

Міністерство освіти і науки України

Луцький національний технічний університет

(повне найменування закладу вищої освіти)

Факультет комп'ютерних та інформаційних технологій

(повне найменування факультету)

Кафедра комп'ютерної інженерії та безпеки

(повне найменування кафедри)

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»**

**СИСТЕМА АВТОМАТИЧНОГО ПРОПУСКУ ТРАНСПОРТНИХ
ЗАСОБІВ З ВИКОРИСТАННЯМ ARDUINO UNO**

AUTOMATIC VEHICLE ACCESS SYSTEM USING ARDUINO UNO

спеціальність 123 Комп'ютерна інженерія

(шифр і назва спеціальності)

освітня програма Комп'ютерна інженерія

(назва освітньої програми)

Виконав: здобувач вищої освіти
групи КІс-21
Дифорт Ігор Володимирович

(підпис)

Керівник:
к.т.н., доцент
Гринюк Сергій Васильович

(підпис)

Кваліфікаційну роботу
допущено до захисту
« 10 » червня 2025 р.

Гарант освітньої програми:

к.т.н., доцент
Лавренчук Світлана Василівна

(підпис)

Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій

Кафедра комп'ютерної інженерії та безпеки

Ступінь вищої освіти: бакалавр

Галузь знань: 12 Інформаційні технології

Спеціальність: 123 Комп'ютерна інженерія

Освітня програма: «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри

доц. Тарас ТЕРЛЕЦЬКИЙ

« 10 » 01 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Дифорту Ігорю Володимировичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи *Система автоматичного пропуску транспортних засобів з використанням Arduino Uno*

Керівник роботи *к.т.н., доц. Гринюк Сергій Васильович*

затвержені наказом закладу вищої освіти від «04» січня 2025 року № 11/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи *10.06.2025р.*

3. Вихідні дані до роботи *джерелом розробки є науково-технічна література та публікації в періодичних виданнях з даного питання, опубліковані зарубіжні та вітчизняні роботи в даній області та різні інтернет-ресурси технічного спрямування.*

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

Вступ

Аналіз існуючих систем автоматичного пропуску

Проектування та розробка автопропуску

Тестування власного рішення

Висновки

5. Перелік графічного (ілюстративного) матеріалу:

Існуючі рішення сигналізації з датчиками

Загальна схема

Схема підключення датчиків

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
<i>Аналіз існуючих систем автоматичного пропуску</i>	<i>Гринюк С.В., доцент</i>		
<i>Проектування системи автопропуску</i>	<i>Гринюк С.В., доцент</i>		
<i>Реалізація та тестування системи</i>	<i>Гринюк С.В., доцент</i>		
<i>Нормоконтроль</i>	<i>Багнюк Н.В., доцент</i>		
<i>Гарант ОП</i>	<i>Лавренчук С.В., доцент</i>		
<i>Показник запозичень тексту</i>		_____%	
<i>Академічна доброчесність</i>	<i>Міскевич О.І., ст. викладач</i>		

7. Дата видачі завдання 10.01.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	<i>Огляд літератури із досліджуваної проблеми, аналіз предметної області та наявних рішень</i>	до 08.02.2025 р.	Виконано
2.	<i>Аналіз існуючих систем автоматичного пропуску та розробка автопропуску</i>	до 06.03.2025 р.	Виконано
3.	<i>Реалізація та тестування системи</i>	до 02.04.2025 р.	Виконано
4.	<i>Висновки та пропозиції</i>	до 08.04.2025 р.	Виконано
5.	<i>Формування списку використаних джерел</i>	до 18.04.2025 р.	Виконано
6.	<i>Формування додатків</i>	до 03.05.2025 р.	Виконано
7.	<i>Оформлення ілюстративного матеріалу</i>	до 12.05.2025 р.	Виконано
8.	<i>Представлення остаточного варіанту кваліфікаційної роботи керівникові</i>	до 15.05.2025 р.	Виконано
9.	<i>Нормоконтроль</i>	до 30.05.2025 р.	Виконано
10	<i>Інструментальна перевірка на академічний плагіат</i>	до 03.06.2025 р.	Виконано
11.	<i>Здача кваліфікаційної роботи та всіх супровідних документів на кафедру</i>	до 10.06.2025 р.	Виконано

Здобувач вищої освіти

(підпис)

Дифорт І.В.

(прізвище, ініціали)

Керівник кваліфікаційної роботи

(підпис)

Гринюк С.В.

(прізвище, ініціали)

АНОТАЦІЯ

Дифорт І.В. Система автоматичного пропуску транспортних засобів на базі Arduino Uno. Рукопис.

Кваліфікаційна робота бакалавра ОП «Комп'ютерна інженерія» спеціальності 123 Комп'ютерна інженерія. Луцький національний технічний університет. Луцьк, 2025.

Кваліфікаційна робота складається з вступу, трьох розділів, висновків, списку використаних джерел та додатків.

Перший розділ присвячено аналізу існуючих систем автоматичного пропуску транспортних засобів. У ньому розглядаються сучасні принципи роботи систем контролю доступу, основи функціонування мікроконтролерних платформ, а також порівнюються різні технології та підходи до автоматизації паркувальних систем, зокрема використання датчиків і IoT-рішень.

Другий розділ зосереджено на розробці системи автоматичного пропуску транспортних засобів на базі мікроконтролера Arduino Uno. Виконано вибір апаратних компонентів, таких як ультразвуковий датчик HC-SR04, серводвигун SG90, RGB-світлодіод та звуковий модуль. Описано схему підключення компонентів, розроблено алгоритми роботи системи та наведено архітектуру програмного забезпечення для керування шлагбаумом.

Третій розділ присвячено реалізації та тестуванню розробленої системи. У ньому детально описано програмування мікроконтролера в середовищі Arduino IDE, включаючи написання скетчу для обробки даних з ультразвукового датчика та керування виконавчими механізмами. Проведено тестування системи, перевірено її працездатність, чутливість і стабільність у реальних умовах. Результати підтверджують ефективність системи для автоматизації доступу транспортних засобів.

Ключові слова: Arduino Uno, автоматичний пропуск, ультразвуковий датчик, серводвигун, паркувальна система, IoT, мікроконтролер.

ANNOTATION

Dyfort I. Automatic Vehicle Passing System on Arduino Uno. Manuscript.

Bachelor's qualification work for the Educational Program «Computer Engineering» specialty 123 Computer Engineering. Lutsk National Technical University. Lutsk, 2025.

The qualification work consists of an introduction, three chapters, conclusions, a list of references, and appendices.

The first chapter is dedicated to the analysis of existing automatic vehicle passing systems. It examines modern principles of access control systems, the fundamentals of microcontroller platform operations, and compares various technologies and approaches to parking system automation, including the use of sensors and IoT solutions.

The second chapter focuses on the development of an automatic vehicle passing system based on the Arduino Uno microcontroller. The selection of hardware components, such as the HC-SR04 ultrasonic sensor, SG90 servo motor, RGB LED, and sound module, is performed. The connection scheme of the components is described, algorithms for the system's operation are developed, and the software architecture for barrier control is presented.

