системи, а також забезпечує правильну взаємодію між її модулями.

3.2 Аналіз схеми електричної принципової

З використанням сучасної елементної бази розроблено простий, але якісний прилад, який має високі метрологічні характеристики з урахуванням наведених схем.

Принципова схема - це повна електрична схема пристрою, яка включає всі електричні елементи та пристрої, необхідні для реалізації та керування

					КВРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		38

зазначеними електричними процесами. Він також включає з'єднання між цими елементами та з'єднувальні елементи, такі як роз'єми та клеми, вхідні та вихідні схеми.

На принциповій схемі використовуються різні елементи, наприклад інтегральні схеми різного рівня інтеграції, а також окремі дискретні елементи. Електричні елементи на схемі зображують умовними графічними позначеннями, креслення та розміри яких встановлені відповідними стандартами.

Усі елементи, які використовуються у виробі та представлені на схемі, повинні бути включені до списку елементів, який може бути розміщений на першому аркуші схеми або представлений окремим документом. Щоб зменшити кількість ліній на схемі та зробити їх коротшими, їх рекомендується об'єднувати в групі зв'язки, так звані «шини». При цьому кожна лінія, що входить в шину, повинна бути позначена в точці їх стику. Позначення можуть бути представлені в цифровій, буквено-цифровій або буквено-цифровій формі.

Принципова електрична схема, наведена на рисунку 3.2 та в додатку В. Використання електричної принципової схеми не тільки дозволяє отримати повне уявлення про проект, але і надає можливість розробляти індивідуальні схеми підключення та вузли з'єднання.

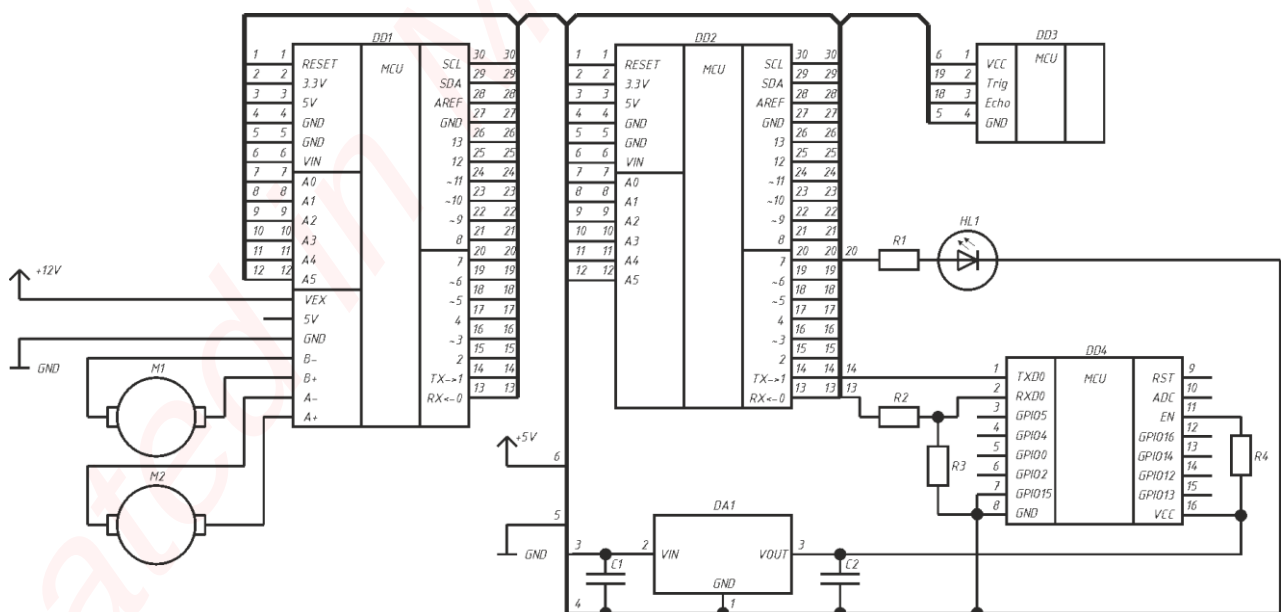


Рисунок 3.2 – Схема електрична принципова системи

Зм..	Арк.	№докум.	Підпис	Дата

Ця схема є основою для перевірки правильності монтажу електрообладнання і допомагає забезпечити його належну функціональність. Завдяки електричній принциповій схемі можна перевірити відповідність розподілу з'єднань і елементів проекту та впевнитися в правильному з'єднанні всіх компонентів системи.

Розглянемо детальніше підключення основних елементів.

Керуюча частина системи отримує живлення від джерела напруги 5 В. Також можна подавати напругу в діапазоні від 7 до 12 В на плату Arduino, яка автоматично перетворює її до необхідних 5 В. Щоб забезпечити живлення, необхідно вставити штекер адаптера (діаметр - 2.1 мм, позитивний контакт - центральний) в роз'єм живлення на платі. У випадку живлення від 5 В, його слід підключити до виходів Gnd і Vin роз'єму POWER [44].

Для живлення двигунів рекомендується використовувати джерело напруги від 9 до 12 В. Це може бути акумулятор або інше компактне джерело. Для живлення модуля ESP8266 використовується стабілізатор напруги AMS1117, який зображений на рисунку 3.3. Він перетворює вхідну напругу 5 В на необхідні 3.3 В для живлення модуля [50].

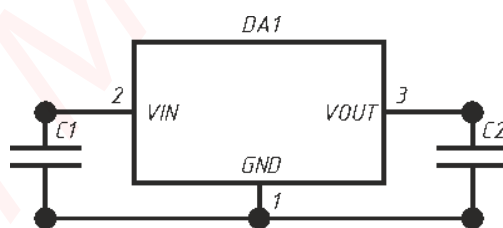


Рисунок 3.3 – Стабілізатор напруги AMS 1117

Ця схема включення є типовим прикладом застосування стабілізатора AMS1117 [51]. Конденсатор С1 використовується для згладження імпульсів вхідної напруги, тобто він допомагає забезпечити стабільну та рівномірну напругу на вході стабілізатора. Конденсатор С2 використовується для фільтрації можливих завад, які можуть впливати на роботу мікросхеми. Цей конденсатор

допомагає зменшити шуми та перешкоди, що можуть виникати під час роботи стабілізатора.

Електричні параметри стабілізатора AMS1117 наступні:

- вхідна напруга (Input Voltage) 4.5 В - 12 В;
- вихідна напруга (Output Voltage) - 3.3 В;
- максимальний струм (Maximum Current) - це максимальний допустимий струм, який може пройти через стабілізатор: до 1 Ам;
- точність вихідної напруги (Output Voltage Accuracy) - це відхилення вихідної напруги від заданого значення: 1%;
- температурний діапазон (Temperature Range) - діапазон температур, при якому стабілізатор може надати відповідні електричні характеристики: від -40°C до +125°C.

Arduino виступає як головний керуючий елемент, отримуючи команди через послідовний порт і виконуючи їх. На рисунку наведена схема виходів Arduino, яка показує доступні порти та їх позначення (рис. 3.4).

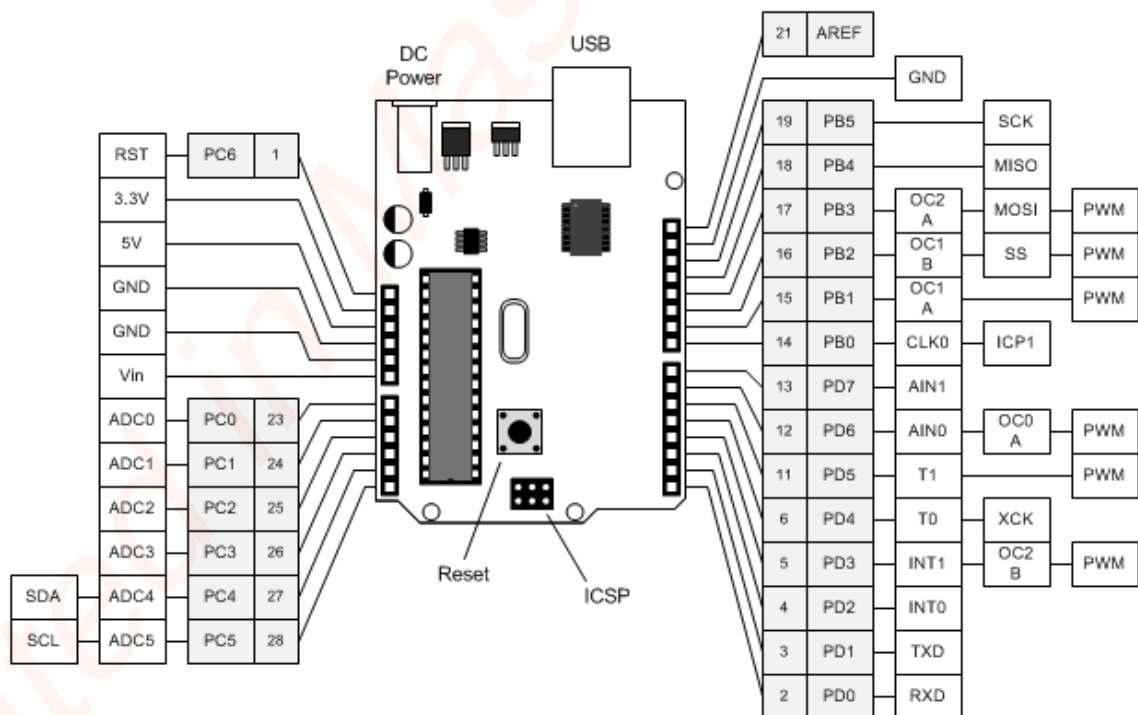


Рисунок 3.4 – Схема виходів Arduino

Arduino Uno має 14 цифрових входів, які можуть діяти як входи або виходи за допомогою функцій `pinMode()`, `digitalWrite()` і `digitalRead()`. Кожен із цих входів/виходів обмежено 5 В. Один вихід може бути джерелом або споживати до 40 мА. Усі цифрові виходи мають внутрішні резистори від 20 до 50 кОм, які за замовчуванням відключені.

Деякі виходи Arduino можуть виконувати додаткові функції на додаток до звичайного введення та виведення цифрових сигналів. Ось декілька прикладів:

- ШІМ (PWM) вихід. Виходи виходи 3, 5, 6, 9, 10 і 11 Arduino можуть генерувати широтно-імпульсну модуляцію для керування інтенсивністю світла світлодіодів, швидкістю обертання моторів або регулювання аналогових сигналів;

- деякі цифрові входи Arduino можуть працювати в якості аналогових входів, що дозволяє зчитувати значення аналогових сигналів, таких як сигнали від датчиків або потенціометрів;

- виходи Arduino можуть бути використані для керування зовнішніми пристроями, такими як сервоприводи, реле або електронні ключі. Це дозволяє вам забезпечити управління різними пристроями у нашому проекті;

- комунікація з іншими пристроями. виходи 11 (MOSI), 12 (MISO), 13 (SCK), 10 (SS) Arduino можуть бути використані для встановлення зв'язку з іншими пристроями через різні протоколи, такі як I2C, SPI або UART, що дозволяє обмінюватися даними з сенсорами, дисплеями, бездротовими модулями та іншими пристроями.