The third chapter is devoted to the implementation and testing of the developed system. It details the programming of the microcontroller in the Arduino IDE environment, including the creation of a sketch for processing data from the ultrasonic sensor and controlling actuators. The system was tested, and its functionality, sensitivity, and stability were verified under real-world conditions. The results confirm the system's effectiveness for automating vehicle access.

Keywords: Arduino Uno, automatic passing, ultrasonic sensor, servo motor, parking system, IoT, microcontroller.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 АНАЛІЗ СИСТЕМ АВТОМАТИЧНОГО ПРОПУСКУ	9
1.1 Принципи роботи сучасних систем контролю доступу	9
1.2 Аналіз існуючих рішень на базі мікроконтролерів	15
1.3 Вибір оптимальної технології для реалізації системи	22
РОЗДІЛ 2 ПРОЄКТУВАННЯ СИСТЕМИ АВТОПРОПУСКУ	25
2.1 Загальна схема роботи системи	25
2.2 Вибір компонентів	28
2.3 Опис апаратної частини	32
2.4 Архітектура програмного забезпечення	34
2.5 Розробка алгоритмів роботи системи	36
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ	38
3.1 Програмування мікроконтролера Arduino Uno	38
3.2 Робота сервоприводу для відкриття/закриття шлагбаума	40
3.3 Відправка даних на сервер або ПК	42
3.4 Аналіз результатів тестування	45
ВИСНОВКИ	49
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	51
ДОДАТКИ	53

ВСТУП

Зростання кількості транспортних засобів, урбанізація та економічний розвиток спричинили підвищену потребу в ефективних системах контролю доступу до паркувальних зон. Людський фактор у управлінні паркуванням часто призводить до затримок, помилок і неефективного використання ресурсів, що ставить під сумнів надійність традиційних рішень. У зв'язку з цим виникає потреба в автоматизованих системах пропуску транспортних засобів, які дозволяють оптимізувати процеси, підвищити надійність та забезпечити зручне керування. Мікроконтролери, ультразвукові датчики, сервоприводи та індикаторні модулі є основними складовими таких систем, що робить їх розробку актуальною задачею.

Метою роботи є розробка системи автоматичного пропуску транспортних засобів на базі мікроконтролера Arduino Uno з можливістю автоматичного виявлення транспортних засобів, керування шлагбаумом і візуальної індикації стану системи.

Об'єкт дослідження – розробка власного рішення системи автоматичного пропуску транспортних засобів на базі Arduino Uno.

Предмет дослідження – дослідження методів розробки автоматизованих систем пропуску транспортних засобів з використанням мікроконтролерів.

Завдання, які необхідно виконати:

- проаналізувати існуючі системи автоматичного пропуску транспортних засобів, що базуються на мікроконтролерах;
- спроектувати систему автоматичного пропуску на основі мікроконтролера Arduino Uno;
- реалізувати алгоритм обробки даних з ультразвукових сенсорів для виявлення транспортних засобів;
- розробити механізм керування шлагбаумом за допомогою сервоприводу та індикації стану системи через RGB-світлодіоди й звуковий сигнал;

– створити програмне забезпечення для Arduino, яке забезпечує синхронізацію роботи датчиків, сервоприводу, світлодіодів і звукового модуля;

– виготовити та протестувати працездатну систему, здатну ефективно функціонувати як автономний пристрій контролю доступу.

РОЗДІЛ 1

АНАЛІЗ СИСТЕМ АВТОМАТИЧНОГО ПРОПУСКУ

1.1 Принципи роботи сучасних систем контролю доступу

З розвитком технологій інтелектуальних та колаборативних обчислень та відповідних рішень безпеки з'явилися та набули дедалі більшої популярності різноманітні програми для інтелектуального та колаборативного обміну інформацією та управління нею. На відміну від інших обчислювальних областей, інтелектуальні та колаборативні обчислювальні системи (СКО) сприяють складній взаємодії між користувачами/пристроями/організаціями для спільного використання ресурсів та адміністрування діяльності між різними учасниками, такими як інтелектуальні об'єкти, користувачі, хмари або периферійні комп'ютери. У таких СКО кілька учасників спільно створюють, обмінюються, керують та захищають цифровий контент та інші ресурси [1]. Ці складні взаємодіючі дії вимагають складної системи контролю доступу. Зокрема, рішення щодо доступу приймаються колективно на основі політик, якими керують кілька адміністративних органів, таких як власники/адміністратори пристроїв, користувачі, чия інформація включена до контенту, постачальники хмарних послуг, адміністратори безпеки та сторонні програми. Ця інтелектуальна та пов'язана екосистема пропонує безмежні можливості з точки зору безпеки, охорони здоров'я, доступності та економічного зростання. Вона змінює загальну якість життя завдяки потужності даних, які вона створює, та неймовірним програмам, які вона дозволяє використовувати. У поєднанні з фізичними пристроями, такими як машини, датчики, будівлі тощо, такі технології створюють інтелектуальну та автоматизовану систему, яка є більш ефективною, результативною та працює в режимі реального часу. Маючи мільярди підключених пристроїв, хмара надає базові послуги для підтримки пристроїв Інтернету речей (IoT), які зазвичай обмежені в ресурсах, а також додатків IoT для сталого успіху та широкого впровадження IoT у реальному світі [2].

Однак традиційні моделі контролю доступу не розроблені для такого типу складного та динамічного середовища SCS. Було запроваджено кілька моделей контролю доступу на основі взаємовідносин (ReBAC) [3], які використовують інформацію про взаємовідносини учасників для специфікації політик та рішень щодо контролю доступу. Хоча використання взаємовідносин користувачів (або навіть ресурсів) для контролю доступу є необхідним та природним підходом для системи онлайн-соціальних мереж (OSN), це найкраще працює для системи, в якій доступна інформація про соціальні взаємовідносини. Інші SCS, такі як системи соціальних рекомендацій та репутації, системи спільного Інтернету речей та системи підключених транспортних засобів, які не підтримують чітко визначену та постійну інформацію про взаємовідносини користувачів/ресурсів, потребують іншого рішення для контролю доступу, ніж ReBAC. Крім того, типові системи ReBAC не сприяють складному та динамічному контролю різних дій з використання та контролю, що виконуються кількома учасниками, такими як користувачі та сторонні програми, що, у свою чергу, може впливати на діяльність діючого учасника або інших учасників. Поняття контролю активності (ACON) було вперше запропоновано для охоплення контролю доступу до цих багатосторонніх дій, що зустрічаються в середовищах онлайн-соціальних мереж.

З появою нових обчислювальних систем та застосувань дослідники безпеки пропонують різні рішення для контролю доступу, щоб задовольнити вимоги безпеки цих систем. Хоча останні академічні дослідження зазвичай зосереджуються на конкретній проблемі та питанні контролю доступу, що зустрічається в певній цільовій системі, та пропонують часткове рішення з перевіркою концепції, також були деякі помітні дослідження, які визначили ключові принципи проектування високого рівня для систем безпеки та контролю доступу. Наприклад, у 1975 році Сальцер та Шредер визначили вісім принципів проектування для забезпечення та впровадження систем безпеки. Принципи RBAC96 [4] та принципи ASCAA були запропоновані для

проєктування політики та моделі контролю доступу на основі ролей. Пізніше Сміт переглянув та дослідив принципи проєктування Сальцера та Шредера для безпеки сучасних обчислювальних систем [5]. Ці принципи безпеки мають вирішальне значення для проєктування та розробки рішень контролю доступу для різних типів обчислювальних систем та застосувань. Хоча багато з цих існуючих принципів досі дійсні, їх недостатньо для адекватного охоплення основних аспектів контролю діяльності, оскільки вони представляють унікальні та складні засоби контролю, які відсутні в багатьох існуючих моделях контролю доступу.