Також, на 7 виході Arduino під'єднаний світлодіод HL1, який можна використовувати для індикації роботи пристрою. Щоб запобігти згоранню світлодіода, до нього під'єднаний резистор R1 з опором приблизно 200 Ом.

У моделі Arduino Uno є 6 аналогових входів позначених як A0 до A5, кожен з яких може надати аналогове значення напруги у вигляді 10-бітного числа (1024 різних значень). За замовчуванням, вимірювання напруги здійснюється в діапазоні від 0 до 5 В. Проте, верхню межу цього діапазону

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм..	Арк.	№докум.	Підпис	Дата		42

можна змінити, використовуючи вихід AREF (аналоговий опорний вихід) та функцію `analogReference ()`.

За допомогою функції `analogReference ()`, можна встановити зовнішнє джерело напруги або внутрішній опорний струм, що дозволить розширити діапазон вимірювання аналогових сигналів. Це корисно, наприклад, при підключенні більш точних аналогових сенсорів або при роботі з великими значеннями напруги.

Для програмування Arduino Uno використовується програмне забезпечення Arduino IDE. Для початку, у програмі Arduino IDE потрібно обрати відповідну плату, зазначивши "Arduino Uno" в меню Tools > Board, що відповідає мікроконтролеру вашої плати.

Arduino Uno має мікроконтролер ATmega328, який вже має завантажений загрузчик. Це означає, що ви можете завантажувати нові програми без використання зовнішнього програматора. Взаємодія з мікроконтролером здійснюється за допомогою оригінального протоколу STK500.

Проте, також є можливість прошивки мікроконтролера через роз'єм для внутрішньосхемного програмування ICSP (In-Circuit Serial Programming) без використання загрузчика.

Це дозволяє вам вибрати найзручніший спосіб програмування залежно від вашого завдання та доступних ресурсів.

До плати Arduino підключена плата-розширення Arduino Motor Driver, яка дозволяє керувати двома електродвигунами постійного току. На платі розширення розташовані три перемикачі з наступними налаштуваннями:

– перемикач VLO, якщо перемикач включений, для живлення плати використовується вхід VEX, якщо перемикач вимкнений, для живлення логіки плати використовується вхід 5V;

– перемикач VLC, якщо перемикач включений, для живлення плати використовується вхід з плати, якщо перемикач вимкнений, для живлення логіки плати використовується вхід 5V;

– перемикач VM, якщо перемикач включений, для живлення двигунів використовується 5 В з плати Arduino, якщо перемикач вимкнений, для живлення двигунів використовується джерело напруги в діапазоні від 6 до 12 В, підключене до входу VEX [51].

Ці перемикачі надають можливість регулювати джерела живлення та відповідно налаштовувати роботу плати-розширення Arduino Motor Driver залежно від потреб проекту.

На схемі видно, що двигуни підключені до плати Arduino через плату-розширення Motor Driver. Живлення подається до плати Arduino та Motor Driver. Також є три перемикача - VLO, VLC та VM, які впливають на живлення плати та електродвигунів в залежності від їх положення (рис.3.5).

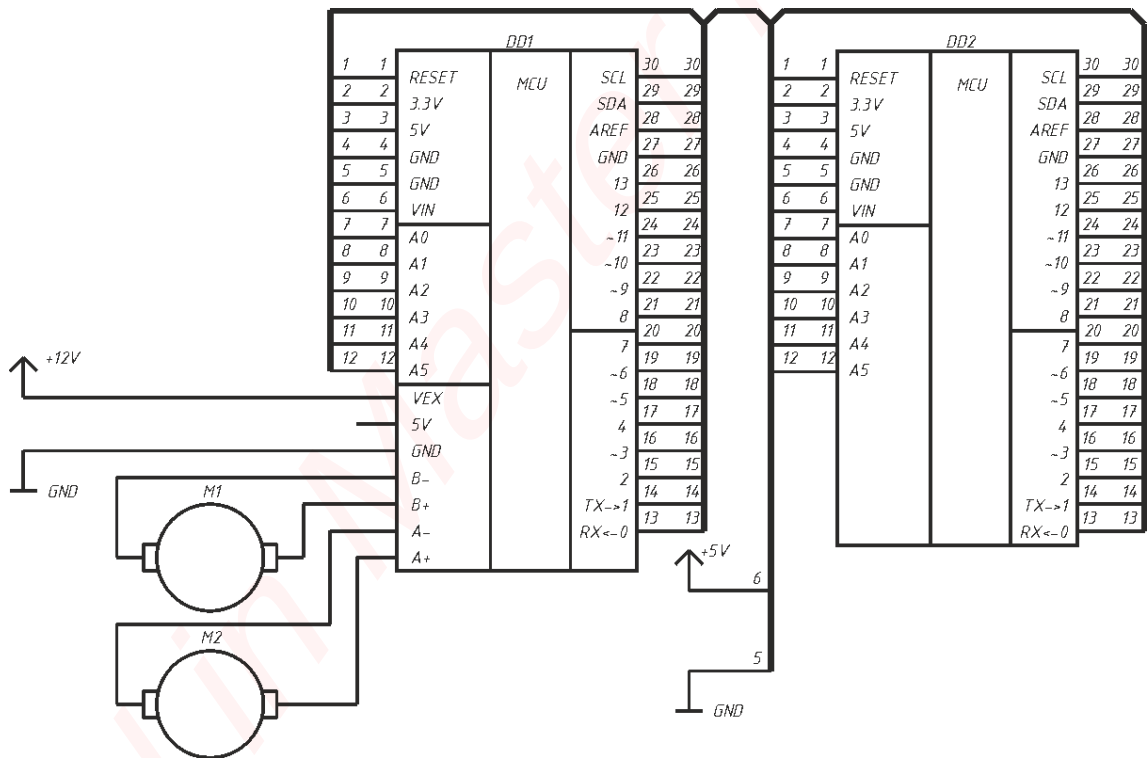


Рисунок 3.5 – Схема підключення модуля Arduino Motor Driver

Далекомір HC-SR04 [52] є популярним ультразвуковим датчиком відстані, який використовується для вимірювання відстані до об'єкта без контакту. Основним принципом роботи цього датчика є відправлення ультразвукового сигналу і приймання його ехо для визначення відстані (рис. 3.6).



Рисунок 3.6 – датчик вимірювання відстані HC-SR04

HC-SR04 складається з двох основних компонентів: передавача (трансмiттера) і приймача (рецептора). Передавач генерує короткий ультразвуковий імпульс, який направляє до цільового об'єкта. Цей ультразвуковий імпульс відбивається від об'єкта і сприймається приймачем. Далекомір вимірює час, який потрібний для того, щоб ультразвуковий сигнал пройшов туди й назад, і на основі цього обчислює відстань до об'єкта [53]

Основні характеристики далекоміра HC-SR04:

- діапазон вимірювання: від 2 сантиметрів до 4 метрів;
- висока точність: ± 3 міліметри.
- широка напругова сумісність: від 5 В, що робить його сумісним з багатьма мікроконтролерами та платами розвитку;
- простота використання ;
- малий розмір;
- 4 виходи;
- може бути використаний для вимірювання швидкості об'єктів, визначення напрямку руху та інших додаткових функцій залежно від програмного забезпечення, що використовується.

Загалом, далекомір HC-SR04 є надійним і простим у використанні датчиком, який знаходить широке застосування в багатьох проектах, які потребують безконтактного вимірювання відстані або виявлення об'єктів.

підключити вихід Trig датчика до виходу 6 плати Arduino і підключити вихід Echo датчика до входу 5 плати Arduino.

Модуль WiFi ESP8266 ESP-07 — це потужний вбудований пристрій бездротового зв'язку, який можна використовувати для підключення до Інтернету. Він базується на мікроконтролері ESP8266, який працює на основі архітектури Tensilica Xtensa LX106 з тактовою частотою 80 МГц (рис 3.8).

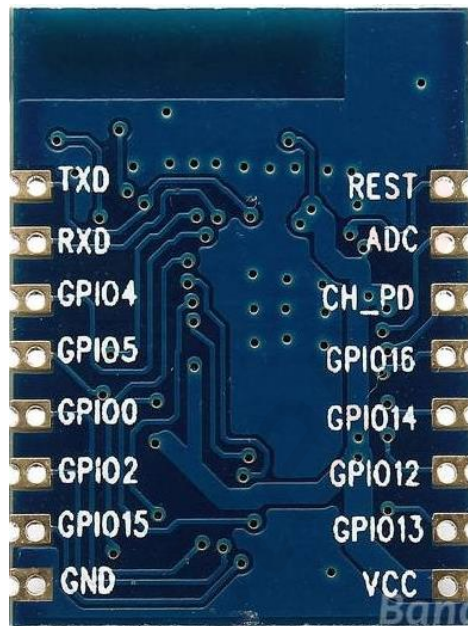


Рисунок 3.8 - Виходи ESP8266 ESP-07

ESP-07 має вбудований модуль WiFi, який підтримує стандарти безпроводового зв'язку IEEE 802.11b/g/n. Це дозволяє модулю здійснювати підключення до доступних WiFi мереж і обмін даними з Інтернетом [54].

Модуль ESP-07 має вбудовану антену, яка забезпечує добре зв'язку і приймання сигналу WiFi. Також на модулі є роз'єм для підключення зовнішньої антени, що може покращити якість зв'язку у випадках, коли сигнал слабкий.

ESP-07 має вбудовану пам'ять Flash для зберігання програмного забезпечення та даних. Цей модуль підтримує виконання коду зовнішньої пам'яті, що дозволяє збільшити обсяг програмного забезпечення, яке можна виконати.

Для програмування модуля ESP-07 використовуються різні середовища розробки, такі як Arduino IDE, PlatformIO, Lua та інші. Модуль має можливість спілкування через послідовний порт (UART), що дозволяє передавати команди та отримувати дані від модуля.

Для успішного запуску чіпу ESP-07 необхідно подати живлення і з'єднати VCC та EN за допомогою резистора R4 10 кОм. У випадку відсутності такого резистора, можна використати будь-який резистор з діапазону 1-20 кОм. Крім цього, саме модель ESP-07 вимагає заземлення GPIO15 (найближчого до GND). Для зниження напруги 5 В з логіки Arduino використовуються два резистори: R2 1 кОм та R4 2 кОм. На рисунку 3.9 зображена схема підключення.

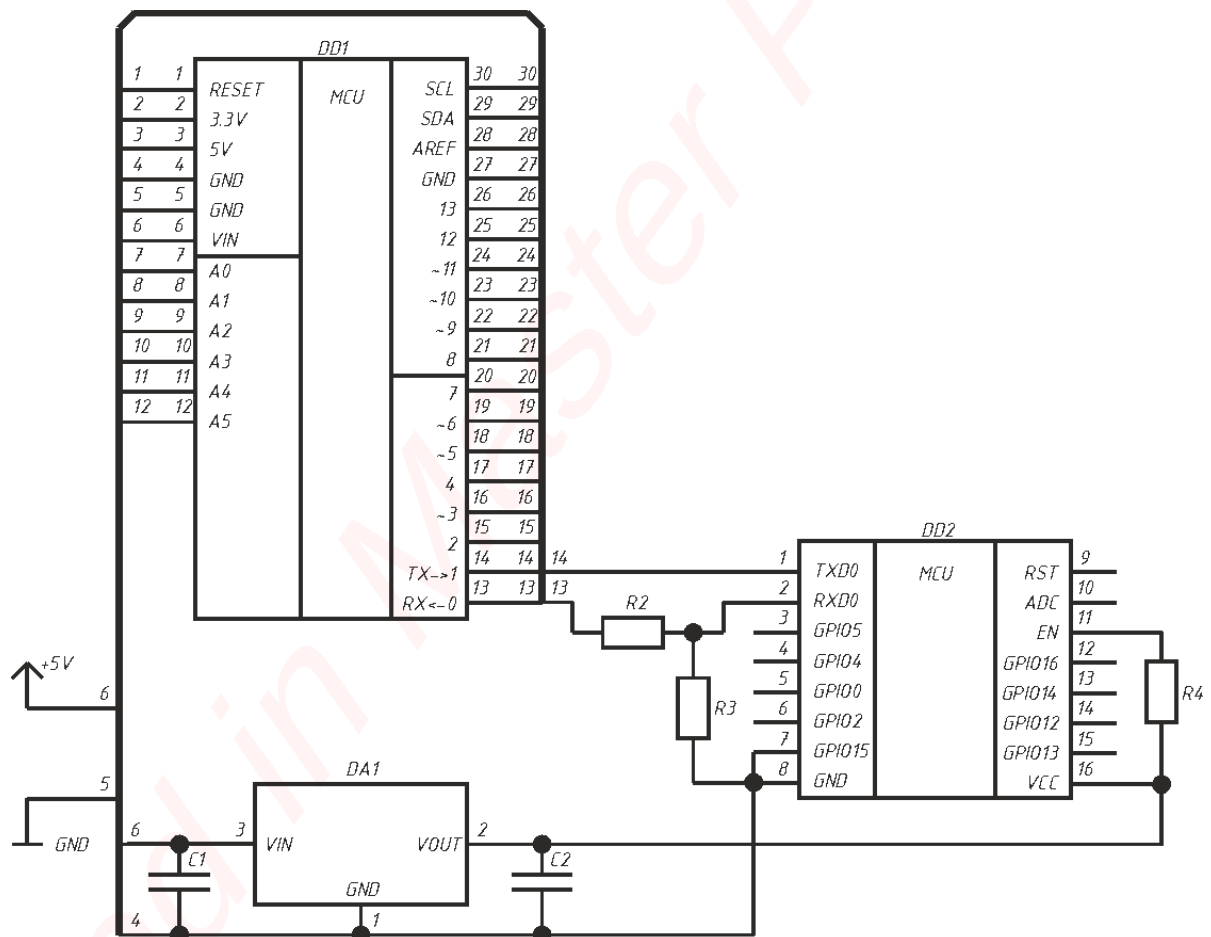


Рисунок 3.9 – Схема під'єднання модуля ESP8266 до плати Arduino

Згідно з схемою, Arduino і ESP8266 з'єднуються через послідовний порт для передачі команд. Для живлення ESP8266 використовується понижувач

напруги, а саме модулю під'єднані VCC і EN через резистор R4, що дозволяє вивести модуль з режиму сну.

3.3 Опис алгоритму роботи

Після подачі живлення система працює за наступною схемою:

- ESP8266 створює точку доступу Wi-Fi для з'єднання з користувачем;
- ESP8266 очікує підключення користувача до точки доступу;
- після підключення, ESP8266 очікує передачі даних від клієнта;
- коли клієнт переходить за вказаною веб-адресою, ESP8266 реєструє це і передає команду по послідовному порту на Arduino;
- Arduino отримує команду з послідовного порту у форматі ASCII коду, і аналізує отримані дані і виконує відповідні дії залежно від команди.

Наприклад, команди можуть включати або вимикати світлодіод, змінювати напрям і швидкість руху електродвигунів, зчитувати дані про відстань до найближчого об'єкта, або вмикати та вимикати автоматичний режим.

Приклади команд:

- "A0 " - команда для зупинки пристрою;
- "L2 " - команда для зміни швидкості лівого двигуна на максимальну;
- "R-1 " - команда для зміни напрямку і швидкості руху правого двигуна на половинну;
- "AUTOON" - команда для ввімкнення автоматичного режиму;
- "SON" - команда для ввімкнення світлодіода;
- "A-2 " - команда для зміни напрямку і швидкості руху двох двигунів на максимальну;
- "DIS " - команда для запиту та передачі відстані до найближчого об'єкта.

Алгоритм роботи пристрою зображений на рисунку 3.10

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм..	Арк.	№докум.	Підпис	Дата		49

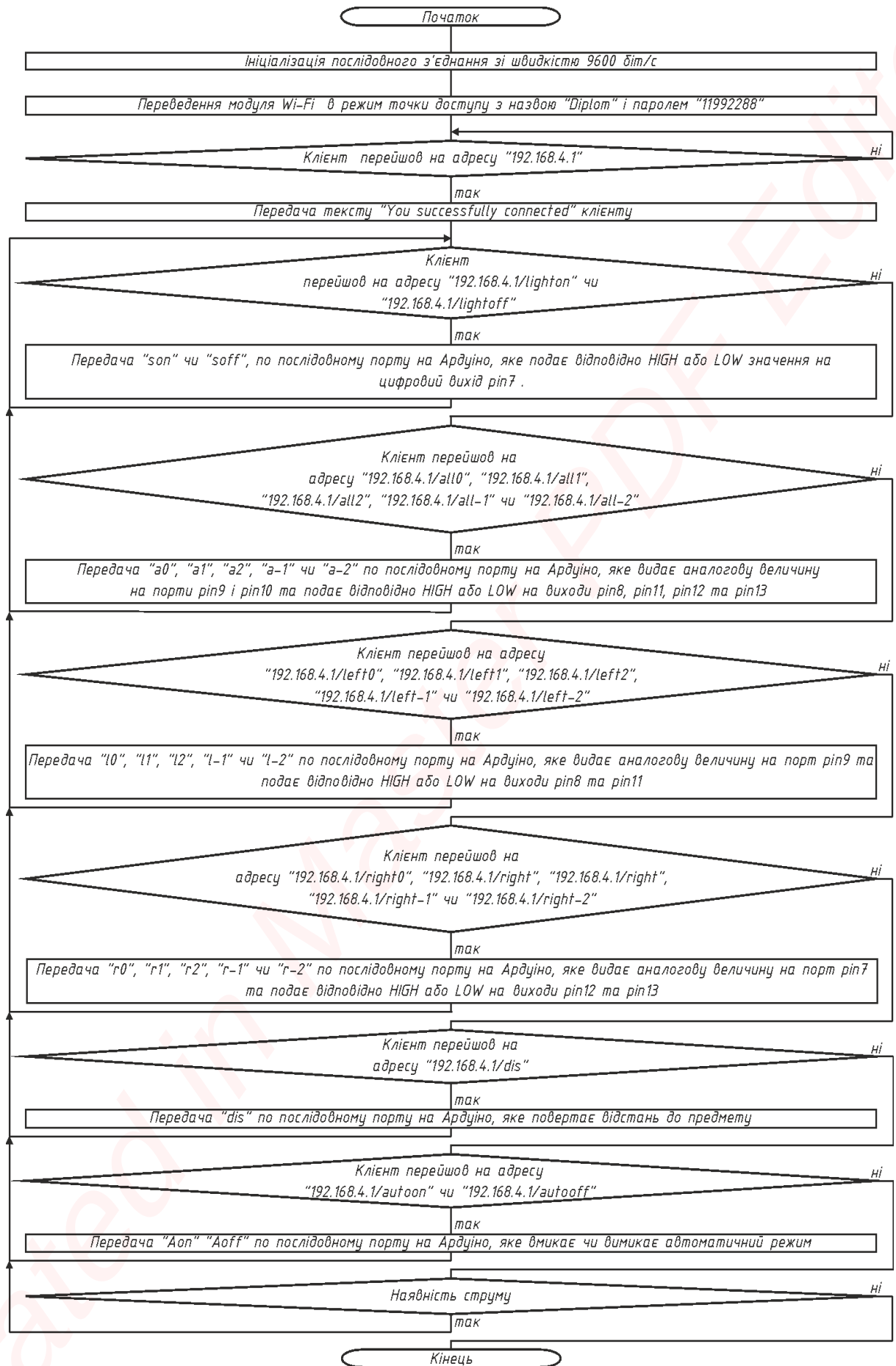


Рисунок 3.11 – Узагальнений алгоритм роботи

Зм.	Арк.	№докум.	Підпис	Дата

Алгоритм роботи керуючого пристрою зображено на рисунку 3.12.