Замість розробки ще однієї моделі контролю доступу, яка використовує один або декілька типів параметрів прийняття рішень, головною метою цієї статті є визначення сфери застосування контролю активності та розробка його базової бази знань для інтелектуальних та колаборативних систем. Зокрема, ми обговорюємо кілька важливих характеристик існуючих рішень контролю доступу та їхніх підходів, які є важливими та корисними для розуміння контролю активності та принципів його проєктування.

Ми пропонуємо вдосконалену структуру ACON та обговорюємо її сферу застосування та ключові компоненти. Потім ми розглядаємо та порівнюємо існуючі принципи проєктування систем контролю доступу та обговорюємо обмеження цих принципів. На основі цього обговорення пропонується набір нових принципів проєктування для контролю активності. Запропонована структура та принципи проєктування ACON надають орієнтир для проєктування та розробки систем контролю активності та допомагають дослідникам і практикам зрозуміти сучасні системи контролю доступу для інтелектуальних та колаборативних обчислювальних систем. Хоча запропонована структура та принципи проєктування ACON розроблені головним чином для підтримки деяких унікальних характеристик, що зустрічаються в інтелектуальних та колаборативних системах, ми вважаємо, що вони застосовні до інших обчислювальних систем, які включають кількох

учасників, які можуть бажати та потребувати контролювати доступ інших учасників (рис. 1.1).



Рисунок 1.1 – Приклад системи безпеки автопарку [6]

Контроль доступу для пов'язаної та спільної екосистеми створює унікальні проблеми. Використовуючи адекватні моделі та механізми контролю доступу, можна розробити підключену інтелектуальну систему, яка дозволяє контролювати активність учасників на різних ресурсах, одночасно підтримуючи конфіденційність або вподобання інших учасників. Наприклад, вона може дозволити контролювати, які інтелектуальні пристрої можуть створювати та обмінюватися даними з ким, та які програми можуть збирати дані з польових пристроїв. Тут контроль над тим, які користувацькі програми або пристрої можуть отримувати доступ до інших підключених пристроїв, є складним питанням. Крім того, важливим питанням, яке потребує вирішення, є захист даних у хмарі, локальному граничному шлюзі або транзиті. Ці проблеми можуть бути ще складнішими, коли дані та ресурси розподілені та розкидані між різними об'єктами, що адмініструються різними підрозділами.

Для забезпечення дотримання політик контролю доступу було використано кілька підходів, включаючи криптографічні механізми, можливості, списки контролю доступу та рішення на основі політик ABAC [7], підтримує можливості детальної авторизації для ресурсів, пропонуючи

гнучкість у розподіленому середовищі з кількома сутностями, де атрибути сутностей разом із контекстною інформацією використовуються для рішень щодо авторизації доступу та зв'язку. Було запропоновано кілька моделей контролю доступу, та механізмів, для задоволення потреб авторизації як в периферійних, так і в хмарних архітектурах Інтернету речей та кіберфізичних систем (CPS). Уадда та ін. [8] представили всебічний огляд моделей контролю доступу до Інтернету речей, тоді як дослідження на основі опитувань в галузі розумного домашнього Інтернету речей також підкреслили необхідність нової перспективи контролю доступу на основі взаємовідносин між власником пристрою та суб'єктом. У роботі Фернандеса та ін. [9] запропоновано нову модель збору та обміну даними для архітектур хмарного Інтернету речей, що забезпечує модуль плагіна для підтримки розробки додатків Інтернету речей. Конвергентна структура контролю доступу, нещодавно запропонована Бхаттом та Сандху, підкреслює необхідність синергетичної конвергенції моделей контролю доступу як на рівні політики, так і на рівні правозастосування, що може враховувати зростаючі вимоги до контролю доступу динамічних додатків для майбутніх розумних спільнот. У автори використовували як RBAC, так і ABAC у запропонованій системі електронних медичних записів (EHR) як для грубого, так і для дрібного контролю доступу.

Використання моделі контролю доступу на основі можливостей (CAC) для Інтернету речей було запропоновано, оскільки суб'єкти мають надані права, які підтримують різні рівні деталізації з можливістю делегування, в тому випадку подібна функціональність неможлива зі списками контролю доступу (ACL) [10]. Однак двома основними недоліками використання підходу на основі можливостей є поширення та скасування. Запропоновано модель автентифікації ідентифікації та контролю доступу на основі можливостей (IAC) [11], де пристрої використовують точку доступу та модель CAC для підключення один до одного. Крім того, система контролю доступу на основі можливостей (CapBAC) використовується для контролю доступу до послуг та інформації. Автори описують варіанти використання та

стверджують, що CapBAC підтримує делегування прав, принцип доступу з найменшими привілеями, більш детальний контроль доступу, менше проблем безпеки та менше проблем, пов'язаних зі складністю та динамікою ідентичностей суб'єкта, ніж ACL, RBAC та ABAC. Бхатт та ін. розробили формальну модель контролю доступу для платформи Інтернету речей Amazon Web Service (AWS) та запропонували вдосконалення ABAC для AWS IoT, реальної хмарної платформи Інтернету речей. Автори також нещодавно запропонували конвергентну структуру контролю доступу (CAC), яка може об'єднувати функції контролю доступу різних моделей контролю доступу (наприклад, RBAC, ABAC, ReBAC тощо) для забезпечення безпечних інтелектуальних спільнот у майбутньому. Крім того, для рівня сприйняття (об'єкта) Інтернету речей пропонується проста, ефективна взаємна автентифікація та безпечне встановлення ключів на основі криптографії еліптичних кривих (ECC), яка має значно менші накладні витрати на зберігання та зв'язок. Гупта та Сандху запропонували нову перспективу, впровадивши контроль доступу, орієнтований на діяльність, для інтелектуальних спільних систем, вважаючи діяльність, або завдання, основним поняттям для контролю нових дій у підключених системах CPS.

Інтеграція хмари та Інтернету речей була широко впроваджена в галузі основними постачальниками послуг хмарних обчислень – AWS³, Google Cloud IoT⁴ та Microsoft Azure⁵, щоб забезпечити роботу послуг та додатків Інтернету речей на базі інтелектуальних пристроїв. Ці постачальники хмарних послуг мають спеціалізовані платформи Інтернету речей та CPS, що обслуговують різноманітні додатки та варіанти використання, що підтримують як хмарні, так і периферійні користувацькі додатки та послуги в реальному часі. Для позначення цієї інтеграції використовується кілька термінів, таких як Інтернет речей, що підтримується хмарою, Інтернет речей з підтримкою хмари та Інтернет речей з підтримкою хмари, більш широко використовується як Інтернет речей з підтримкою хмари (CE-IoT) [12]. Такі підключені системи мають можливість підтримувати послуги в різних сферах,

включаючи сільське господарство, транспорт, охорону здоров'я, енергетику та виробництво, пропонуючи середовища, керовані даними та інтелектуальні.