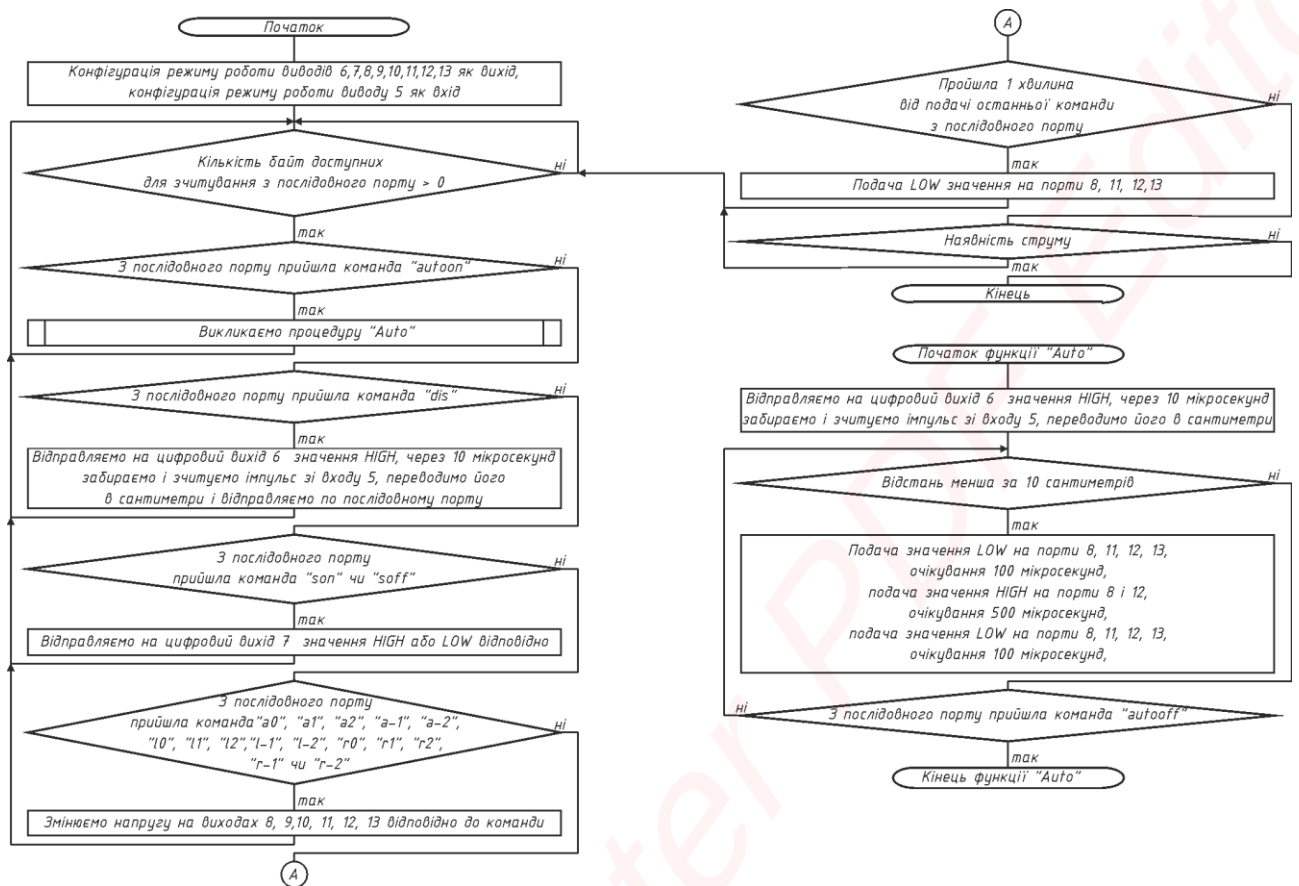


Рисунок 3.12 – Алгоритм роботи Arduino

Як видно на схемі, Arduino отримує команди через послідовний порт у форматі ASCII кодів. Наприклад, якщо ESP8266 відправляє на послідовний порт команду "a1", на Arduino будуть отримані дві послідовні цифри - 97 і 49. Таким чином, коли Arduino отримує цю послідовність цифр, вона виконує команду - передає напругу на електродвигуни на половину від їх потужності.

У Arduino присутня функція автоматичного режиму, яка дозволяє роботу об'їжджати перешкоди. Цей режим активується, коли на послідовний порт Arduino надсилається команда "autoon". Після отримання цієї команди починає виконуватись процедура "Auto".

Для відстеження часу, що пройшов з моменту отримання останньої команди, використовується функція millis(). Ця функція повертає кількість мілісекунд, які пройшли з початку виконання поточної програми на Arduino. Значення цього лічильника обнуляється при переповненні, яке відбувається

приблизно через 50 днів. За допомогою функції `millis()` ми відстежуємо час, що пройшов з отримання останньої команди, і якщо цей час перевищує одну хвилину, то зупиняємо рух двигунів. Це зроблено з метою того, щоб у разі, наприклад, виходу робота за межі зони дії або відсутності отримання команд від користувача, робот не продовжував рухатись безконтрольно.

3.4 Розробка програмного забезпечення

Для розробки програми на платформі Android я використовував середовище візуальної розробки додатків - App Inventor [44, 45].

App Inventor використовує графічний інтерфейс користувача, що дозволяє програмувати за допомогою візуальних блоків. Для компіляції візуальних блоків в байт-код Android використовується компілятор, який базується на фреймворку GNU і мові Кава для реалізації динамічних мов [57, 58].

На рисунку 3.13 зображено інтерфейс візуального блокового програмування в середовищі App Inventor.



Рисунок 3.13 –Середовище App Inventor

На рисунку 3.14 представлено створення візуальної частини програми.

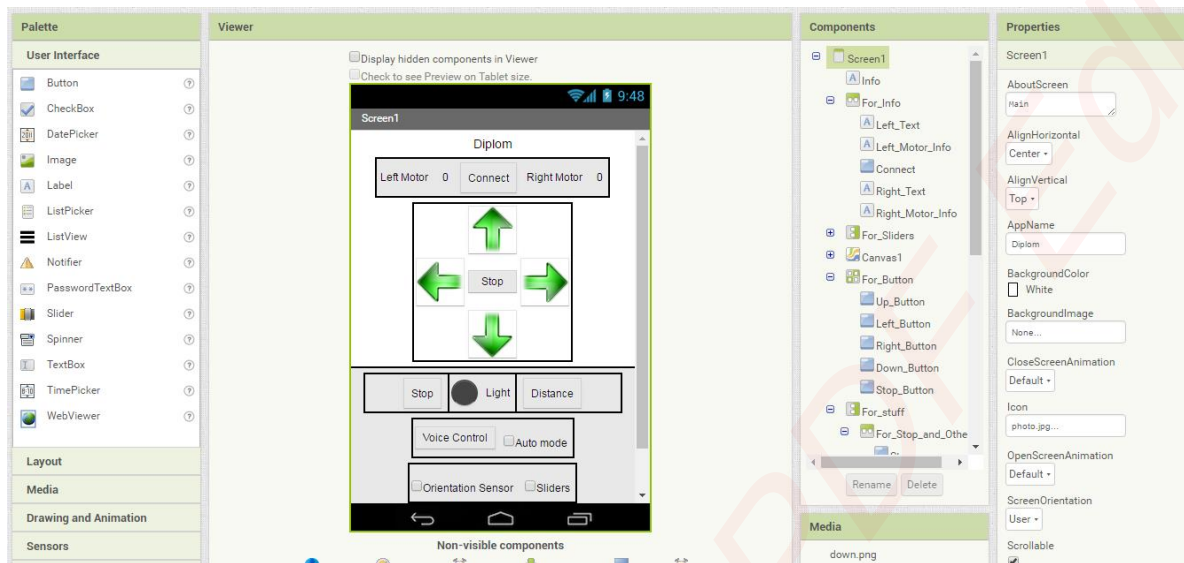


Рисунок 3.14 – Візуальна частина програми

На цьому рисунку можна побачити основні елементи керування програмою, які відображені на екрані. Інформація щодо швидкості руху двигунів також відображена. Для керування напрямком руху використовуються стрілки. Кнопка "Exit" призначена для виходу з програми, кнопка "Connect" використовується для під'єднання до сервера. Кнопка "Stop" призначена для зупинки руху. Кнопка "Voice Control" дозволяє керувати програмою за допомогою голосу. Кнопка "Light" використовується для подачі живлення на світлодіод.

На рисунку 3.15 представлено фрагмент коду яка відповідає за зупинку пристрою.

На цьому рисунку видно, що при натисканні на кнопку "Stop" програма виконує перехід за адресою <http://192.168.4.1/all0>. Модуль ESP8266 фіксує цю подію і передає команду "a0" на Arduino. Arduino зчитує цю команду і зупиняє роботу двигунів. Крім того, в програмі є змінна "global wait", яка дозволяє захистити модуль ESP8266 від надмірної передачі даних [46].

Зм..	Арк.	№докум.	Підпис	Дата

```

to Stop
do
  if
    get global wait = false
  then
    set global wait to true
    set Wifi . Url to "http://192.168.4.1/all0"
    call Wifi . Get
    set global am to 0
    set global lm to 0
    set global rm to 0
    set Left_Motor_Power . Text to 0
    set Left_Motor_Power . Text to 0
    set Left_Motor_Info . Text to 0
    set Right_Motor_Info . Text to 0
    set All_Motor . ThumbPosition to 0
    set Left_Motor . ThumbPosition to 0
    set Right_Motor . ThumbPosition to 0
    set Up . Visible to false
    set Down . Visible to false
    set Left . Visible to false
    set Right . Visible to false
    set Info . Text to "you stoping."
    set Info . Text to "you stoping.."
    set Info . Text to "you stoping..."
    set Info . Text to "you stopped"
  else
    set Info . Text to "Sorry, wait or connect again"

```

Рисунок 3.15 – Фрагмент коду для зупинки пристрою

3.5 Розрахунок електричних параметрів

Оскільки споживана потужність пасивних елементів, таких як резистори, конденсатори і діоди, не є значною, то можна не враховувати їх у споживаній потужності. Для розрахунку сумарної потужності розроблюваного пристрою будемо враховувати лише споживану потужність активних елементів.

Для розрахунку сумарної споживчої потужності розпишемо потужність $P_{\text{спож}}$, що споживається усіма активними елементами за формулою:

$$P_{\text{спож}} = I_{\text{спож}} \cdot U_{\text{спож}} \quad (3.1)$$

Для визначення сумарної потужності розроблюваного пристрою, проведемо розрахунок потужності, споживаної активними елементами згідно формули 3.1. Почнемо з розрахунку потужності керуючих елементів: Arduino витрачає приблизно 20 мА при напрузі 5 В. Таким чином, споживана потужність Arduino:

$$P_{\text{Arduino}} = 0.02 * 5 = 0.1, \text{ Ватт}$$

Потужність споживання ESP8266 :

$$P_{\text{ESP8266}} = 0.215 * 3.3 = 0.7, \text{ Ватт}$$

Тепер розрахуємо загальну потужність для керуючих елементів:

$$P_{\text{operating}} = P_{\text{Arduino}} + P_{\text{ESP8266}} = 0.1 + 0.7 = 0.8, \text{ Ватт}$$

Аналогічно розрахуємо потужність виконуваних елементів в робочому стані. Потужність споживання Arduino Motor Driver:

$$P_{\text{Arduino Motor Driver}} = 0.36 * 5 = 1.8, \text{ Ватт}$$

Потужність споживання HC-SR04:

$$P_{\text{HC-SR04}} = 0.015 * 5 = 0.075, \text{ Ватт}$$

Отже потужність споживання двигунів:

$$P_{\text{motors}} = (1 * 12) * 2 = 24, \text{ Ватт}$$

Тепер розрахуємо загальну потужність для виконуючих елементів в робочому стані:

$$P_{executing} = P_{HC-SR04} + P_{motors} + P_{LED} + P_{Arduino\ Motor\ Driver} = 0.075 + 24 + 0.125 + 1.8 = 26, \text{ Wamm}$$

Отже сумарна потужність споживання в робочому стані:

$$P_{all\ action} = P_{operating} + P_{executing} = 0.8 + 26 = 26.8, \text{ Wamm}$$

Отже, можна зробити висновок, що прилад споживає значну кількість енергії при максимальному навантаженні всіх його елементів, що може швидко розрядити навіть потужні акумулятори. Однак, варто врахувати, що пристрій не завжди працює на максимальній потужності, і в режимі очікування він може працювати досить тривалий час завдяки низькому споживанню енергії. Для даного пристрою використовувалися низькопотужні двигуни.

3.6 Висновок

На основі запропонованої вище електричної структурної схеми, відповідно до вимог проектної документації, розроблено електричну функціональну схему. Ця схема детально пояснює режими роботи модулів підсистеми мікроконтролера, які відповідають за управління роботою системи керування, а також взаємодію її компонентів з мікроконтролером і між собою.

Однак варто відзначити, що опис електричної функціональної схеми дає більш детальну інформацію про принципи загальної будови системи мікроконтролера. Опис принципової електричної схеми має характер довідкової інформації, оскільки вже детально описано у функціональній схемі. Крім того, розроблено програмне забезпечення, яке значно розширює функціональність системи та коло завдань. Програма розроблена в середовищі Arduino.

Повідомлення про можливі помилки відображаються після компіляції та компіляції програми. Для роботи з дошкою використовуються готові бібліотеки, багато з яких на даний момент доступні в Інтернеті, при необхідності можна додавати додаткові бібліотеки.

Також розроблено програму керування для платформи Android з використанням середовища візуальної розробки додатків - App Inventor

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
						57
Зм..	Арк.	№докум.	Підпис	Дата		

ВИСНОВКИ

В результаті виконання моєї дипломної роботи було успішно вирішено реальну практичну задачу в області комп'ютерної інженерії, а саме розробку системи дистанційного керування роботом, який працює на базі контролера Arduino і керується Android через бездротову лінію зв'язку.

Актуальність даної теми полягала в задачі створення унікального та інноваційного пристрою, який був би економічно доступним. Тому для розробки був використаний новий модуль Wi-Fi - ESP8266, який забезпечує всі функції Wi-Fi і має низьку вартість. На основі цього модуля створено точку доступу Wi-Fi і веб-сервер, до якого клієнт може підключатися і відправляти дані. ESP8266 отримує ці дані та передає їх на плату Arduino через послідовний порт. Arduino обробляє отримані дані і виконує відповідні команди.

Даний пристрій має широкий спектр застосування, як в побутовій та господарській сфері для дистанційного керування побутовою технікою, так і в промисловості для управління виробничими процесами та електрообладнанням. Крім того, його можна використовувати в комплексі з іншими системами та засобами автоматизації виробничих процесів, що гарантує більш ефективне та автоматизоване управління виробництвом.

Для досягнення поставленої мети були проведені основні етапи проектування цифрових пристроїв, включаючи розробку структурної схеми, проектування функціональної схеми, вибір базового елемента та розробку електричної принципової схеми пристрою. Крім того, проведено необхідні розрахунки елементів основної схеми.

В процесі роботи над проектом для нашого пристрою була розроблена програма, яка дозволяє йому працювати з максимальною ефективністю. Отримані технічні характеристики пристрою відповідають поставленому призначенню та вимогам кваліфікаційної роботи.

					КвРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм..	Арк.	№докум.	Підпис	Дата		58

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Вікіпедія. Wi-Fi. URL: <https://uk.wikipedia.org/wiki/Wi-Fi> (дата звернення: 15.01.2023).
2. Паламар М.І., Стрембіцький М.О., Паламар А.М. Проєктування комп'ютеризованих вимірювальних систем і комплексів: навч. посіб. Тернопіль: ТНТУ. 2019. 150 с.
3. Голубєв Л. П., Чумак Б. І. Дистанційне керування роботом-маніпулятором. *Технології та дизайн*. Харків, 2019. - № 2 (31).
4. Carroll S., Kalaitzakis M., Vitzilaios N. UAS Sensor Deployment to the Underside of Structures. *In 2021 International Conference on Unmanned Aircraft Systems (ICUAS)*. 2021. Pp. 895-900.
5. Приходько В.М. Комп'ютерна схемотехніка: навч. посіб. Харків : ХНЕУ, 2017. 300с.
6. Харченко В. С. Інтернет речей для індустриальних і гуманітарних застосунків : навч. посіб. У трьох томах. Том 1. Основи і технології. Харків : ХАІ, 2019. – 547 с.
7. Giernacki W., Horla D., Saska M. In-flight Efficient Controller Auto-tuning using a Pair of UAVs. *In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2020. Pp. 1300-1307.
8. Белякова І. В., Ларіна К. І. Сучасні технології в освітленні, парків, скверів, зелених насаджень. *Актуальні задачі сучасних технологій*, : зб. тез доп. IV міжнар. наук.-практ. конф., м. Запоріжжя, 3-4 берез. 2016 р. Тернопіль, 2015. Том 2. С. 87-88.
9. Офіційний сайт Arduino. URL: <http://www.arduino.cc> (дата звернення: 16.02.2023).
10. Вікіпедія / Arduino. URL: <https://uk.wikipedia.org/wiki/Arduino> (дата звернення: 21.03.2023).
11. Клімат у домі. URL: http://www.klimatvdomi.com/index_heat_ua.html (дата звернення: 13.03.2023).

					КВРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		59

12. Опалення: практичний посібник із пошуку оптимальної системи опалення. URL: <https://uk.parisynovcampus.com/717-heating-practical-guide-to-finding-your-heating-system> (дата звернення: 14.04.2023).

13. Arduino Duemilanove URL: <http://www.arduino.cc/en/Main/ArduinoBoardDuemilanove> (дата звернення: 28.02.2023).

14. Blog nyl.io / Wifi Controlled Motors using ESP8266 and ATtiny85. URL: <http://blog.nyl.io/esp8266-motor/> (дата звернення: 18.02.2023).

15. Arduino - Info /Arduino L298N Motor Drive Shield. URL:arduino-info.wikispaces.com/ArduinoShieldL298N (дата звернення: 12.04.2023).

16. Geektimes / Збираєм Wi-Fi робота. URL: <https://geektimes.ru/post/255448/> (дата звернення: 11.03.2023).

17. AESP8266.io URL: <http://esp8266.ru/modules-esp8266/> (дата звернення: 28.04.2023).

18. ESP8266 Community Forum URL: <http://www.esp8266.com/index.php> (дата звернення: 15.02.2023).

19. Android controlled robot: the best arduino tutorial. URL: <http://diyhacking.com/make-android-controlled-robot/> (дата звернення: 28.03.2023).

20. Hou A. S., Lin C. E. Uas Delivery Multi-Rotor Robot Based on Ardu-Pilot Framework Using S-Bus Protocol. *In 2021 Integrated Communication, Navigation and Surveillance Conference (ICNS)*. 2022. Pp. 1-10.

21. García J., Molyna J. M., & Trincado J. Real evaluation for designing robot fusion in UAV platforms. *Information Fusion*. 2020. Pp. 136-152.

22. Palamar A. Intelligent monitoring and control 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.

23. Mariya Yao *Applied Artificial Intelligence: A Handbook For Business Leaders* Kindle Edition, Publisher: TOPBOTS, 2018, 246 p.

					КВРКІ. 2102100.19.02.37 ПЗ	Арк.
						60
Зм.	Арк.	№докум.	Підпис	Дата		

24. Ormsbe L. Optimal Scheduling For Robot Distribution Systems. *Proceedings of the 4th International Scheduling Conference: Theory and Applications (MISTA 2019)*, 10-12 Aug 2019. Dublin, Ireland, 2019. P. 341-356.

25. Flora Mercader-Trejo, Maria Guadalupe López Granada, Luz Elena Narváez Hernández, *Industrial Metrology Engineering: An Educational Strategy to Fulfill the Needs of Industry and Society NCSLI Measure : The Journal of Measurement Science*. 2013. Volume 8, Issue 4. P. 28-30.

26. Совин Я. Р., Наконечний Ю. М. Аналіз апаратної підтримки криптографії в пристроях інтернету речей. *Ukrainian Scientific Journal of Information Security*. 2018. Том 24, № 1 (2018). С. 36-48.

27. Cotryna G. Gaussian Pseudorandom Number Generator Based on Linear Feedback Shift Registers. *Sensors (Basel, Switzerland)*. 2020, P.1-18.

28. Лорія М. Г. Цифрова схемотехніка операції : навч. посіб. Запоріжжя : Сєверодонецьк: Вид-во Східноукр. НУ ім. Володимира Даля, 2016. 280 с.

29. Моделювання та аналіз цифрових схем: підручник. Рівне: НУВГП, 2018. 463 с.

30. ESP8266 Community Forum / ESP8266 Community Forum. URL: <http://www.esp8266.com/index.php>, (дата звернення: 18.03.2023).

31. ESP8266 esp-01 wi-fi модуль Arduino. URL: <https://arduinoomania.in.ua/esp8266-esp-01-wi-fi-modul-ua> (дата звернення: 19.04.2023).

32. Interface L298N DC Motor Driver Module with Arduino. URL: <https://lastminuteengineers.com/l298n-dc-stepper-driver-arduino-tutorial/> (дата звернення: 19.04.2023).

33. Гапак О. М., Болога С.І. Захист інформації в комп'ютерних системах : підручник. Ужгород : ДВНЗ «Ужгородський національний університет», 2021. 184 с.

34. Козюрата В.Д. Захист інформації в комп'ютерних системах: підручник. Ніжин: ТПК «Орхідея», 2020. 236 с.

35. Національні (ДСТУ), міждержавні (ГОСТ) і міжнародні (ISO, IEC)

					КВРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		61

стандарти; керівні нормативні документи України; законодавчі документи
Европейського Союзу (ECD). URL:
<http://www.leonorm.com.ua/Default.php?rescode=0510&Page=stcatalog> (дата
звернення: 15.03.2023).

36. Конструкторська документація. URL:
https://www.wikiwand.com/uk/Конструкторська_документація. (дата звернення:
16.04.2023).

37. Приходько В.М. Комп'ютерна схемотехніка : навч. посіб. Харків : Вид-
во ХНЕУ. 2017. 300с.

38. Макаренко В. В. Цифрова і імпульсна схемотехніка. Моделювання та
аналіз : навч. посіб. Київ : НТУУ "КПІ". 2015. 312 с.

39. Нестерчук Д. М. Методи та засоби вимірювань електричних та
неелектричних величин: навчальний посібник Мелітополь: Виданвико-
поліграфічний центр «Люкс». 2017. 206 с.

40. Ультразвуковий датчик відстані HC-SR04 URL:
<https://arduino.ua/prod182-yltrazvykovoii-datchik-rasstoyaniya-hc-sr04> (дата
звернення: 18.03.2023).

41. Robocraft / Ультразвуковий датчик виміру відстані HC-SR04. URL:
robocraft.ru/blog/electronics/772.html (дата звернення: 18.03.2023).

42. Wi-Fi модуль ESP8266 ESP-07 URL: [https://diylab.com.ua/p182639812-
modul-esp8266-esp.html](https://diylab.com.ua/p182639812-modul-esp8266-esp.html) (дата звернення: 19.03.2023).