1.2 Аналіз існуючих рішень на базі мікроконтролерів

Пошук підходящої парковки для автомобіля в сучасних містах вважається проблемою з огляду на втрачений час і енергоспоживання. У сучасну епоху кількість ліцензій на автомобілі щодня зростає дуже високими темпами, а це означає, що кількість транспортних засобів зростає, що призводить до серйозних транспортних заторів і важкості пошуку підходящої парковки. У сучасних містах паркувальні місця залишаються такими, як є, а кількість транспортних засобів збільшується. Тому сучасні міста страждають від нестачі паркувальних місць. За даними, у США щороку витрачається понад 70 мільярдів доларів на пошук місця для паркування, що дорівнює 3,6 мільярдам годин часу та 1,7 мільярдам галонів пального. Відповідно до, одна година заторів може збільшити викиди забруднювачів повітря та вуглекислого газу на 14,3-30,4 %.

За даними, основною причиною транспортних заторів у великих містах є проблема паркування автомобілів. У великих містах важко знайти відповідне місце для паркування. Тому уряди повинні впроваджувати закони для вирішення цієї проблеми. Розроблені системи довели актуальність використання інтелектуальних режимів паркування в сучасних містах. Незважаючи на те, що сучасні міста використовують різні інтелектуальні рішення для вирішення проблем паркування, проблема все ще існує.

Є два типи паркувальних місць: для окремих осіб і безкоштовні для всіх. Паркувальні місця, зарезервовані для конкретних осіб, можуть бути незайнятими, якщо особа перебуває у відпустці або не працює. Проблема стає очевидною, коли кількість співробітників в організації перевищує кількість паркувальних місць, виділених організації. З іншого боку, коли кількість

відвідувачів перевищує кількість доступних місць для паркування, відвідувачі можуть неправильно припаркуватися, що спричинить затори.

Щоденні проблеми з паркуванням можна підсумувати як:

- забезпечити динамічний розподіл паркувальних місць для вирішення проблеми дефіциту паркувальних місць.

- затори, спричинені відвідувачами, які шукають місця для паркування.

- неправильне паркування, коли водії паркують транспортні засоби у невідведеному для них місці. Це може статися в різних ситуаціях, наприклад, на автостоянці, де кожне місце призначено для певної особи, або на громадській вулиці, де існують спеціальні правила щодо місця та тривалості паркування.

Отже, оптимальне рішення, яке могло б вирішити вищезазначені проблеми, повинно:

- забезпечити динамічний розподіл паркувальних місць для вирішення проблеми дефіциту паркувальних місць.

- забезпечити механізм попереднього бронювання шляхом вказівки конкретного паркувального місця на певний час для вирішення проблеми неправильного паркування.

- забезпечити механізм, щоб переконатися, що автомобіль припаркований у правильному паркувальному місці; це рішення має вирішити проблему неправильного паркування.

Крім того, оптимальне рішення має вести облік схем руху на парковці. Ці кроки мають сформулювати механізм контролю та моніторингу для розумної системи паркування. Щоб забезпечити успіх цього механізму, він повинен працювати в режимі реального часу. На рисунку 1.2 показано архітектуру системи, яка складається з чотирьох основних компонентів: блоку моніторингу, блоку обробки, блоку на стороні хмари та програми користувача. Кожен компонент обговорюється далі.

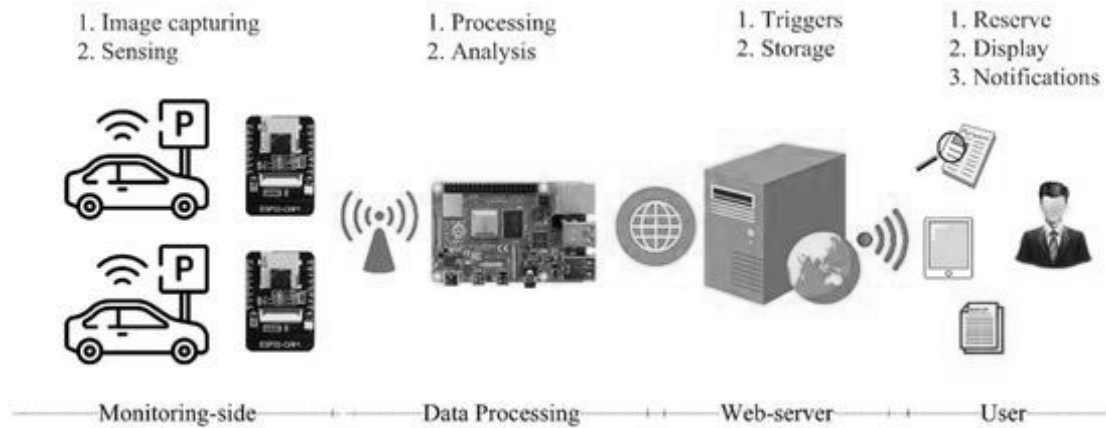


Рисунок 1.2 – Основна архітектура системи [13]

Блок моніторингу: блок моніторингу розміщено на кожному паркувальному місці та складається з розумного пристрою IoT. Цей недорогий, невеликий за розмірами та з низьким енергоспоживанням блок може виявити присутність транспортного засобу, зробити знімок і передати його на блок обробки.

Блок обробки: цей блок складається з недорогого комп'ютера (Raspberry Pi 4), який може отримувати зображення, зроблені блоками моніторингу, обробляти та отримувати необхідну інформацію із зображень за допомогою системи автоматичного розпізнавання номерних знаків (ANPR) і передавати отримані дані в блок на стороні хмари, щоб зробити їх доступними для програми користувача.

Хмарний блок цей блок збирає витягнуту інформацію із зображень, отриманих від пристрою IoT, записує остаточні деталі в базу даних і виконує функцію сповіщення користувачів. Інформація включає номерний знак транспортного засобу, час паркування та ідентифікаційний номер паркувального місця.

Програма користувача: Ця програма була розроблена, щоб дозволити користувачам виконувати різні функції, такі як реєстрація, бронювання та зміна бронювань.

На рисунку 1.3 зображено передачу даних запропонованої системи. Блок моніторингу робить фото припаркованого транспортного засобу та передає

його на блок обробки (Raspberry Pi 4). Блок обробки ідентифікує номерний знак автомобіля та розпізнає символи за допомогою оптичного розпізнавання символів (OCR) від PyTorch. Після ідентифікації персонажів у хмарній базі даних Firestore створюється новий запис, а користувачеві надсилається сповіщення за допомогою Firebase Cloud Messaging (FCM).