43. Колонтаєвський Ю. П. Комп'ютерна електроніка: навчальний посібник
Харків: ХНУМГ ім. О. М. Бекетова, 2019. – 156 с

44. MIT App Inventor URL: <https://appinventor.mit.edu/> (дата звернення:
24.04.2023).

45. Революція в інтернет речах. URL:
<https://habrahabr.ru/company/coolrf/blog/235881/> (дата звернення: 16.04.2023)

46. Робототехніка. URL: <https://uk.wikipedia.org/wiki/Робототехніка>. (дата
звернення: 28.04.2023)

					КВРКІ. 2102100.19.02.37 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		62

ДОДАТОК А

(обов'язковий)

Лістинг програми на Ардуіно

```
int incomingByte = 0;
#define Trig 6
#define Echo 5
const int leftdir1 = 11;
const int leftdir2 = 8;
const int left = 9;
const int rightdir1 = 13;
const int rightdir2 = 12;
const int right = 10;
const int light = 7;
unsigned int time_us = 0;
unsigned int distance_sm = 0;
const long interval = 60000;
unsigned long previousMillis=0;
void setup()
{
  pinMode(Trig, OUTPUT);
  pinMode(Echo, INPUT);
  pinMode(leftdir1, OUTPUT);
  pinMode(rightdir1, OUTPUT);
  pinMode(leftdir2, OUTPUT);
  pinMode(rightdir2, OUTPUT);
  pinMode(left, OUTPUT);
  pinMode(right, OUTPUT);
  pinMode(light, OUTPUT);
  analogWrite(left, 0);
  analogWrite(right, 0);
  analogWrite(leftdir1, 0);
  analogWrite(rightdir1, 0);
  analogWrite(leftdir2, 1);
  analogWrite(rightdir2, 1);
  analogWrite(light, 0);
  Serial.begin(9600);
  delay(3000);
}
void Auto()
{
```

```

analogWrite(left, 255);
analogWrite(right, 255);
while (1)
{
  digitalWrite(Trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(Trig, LOW);
  time_us = pulseIn(Echo, HIGH);
  distance_sm = time_us / 58;
  if (distance_sm < 10) //
  {
    digitalWrite (leftdir1, LOW);
    digitalWrite (leftdir2, LOW);
    digitalWrite (rightdir1, LOW);
    digitalWrite (rightdir2, LOW);
    delay(100);
    digitalWrite (leftdir1, LOW);
    digitalWrite (leftdir2, HIGH);
    digitalWrite (rightdir1, HIGH);
    digitalWrite (rightdir2, LOW);
    delay(500);
    digitalWrite (leftdir1, LOW);
    digitalWrite (leftdir2, LOW);
    digitalWrite (rightdir1, LOW);
    digitalWrite (rightdir2, LOW);
    delay(100);
  }
  else
  {
    digitalWrite (leftdir1, LOW);
    digitalWrite (leftdir2, HIGH);
    digitalWrite (rightdir1, LOW);
    digitalWrite (rightdir2, HIGH);
  }
  delay(100);
  if (Serial.available() > 0)
  {
    incomingByte = Serial.read();
    if (incomingByte == 65)
    {
      delay(10);
      incomingByte = Serial.read();
      if (incomingByte == 111)

```

```

        {
            delay(10);
            incomingByte = Serial.read();
            if (incomingByte == 102)
            {
                break;
            }
        }
    }
}

void loop()
{
    if (Serial.available() > 0)
    {
        /******auto*****
        incomingByte = Serial.read();
        if (incomingByte == 65)
        {
            delay(10);
            incomingByte = Serial.read();
            if (incomingByte == 111)
            {
                delay(10);
                incomingByte = Serial.read();
                if (incomingByte == 110)
                {
                    Auto();
                    analogWrite(left, 0);
                    analogWrite(right, 0);
                }
            }
        }
    }

    /******distance*****
    if (incomingByte == 100)
    {
        delay(10);
        incomingByte = Serial.read();
        if (incomingByte == 105)
        {

```

```

    delay(10);
    incomingByte = Serial.read();
    if (incomingByte == 115)
    {
        digitalWrite(Trig, HIGH);
        delayMicroseconds(10);
        digitalWrite(Trig, LOW);
        time_us = pulseIn(Echo, HIGH);
        distance_sm = time_us / 58;
        Serial.println(distance_sm);
        previousMillis = millis();
    }
}

//*****light*****
if (incomingByte == 115)
{
    delay(10);
    incomingByte = Serial.read();
    delay(10);
    incomingByte = Serial.read();
    if (incomingByte == 110) digitalWrite(light, 1);
    else if (incomingByte == 102) digitalWrite(light, 0);
    previousMillis = millis();
}

//*****all motor*****
if (incomingByte == 97)
{
    delay(10);
    incomingByte = Serial.read();
    if (incomingByte == 45)
    {
        digitalWrite(leftdir1, 1);
        digitalWrite(rightdir1, 1);
        digitalWrite(leftdir2, 0);
        digitalWrite(rightdir2, 0);
        delay(10);
        incomingByte = Serial.read();
    }
    else
    {
        digitalWrite(leftdir1, 0);

```

```

    digitalWrite(rightdir1, 0);
    digitalWrite(leftdir2, 1);
    digitalWrite(rightdir2, 1);
}
if (incomingByte == 48)
{
    analogWrite(left, 0);
    analogWrite(right, 0);
}
if (incomingByte == 49)
{
    analogWrite(left, 150);
    analogWrite(right, 150);
}
if (incomingByte == 50)
{
    analogWrite(left, 255);
    analogWrite(right, 255);
}
previousMillis = millis();
}
//*****left motor*****
if (incomingByte == 108)
{
    delay(10);
    incomingByte = Serial.read();
    if (incomingByte == 45)
    {
        digitalWrite(leftdir1, 1);
        digitalWrite(leftdir2, 0);
        delay(10);
        incomingByte = Serial.read();
    }
    else
    {
        digitalWrite(leftdir1, 0);
        digitalWrite(leftdir2, 1);
    }
}
if (incomingByte == 48)
{
    analogWrite(left, 0);
}
if (incomingByte == 49)

```

```

    {
        analogWrite(left, 150);
    }
    if (incomingByte == 50)
    {
        analogWrite(left, 255);
    }
    previousMillis = millis();
}
//*****right motor*****
if (incomingByte == 114)
{
    delay(10);
    incomingByte = Serial.read();
    if (incomingByte == 45)
    {
        digitalWrite(rightdir1, 1);
        digitalWrite(rightdir2, 0);
        delay(10);
        incomingByte = Serial.read();
    }
    else
    {
        digitalWrite(rightdir1, 0);
        digitalWrite(rightdir2, 1);
    }
    if (incomingByte == 48)
    {
        analogWrite(right, 0);
    }
    if (incomingByte == 49)
    {
        analogWrite(right, 150);
    }
    if (incomingByte == 50)
    {
        analogWrite(right, 255);
    }
    previousMillis = millis();
}
}
unsigned long currentMillis = millis();
if (currentMillis - previousMillis >= interval)

```

```
{  
    digitalWrite(rightdir1, 0);  
    digitalWrite(rightdir2, 0);  
    digitalWrite(leftdir1, 0);  
    digitalWrite(leftdir2, 0);  
}  
}
```

ДОДАТОК Б

(обов'язковий)

Лістинг програми на ESP8266

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
const char *ssid = "Diplom";
const char *pass = "11992288";
String incomingByte = "0";
ESP8266WebServer server(80);
void setup()
{ Serial.begin(9600);
  delay(1000);
  Serial.println("starting");
  delay(1000);
  WiFi.softAP(ssid, pass);
  delay(1000);
  Serial.println("started");
  //IPAddress myIP = WiFi.softAPIP();
  //*****Main Page*****
  server.on("/", [] () {
    server.send(200, "text/plain", "You successfully connected");
    Serial.println("start"); });
  //*****Light*****
  server.on("/lighton", [] () { //http://192.168.4.1/lighton
    server.send(200, "text/plain", "Light is on now.");
    Serial.println("son"); });
  server.on("/lightoff", [] () { //http://192.168.4.1/lightoff
    server.send(200, "text/plain", "Light is off now.");
    Serial.println("soff");
  });
  //*****0 speed*****
  server.on("/left0", [] () { //http://192.168.4.1/left0
    Serial.println("l0");
    server.send(200, "text/html", "OK left=0");
  });
  server.on("/right0", [] () { //http://192.168.4.1/right0
    Serial.println("r0");
```

```

server.send(200, "text/html", "OK right=0");
});
server.on("/all0", []() { //http://192.168.4.1/all0
Serial.println("a0");
server.send(200, "text/html", "OK all=0");
});
//*****1 speed*****
server.on("/left1", []() { //http://192.168.4.1/left1
Serial.println("l1");
server.send(200, "text/html", "OK left=1");
});
server.on("/right1", []() { //http://192.168.4.1/right1
Serial.println("r1");
server.send(200, "text/html", "OK right=1");
});
server.on("/all1", []() { //http://192.168.4.1/all1
Serial.println("a1");
server.send(200, "text/html", "OK all=1");
});
//*****-1 speed*****
server.on("/left-1", []() { //http://192.168.4.1/left-1
Serial.println("l-1");
server.send(200, "text/html", "OK left=-1");
});
server.on("/right-1", []() { //http://192.168.4.1/right-1
Serial.println("r-1");
server.send(200, "text/html", "OK right=-1");
});
server.on("/all-1", []() { //http://192.168.4.1/all-1
Serial.println("a-1");
server.send(200, "text/html", "OK all=-1");
});
//*****2 speed*****
server.on("/left2", []() { //http://192.168.4.1/left2
Serial.println("l2");
server.send(200, "text/html", "OK left=2");
});
server.on("/right2", []() { //http://192.168.4.1/right2
Serial.println("r2");
server.send(200, "text/html", "OK right=2");
});
server.on("/all2", []() { //http://192.168.4.1/all2
Serial.println("a2");

```

```

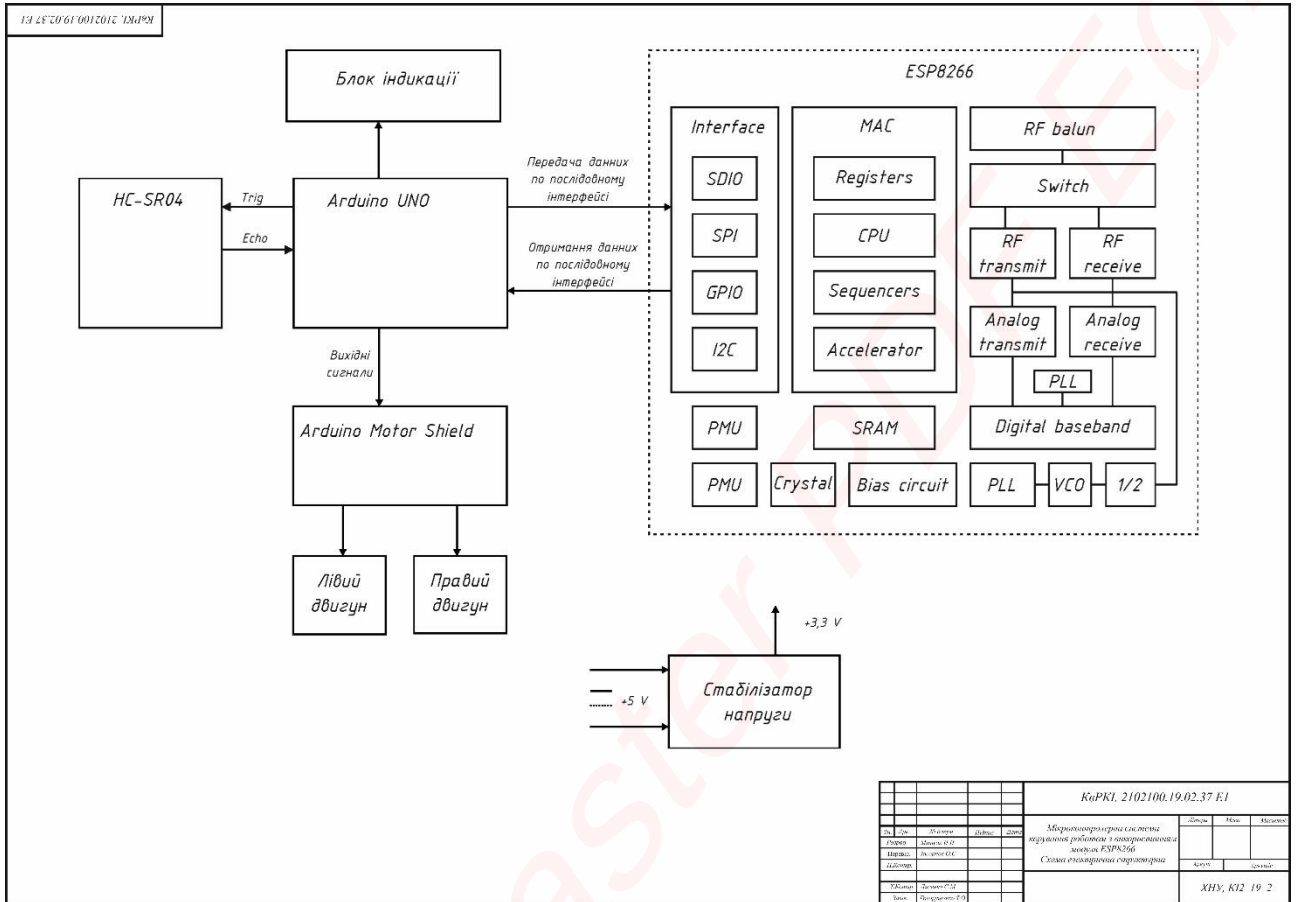
        server.send(200, "text/html", "OK all=2");
    });
    //*****-2 speed*****
    server.on("/left-2", []() { //http://192.168.4.1/left-2
        Serial.println("l-2");
        server.send(200, "text/html", "OK left=-2");
    });
    server.on("/right-2", []() { //http://192.168.4.1/right-2
        Serial.println("r-2");
        server.send(200, "text/html", "OK right=-2");
    });
    server.on("/all-2", []() { //http://192.168.4.1/all-2
        Serial.println("a-2");
        server.send(200, "text/html", "OK all=-2");
    });
    //*****distance*****
    server.on("/dis", []() { //http://192.168.4.1/dis
        Serial.println("dis");
        delay(1000);
        incomingByte = "0";
        while (Serial.available() > 0)
            incomingByte += char(Serial.read());
        incomingByte = "distance = " + incomingByte + " cm";
        server.send(200, "text/html", incomingByte);
    });
    //*****Auto mode*****
    server.on("/autoon", []() { //http://192.168.4.1/autoon
        Serial.println("Aon");
        server.send(200, "text/html", "Auto mode enabled");
    });
    server.on("/autooff", []() { //http://192.168.4.1/autooff
        Serial.println("Aoff");
        server.send(200, "text/html", "Auto mode disabled");
    });
    server.begin(); }
void loop()
{ server.handleClient();}

```

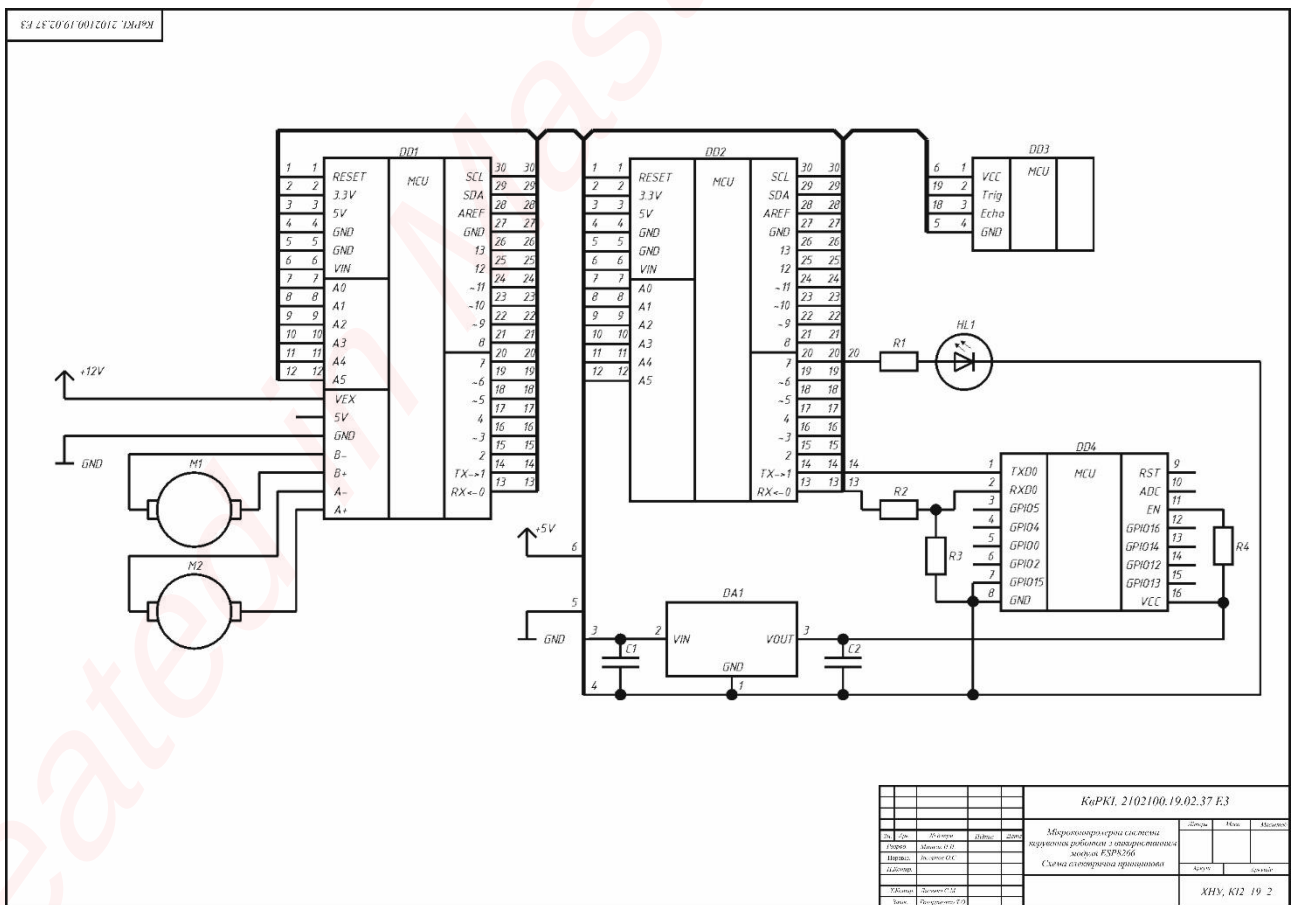
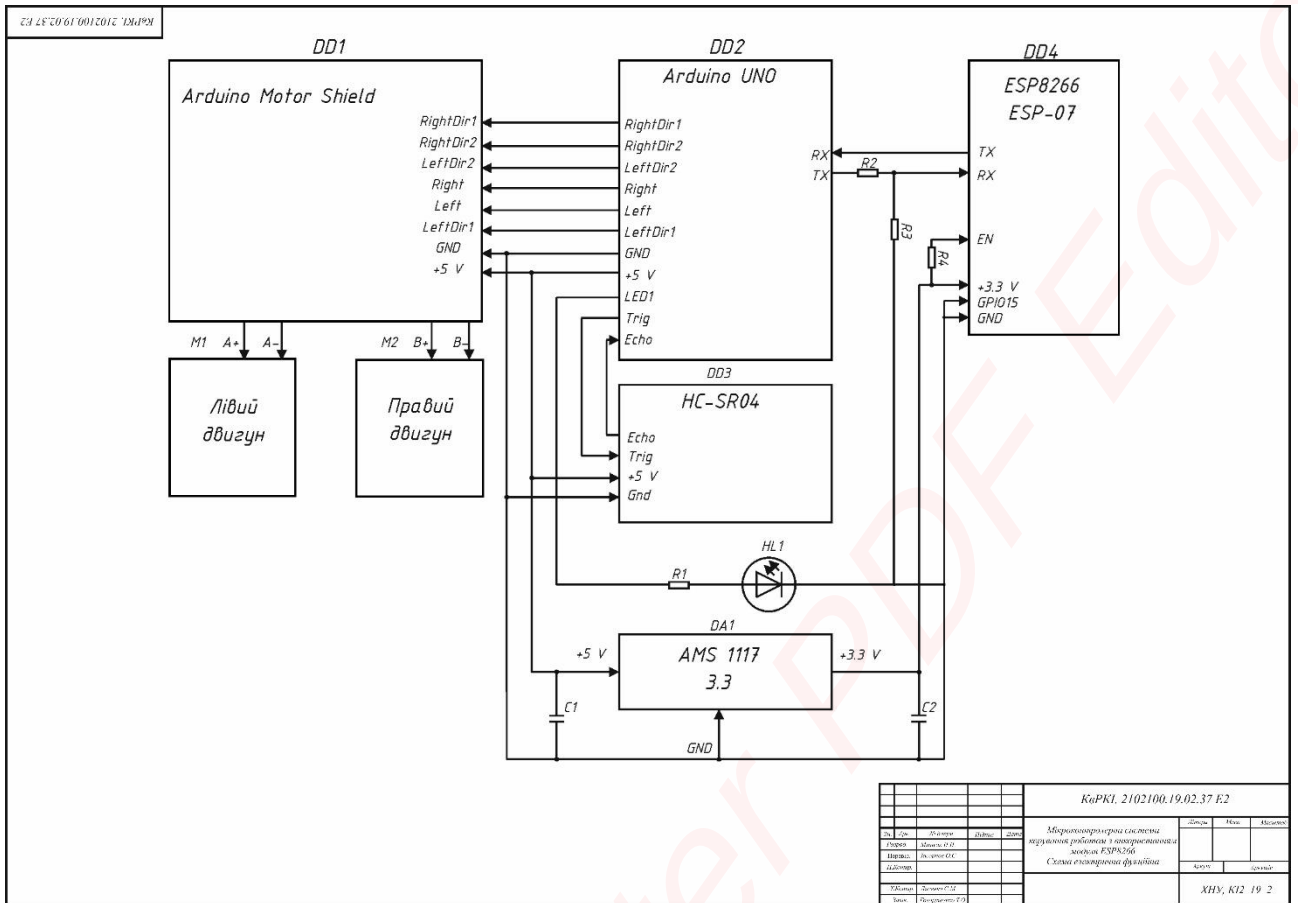
ДОДАТОК В

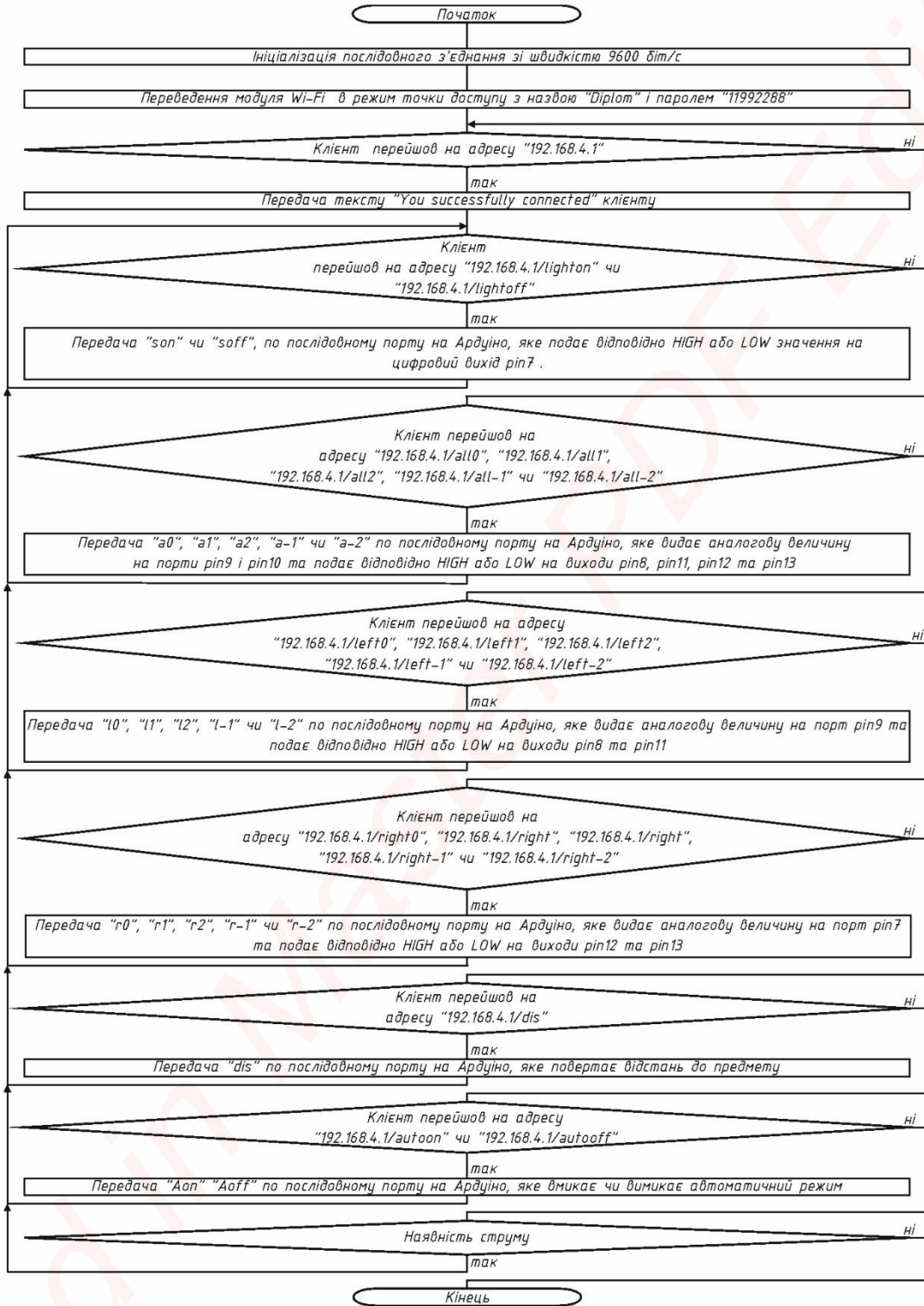
(обов'язковий)

Копія графічної частини



КодРК1. 2102100.19.02.37 E.1				
№ з/п	Видове	Об'єкт	Дата	Стан
1	Модель	ESP8266		
2	Версія	1.0		
3	Статус	Активний		
4	Рівень	Високий		
5	Місце	Україна		
6	Місто	Київ		
7	Район	Печерський		
8	Місцевість	Печерський район		
ХНУ, К12 19 2				





					КвРКІ. 2102100.19.02.37.Е8		
№	Дат.	Місце	Підпис	Дата	Ліцензія	Місце	Місце
Розроб.	Михайло П.П.						
Проєкт.	Людмила О.С.						
Інженер							
І.Викор.	Людмила С.М.						
Зам.	Григорій І.І.						
					Мікроконтролерна система керування роботом з використанням модуля ESP8266 Алгоритми роботи		
					ХНУ, КІ2-19-2		

Ім'я користувача:
Кафедра КІ

Дата перевірки:
30.05.2023 07:26:36 EEST

Дата звіту:
30.05.2023 07:27:23 EEST

ID перевірки:
1015310255

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100005591

Назва документа: Матушак_Мікроконтролерна система керування роботом з використанням модуля ESP8266

Кількість сторінок: 67 Кількість слів: 13086 Кількість символів: 96342 Розмір файлу: 2.62 MB ID файлу: 1014981425

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

16.4% Схожість

Найбільша схожість: 8.51% з Інтернет-джерелом (<http://elar.khmnmu.edu.ua/bitstream/123456789/12076/1/%D0%A7%D0%>)

15.1% Джерела з Інтернету

359

Сторінка 69

13.7% Джерела з Бібліотеки

200

Сторінка 71

0.24% Цитат

Цитати

3

Сторінка 72

Не знайдено жодних посилань

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

20

Підозріле форматування

12
сторінок

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 2.0%

Словники перевірки: en_US, ru_RU, ua_UA. Помилки в документах: 10%

ID: 114264 Назва: БКР Мікроконтролерна система керування роботом з використанням модуля ESP8266 Додано в БД: 2023-05-30 Автора: П.П. Матушак Керівники: О.С. Засорнов Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	70656	1091	4048 (6%)	63 (6%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ
освітнього ступеня «бакалавр»

Студент Матушаку Павлу Петровичу

Тема Мікроконтролерна система керування роботом з використанням модуля ESP8266

Спеціальність 123 – Комп'ютерна інженерія

Обсяг кваліфікаційної роботи освітньо-кваліфікаційного рівня «бакалавр»:

кількість листів креслень 4; кількість сторінок записки 64