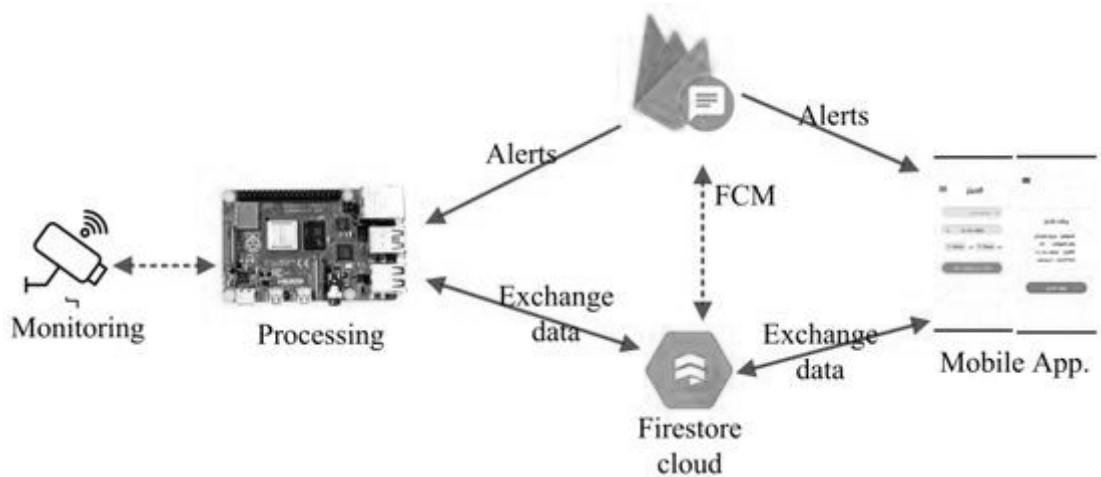


Рисунок 1.3 – Обмін даними між компонентами запропонованої системи [13]

Розроблена розумна система контролю та моніторингу паркування в режимі реального часу працює таким чином: коли автомобіль паркується на паркувальному місці, блок моніторингу виявляє його присутність за допомогою датчика далекоміра або датчика руху. Потім блок моніторингу фіксує зображення припаркованого транспортного засобу та передає його безпосередньо в блок обробки. Блок обробки перевіряє якість зображення і вимагає заміни, якщо отримане зображення має низьку якість. Після цього блок обробки обробляє та аналізує вміст зображення, щоб розпізнати номерний знак.

Після розпізнавання номерного знака деталі автомобіля, такі як номерний знак, номер паркувального місця, час і дата, передаються на хмарний сервер (у нашому випадку Firebase), щоб зберігати облік припаркованого автомобіля. Потім хмарний сервер Firebase запускає

сповіщення, щоб перевірити запис водія, щоб переконатися, що він припаркувався в правильному місці. Якщо водій припаркувався не в те місце, сповіщення буде надіслано на обліковий запис мобільного додатку водія та адміністратора. З іншого боку, коли відвідувач запитує місце для паркування, мобільний додаток надсилає запит на хмарний сервер Firebase, який схвалює запит.

Розроблена система паркування передбачає надання двох різних паркувальних пристроїв, які були розроблені з використанням одного мікроконтролера. Проте, була зроблена невелика різниця у використаній технології визначення присутності автомобіля. Іншими словами, запропонована система доступна у двох різних модулях, спрямованих на забезпечення гнучкості витрат.

Перший модуль – використання датчика руху. Архітектура першого модуля представлена на рисунку 1.4, який використовує датчик руху та плату ESP32-CAM-M5-Stack. Датчик руху ефективний з точки зору виявлення присутності транспортного засобу, однак датчик руху не може визначити особу об'єкта та не може визначити відстань до номерного знака автомобіля. Однак цей модуль забезпечує меншу вартість у порівнянні з другим модулем.

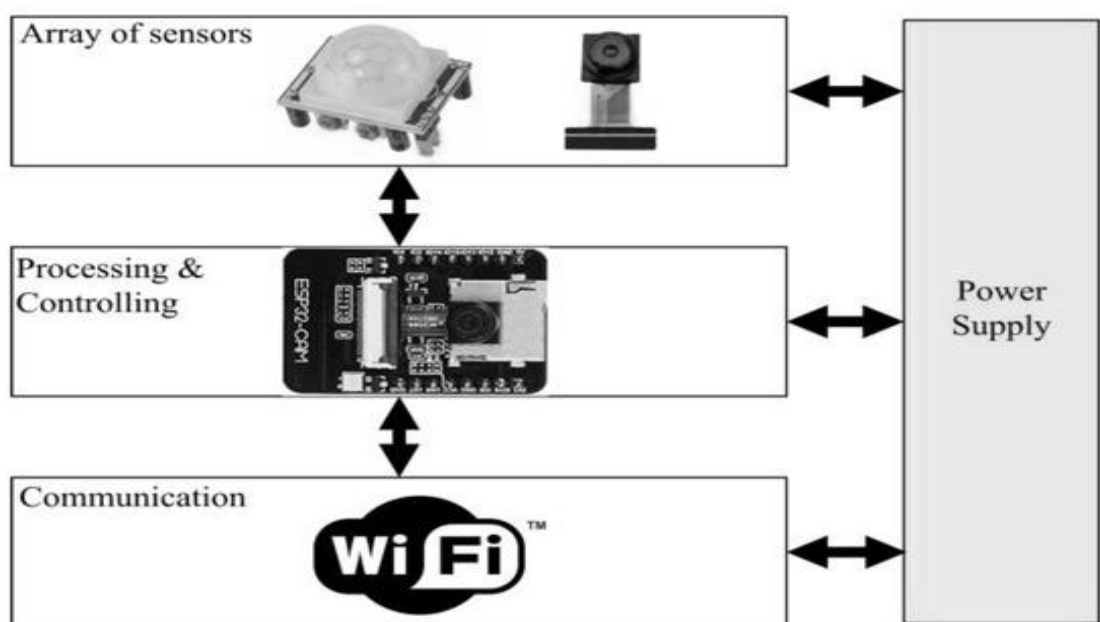


Рисунок 1.4 – Архітектура першого модуля з датчиком руху [13]

Другий модуль: використання датчика далекоміра. Архітектура другого модуля представлена на рисунку 1.5, який складається з плати ESP32-CAM-M5-Stack і датчика далекоміра. Датчик далекоміру визначає наявність автомобіля на паркувальному місці, а модуль ESP32-CAM сфотографує номерний знак автомобіля.