1. Короткий зміст роботи та прийнятих рішень У кваліфікаційній роботі розроблено системи дистанційного управління роботом, яка працює на базі контролера Arduino та керується за допомогою Android-пристрою через бездротову лінію зв'язку

2. Висновок про відповідність кваліфікаційної роботи завданню Кваліфікаційна робота у повній мірі відповідає поставленому завданню як в теоретичній, так і в практичній частині

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі подана загальна характеристика поставленої задачі, сформульована актуальність. Визначені задачі, які необхідно вирішити для досягнення поставленої мети, практична цінність отриманих результатів, У першому розділі проведено огляд використовуваних систем керування та основні підходи до їх проектування, виконане обґрунтування актуальності теми дослідження і виконана постановка задачі. В другому розділі проведено обґрунтування обраного методу рішення та описано будову пристрою на рівні структурної схеми. В третьому розділі описано пристрій на рівні функційної схеми та розроблено алгоритми її роботи, спроектовано принципову схему та описано програмування мікроконтролера

4. Позитивні сторони роботи Кваліфікаційна робота має комплексну практичну цінність. Практична цінність результатів кваліфікаційної роботи полягає у створенні мікроконтролерного пристрою, що має широкий спектр застосувань, як у побуті та господарстві для дистанційного керування побутовими пристроями, так і в промисловості для керування виробничими процесами та електричним устаткуванням

5. Негативні сторони роботи Доцільно було б розглянути питання надійності розробленого пристрою

6. Оцінка графічного оформлення та пояснювальної записки роботи Графічне оформлення виконане відповідно до теми кваліфікаційної роботи з дотриманням стандартів. В загальному графічне оформлення виконане якісно, пояснювальна записка відповідає нормам щодо її оформлення.

7. Відгук про роботу в цілому В загальному кваліфікаційна робота заслуговує позитивної оцінки. Весь матеріал кваліфікаційної роботи структурований, чіткий та послідовний. Усі розділи роботи послідовні та логічні, що дозволяє чітко розуміти викладений матеріал в рамках тематики кваліфікаційної роботи. Графічний матеріал дозволяє наочно побачити доцільність та ефективність рішень, які були прийняті за основу для досягнення поставленої мети.

8. Інші зауваження Окремі описи в пояснювальній записці подано занадто деталізовано, що ускладнює сприйняття матеріалу фахівцями в обраній предметній галузі

9. Оцінка кваліфікаційної роботи Враховуючи всі позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що вона заслуговує оцінку «відмінно» А.

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи) _____

к.т.н., доц, ст.викладача кафедри кібербезпеки Муляр І.В. _____

« 5 » гербня 2023.

 (підпис)

Завідувачу кафедри КІС
д-р.техн.наук, проф. Говорущенко Т. О.

Матушак Павла Петровича
ПІБ здобувача вищої освіти

ФІТ, 4 курсу, групи КІ2-19-2

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

22 квітня 2023 року



РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Мікроконтролерна система керування роботом з використанням модуля ESP8266

Автор: Матушак Павло Петрович

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Засорнов Олександр Сергійович, к.т.н, доц.

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) запозичення розміщені в розділах аналізу існуючих аналогів та технологій, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 3) окремі виявлені збіги є загальноживаними шаблонами, що використовуються при оформленні текстової документації, а саме шаблони рамок
- 4) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту, використання абревіатур.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 16.4% і адресується до 359 першоджерел, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІС







О.С. Засорнов

С. М. Лисенко

Т. О. Говорущенко