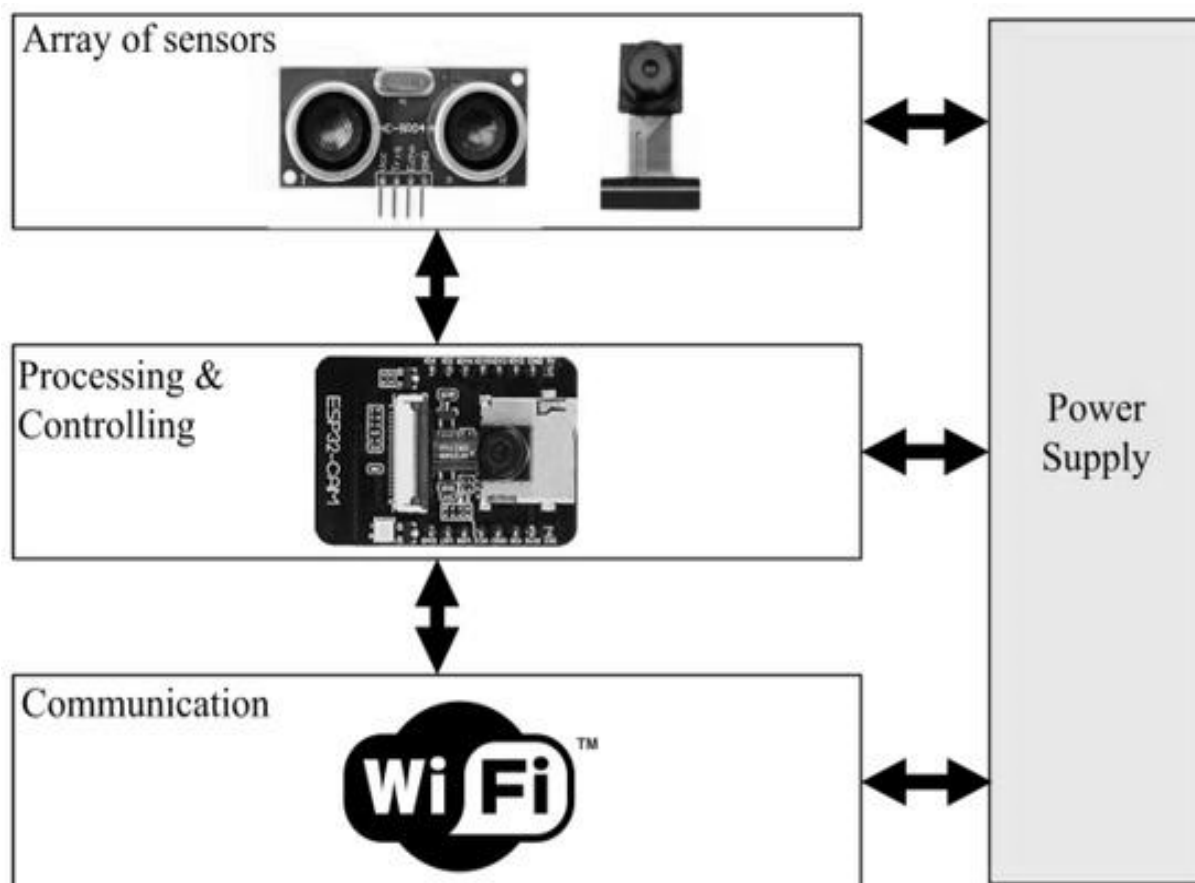


Рисунок 1.5 – Архітектура другого модуля з датчиком далекоміра [13]

Основна відмінність між цими двома пристроями полягає в їхніх сенсорних можливостях, зокрема методі, який використовується для виявлення присутності транспортних засобів. Розроблений моніторинг у першому модулі зображено на рисунку 1.6, тоді як на рисунку 1.7 демонструє розроблений моніторинг у другому модулі.



Рисунок 1.6 – Моніторинг за допомогою датчика руху [13]



Рисунок 1.7 – Моніторинг за допомогою датчика далекоміра [13]

Код IoT, який працює на модулі ESP32-CAM, включає команди для збору сигналів від бортових датчиків (далекомір або рух), на додаток до коду, який робить зображення з певною роздільною здатністю (1024×768) пікселів, і фрагмент коду, який передає зроблене зображення на сервер Firebase.

1.3 Вибір оптимальної технології для реалізації системи

Дизайн і функціональність автостоянки суттєво впливають на вибір технології контролю доступу: зчитувачів карток, брелоків або передових систем розпізнавання транспортних засобів. Великі завантажені об'єкти часто потребують налаштування, яке збалансувало б безпеку та плавний рух транспорту, особливо в години пік. Інвестиції в інноваційну систему контролю доступу підвищують безпеку, підвищують ефективність роботи та навіть забезпечують стійкість, пропонуючи спрощений досвід для користувачів. У сучасному міському ландшафті ефективний контроль доступу – це більше, ніж просто варіант – це вкрай важливо, щоб простори були безпечними, ефективними та готовими до майбутнього.

Ручний контроль доступу, як правило, керований персоналом служби безпеки, є найпростішою формою контролю доступу до паркування та значною мірою покладається на вхід людини. Незважаючи на те, що він ефективний для невеликих установок, він трудомісткий, схильний до людських помилок і не забезпечує бездоганного досвіду, очікуваного в сучасних приміщеннях.

Автоматизований контроль доступу використовує технологію, щоб забезпечити більш плавну та безпечну роботу. Ось кілька відомих типів:

Системи ANPR, які є лідерами в ефективності та результативності, використовують камери, оснащені вдосконаленим програмним забезпеченням для миттєвого зчитування та перевірки номерних знаків, що дозволяє автомобілям безперешкодно проїжджати без зупинок. Це мінімізує затори та час очікування, що особливо важливо у великих місцях з інтенсивним трафіком, таких як аеропорти, корпоративні кампуси та закриті селища.

Нещодавнє прикладне дослідження показує вражаюче 85 % скорочення часу очікування, скорочення доступу лише до 2 секунд після впровадження автоматизованої системи контролю доступу в галасливому торговому центрі.

ANPR також надає цінні дані, дозволяючи операторам відстежувати рух транспорту, оптимізувати використання простору та підвищувати безпеку за допомогою безперервного журналу руху транспортних засобів. Завдяки плавній інтеграції в системи «розумного міста» ANPR не тільки оптимізує транспортний потік, але й відіграє ключову роль у скороченні часу простою та зниженні викидів, допомагаючи містам досягати своїх цілей сталого розвитку.

RFID використовує для входу мітки транспортних засобів, що може бути зручно для об'єктів із зареєстрованими користувачами. Однак RFID-контроль доступу до автомобіля вимагає постійного обслуговування міток і може бути дорогим, якщо мітки потребують частої заміни або перепрограмування. На відміну від ANPR, RFID за своєю суттю не відстежує незареєстровані транспортні засоби, що обмежує його корисність у середовищах змішаного використання.

Ці системи пропонують звичний, низькотехнологічний варіант, коли користувачі проводять пальцем або торкаються картки, щоб отримати доступ. Незважаючи на те, що системи з картками-ключами прості та економічно ефективні, вони можуть спричинити затримки в місцях з інтенсивним рухом людей, оскільки вони потребують ручного введення.

Мобільний доступ дозволяє користувачам розблокувати точки входу за допомогою своїх смартфонів, додаючи зручності, але потенційно створюючи проблеми, якщо у користувачів виникають проблеми з підключенням. Це крок до безконтактного входу, але йому все ще не вистачає швидкості та ефективності гучного зв'язку ANPR.

Біометричні системи, включаючи відбитки пальців і розпізнавання обличчя, забезпечують надійний захист і запобігають несанкціонованому доступу. Однак їх впровадження часто повільніше та дорожче, що робить їх менш практичними для точок доступу великого обсягу транспортних засобів.

Підсумовуючи, ANPR виділяється як оптимальний вибір для жвавого сучасного паркувального середовища. Для великих паркувальних зон із декількома точками в'їзду та виїзду важливо впровадити систему контролю

доступу комерційних будівель або систему контролю доступу житлових воріт, яка збалансує безпеку з необхідністю плавного руху транспорту, особливо в години пік.

Інвестиції в вдосконалену систему контролю доступу на паркування покращують безпеку, підвищують ефективність роботи та забезпечують стійкість, забезпечуючи безперебійну роботу користувача. У сучасному світі це вже не просто розкіш – це життєво важлива частина сучасної інфраструктури.

РОЗДІЛ 2

ПРОЄКТУВАННЯ СИСТЕМИ АВТОПРОПУСКУ

2.1 Загальна схема роботи системи

Існуючі системи ручного керування паркуванням мають низку проблем та недоліків, які потребують нагального вирішення. Традиційний підхід до управління паркуванням значною мірою покладається на охоронців та ручні операції, що призводить до численних недоліків. Для подолання цих проблем і модернізації було впроваджено різноманітні розумні програми та технології для паркування. Ці програми мають на меті революціонізувати спосіб управління паркуванням та покращити загальний досвід користувачів.

Одним з помітних досягнень в управлінні паркуванням є впровадження автоматичних систем паркування, які часто підтримуються надійним програмним забезпеченням для управління паркуванням. Ці системи виявилися значно ефективнішими за традиційні методи. Інтеграція автоматизації в системи паркування стала необхідною, оскільки традиційні підходи демонструють значні обмеження у вирішенні сучасних викликів, пов'язаних зі зростаючою кількістю транспортних засобів.

Arduino використовуються для різних інтелектуальних систем для контролю транспортних засобів та управління дорожнім рухом [14]. Розумне та інтелектуальне програмне забезпечення для управління паркуванням забезпечує та вирішує кілька поширених проблем, пов'язаних з ручними системами паркування, таких як, ручні системи паркування вимагають, щоб охоронці керували шлагбаумами, керували в'їздом і виїздом, а також виконували різні адміністративні завдання. Така надмірна залежність від людського втручання може призвести до неузгодженостей та операційних проблем.

Керівники парковок часто покладаються на ручні перевірки, які включають фізичну перевірку дозволів, підрахунок транспортних засобів і

видачу талонів, написаних від руки. Ці ручні процеси можуть призвести до значних втрат доходу, оскільки ймовірність помилок становить 50 %.

Ручне введення даних і повторювані завдання забирають багато часу та зусиль. Трудомісткі операції не тільки підвищують операційні витрати, але й роблять систему схильною до людських помилок. Витрати на робочу силу, пов'язані з обслуговуванням ручних систем паркування, вищі через необхідність ручного втручання [15].

Клієнти, які користуються ручними системами паркування, часто стикаються з тривалим часом очікування при в'їзді та виїзді з паркінгу. Застарілі або ручні операції можуть призвести до прикрих затримок, марнування часу клієнтів і потенційної шкоди репутації бізнесу або об'єкту.

Ручні системи можуть не забезпечувати належного управління відвідувачами, оскільки охоронці можуть дозволити вхід без належної перевірки, створюючи ризики для безпеки. Такий підхід може призвести до несанкціонованого доступу та потенційних загроз безпеці.

Для вирішення цих проблем необхідний перехід на автоматизовані системи паркування з повністю інтегрованим програмним забезпеченням для управління паркуванням. Автоматизовані системи пропонують такі переваги, як ефективність, економічність та екологічність. Перехід до повної автоматизації загальних завдань, включаючи паркування, обіцяє покращити користувацький досвід і задовольнити потреби сучасного суспільства, що постійно змінюються. Поглиблення цієї трансформації та впровадження інноваційних рішень, таких як Arduino для паркування, може ще більше оптимізувати управління паркуванням [16].

Автоматизовані системи паркування забезпечують безперебійний і зручний досвід для користувачів, оскільки вони можуть швидко знаходити вільні місця для паркування без зайвих затримок і людських помилок. Цей покращений досвід заохочує лояльність клієнтів та позитивні відгуки.

Зменшуючи потребу в ручному продажу квитків і неефективній циркуляції всередині парковок, автоматизовані системи сприяють

збереженню навколишнього середовища. Вони зменшують викиди вуглекислого газу, пов'язані з простоем транспортних засобів та паперовими відходами.

Автоматизація дозволяє точніше відстежувати плату за паркування та зменшує ризик витоку доходів через помилки ручної роботи або несанкціоновані записи. Це може призвести до збільшення прибутковості для операторів парковок [17].

Система автоматичного пропуску транспортних засобів, побудована на базі плати Arduino UNO від Keystudio, призначена для контролю доступу автомобілів через шлагбаум за допомогою сенсорів і виконавчих механізмів. Основним елементом виявлення транспортного засобу є ультразвуковий датчик HC-SR04, який вимірює відстань до об'єкта, надсилаючи звуковий імпульс і фіксуючи час його повернення. Коли датчик фіксує автомобіль на заданій відстані, мікроконтролер обробляє цей сигнал і активує інші модулі системи (рис. 2.1).

Серводвигун SG90 керує підняттям і опусканням шлагбаума, отримуючи сигнали від Arduino. Він повертається на заданий кут (наприклад, 90°) для відкриття проїзду, а потім повертається у вихідне положення. Система зібрана на макетній платі MB-102, що спрощує з'єднання та тестування без пайки.

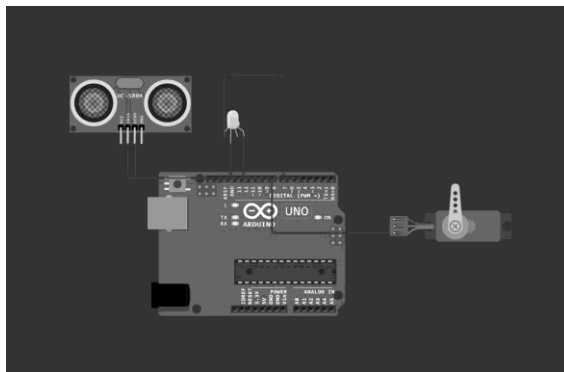


Рисунок 2.1 – Схема розроблюваного пристрою

Для візуальної індикації стану системи використовується RGB світлодіодний модуль KY-016. Залежно від ситуації, Arduino змінює його колір: червоний – шлагбаум закрито, зелений – дозвіл на проїзд, синій – режим очікування. Крім того, для звукової індикації застосовується активний модуль з динаміком KY-012, який подає короткий звуковий сигнал при виявленні автомобіля або при зміні стану шлагбаума. Усі сигнали обробляються Arduino, яка керує логікою роботи системи, приймаючи рішення на основі даних з сенсора та подаючи відповідні команди на серводвигун, світлодіод і динамік.

Таким чином, система забезпечує повністю автоматизоване керування доступом транспортних засобів, є зручною у використанні, легко модифікується, а також демонструє ефективне застосування мікроконтролерної техніки у реальних умовах.

2.2 Вибір компонентів

Arduino Uno – це відома і зручна для користувача мікроконтролерна плата з відкритим вихідним кодом, відома своєю універсальністю і простотою використання. Деякі розширені відомості про її ключові особливості наведено на (рис. 2.2).

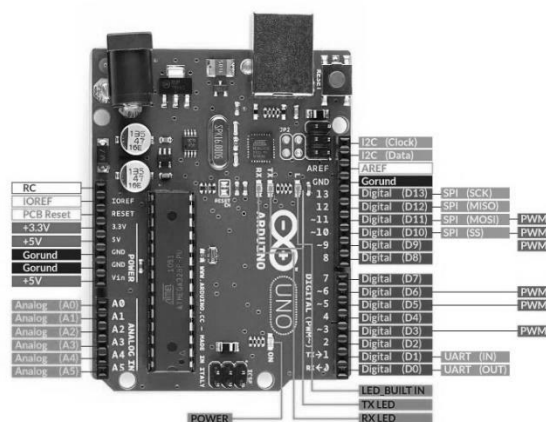


Рисунок 2.2 – Arduino UNO [18]

Arduino Uno сумісна з широким спектром електронних компонентів і плат, що робить її придатною для широкого спектра проєктів. Його мікроконтролер Microchip ATmega328P є серцем плати і рушійною силою її можливостей.

Плата складається з 14 цифрових виводів вводу/виводу (I/O), шість з яких підтримують вихід з широтно-імпульсною модуляцією (ШІМ). Ці виводи дозволяють підключати різні електронні пристрої та датчики, що дає змогу контролювати та збирати дані з різноманітних компонентів. Крім того, Uno має шість аналогових вхідних контактів для зчитування даних з аналогових датчиків.

Arduino IDE [19] пропонує зручну платформу для програмування Uno. Вона спрощує процес написання, завантаження та налагодження коду, роблячи його доступним як для початківців, так і для досвідчених розробників. IDE працює на вашому комп'ютері і дозволяє завантажувати код на плату за допомогою стандартного USB-кабелю.

Arduino Uno можна живити [20] кількома способами. Ви можете підключити її до комп'ютера за допомогою USB-кабелю, живити від стандартної 9-вольтової батареї або використовувати зовнішнє джерело живлення з напругою від 7 до 20 вольт. Така адаптивність гарантує, що дошку можна використовувати в різних сценаріях і умовах живлення.

Назва «Uno», що в перекладі з італійської означає «один», була навмисно обрана, щоб підкреслити простоту і доступність дошки. Вона підходить для широкого кола користувачів, незалежно від рівня їхнього досвіду, і заохочує почуття єдності та спільноти серед ентузіастів Arduino.

Це тип датчика, який призначений для визначення присутності, відсутності або відстані до об'єкта без фізичного контакту з ним. Він працює на основі різних технологій, зокрема

інфрачервоні, ультразвукові, ємнісні та індуктивні методи. Ось коротке пояснення того, як працює датчик наближення:

Цей датчик – це тип датчика, який призначений для виявлення наявності, відсутності або відстані до об'єкта, що знаходиться поблизу, без фізичного контакту. Він працює на основі різних технологій, включаючи інфрачервоні, ультразвукові, ємнісні та індуктивні методи. Ось коротке пояснення того, як працює датчик наближення: андартного USB-кабелю типу B.

Інфрачервоні датчики наближення випромінюють інфрачервоне світло (невидиме для людського ока), а потім реєструють відбиття цього світла. Коли об'єкт потрапляє в поле датчика, він відбиває випромінюване світло назад до датчика. Потім датчик вимірює час, необхідний для повернення світла, що дозволяє йому розрахувати відстань до об'єкта (рис. 2.3).

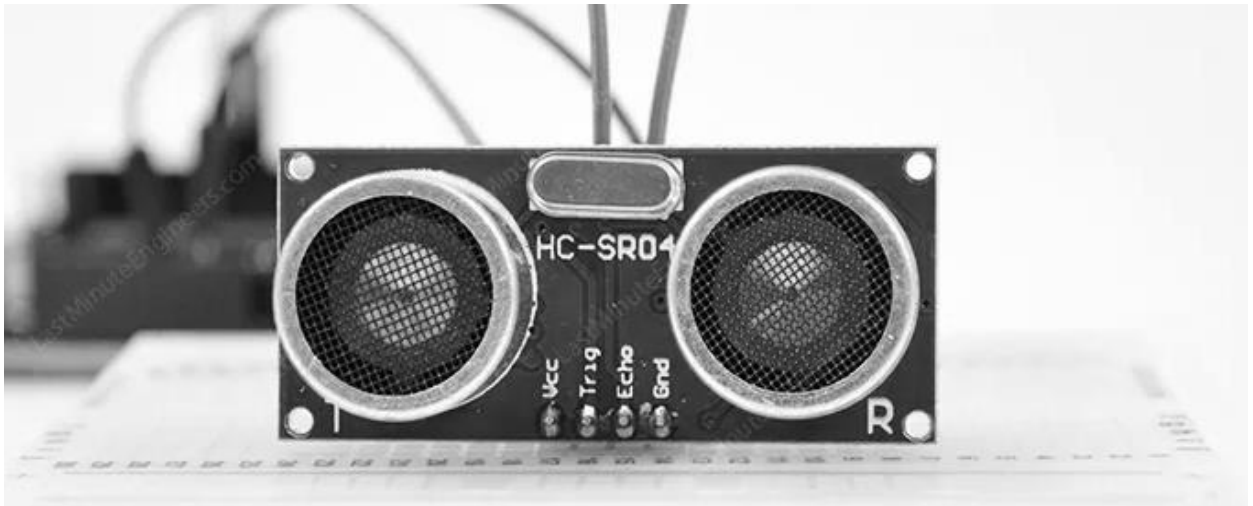


Рисунок 2.3 – Ультразвуковий датчик відстані HC-SR04 [21]

Ультразвуковий датчик відстані HC-SR04 складається з двох ультразвукових перетворювачів, які працюють разом. Один діє як передавач, перетворюючи електричні сигнали на ультразвукові імпульси 40 кГц. Інший працює як приймач, слухаючи ці імпульси після того, як вони відскакують від об'єкта.

Коли приймач виявляє ці зворотні звукові хвилі, він створює вихідний сигнал. Довжина цього сигналу безпосередньо залежить від того, наскільки далеко знаходиться об'єкт. Вимірюючи довжину сигналу, ваш Arduino може обчислити точну відстань до об'єкта.

Він може виявляти об'єкти на відстані від 2 до 400 см (приблизно 13 футів) із вражаючою точністю близько 3 міліметрів.

Оскільки датчик працює від 5 вольт, ви можете легко підключити його безпосередньо до Arduino або будь-якого іншого мікроконтролера, який використовує логіку 5 В. Це робить його простим і універсальним компонентом, який можна додати до будь-яких проєктів.

Серводвигун SG90 – компактний і легкий мікросерводвигун, популярний у робототехніці, радіокерованих моделях і DIY-проєктах завдяки доступній ціні та надійності. Має вагу близько 9 г і розміри 22,2 x 11,8 x 31 мм. Забезпечує кут повороту 180° (90° у кожному напрямку від центру). Працює при напрузі 3,0-7,2 В, оптимально 4,8-6,0 В, забезпечуючи крутний момент 1,2 кг·см при 4,8 В і до 1,8 кг·см при 6,0 В. Швидкість обертання становить 0,12 с/60° при 4,8 В і 0,10 с/60° при 6,0 В. Використовує пластикові (нейлонові) шестерні, що балансують міцність і економічність. Керується аналоговим ШІМ-сигналом із періодом ~20 мс (50 Гц) і шириною імпульсу 500–2500 мкс для позиціонування від 0° до 180°. Оснащений кабелем довжиною ~25 см із трьома проводами: червоним (VCC), коричневим (GND) і помаранчевим (сигнал). Сумісний із JR-роз'ємами та платформами, такими як Arduino чи Raspberry Pi, завдяки чому легко інтегрується з бібліотеками, наприклад, Servo.h. Застосовується в RC-моделях, роботах, гімбалах для камер і освітніх проєктах. Споживає ~10 мА в спокої, 100-250 мА під час руху і до 360 мА при блокуванні, тому для кількох двигунів рекомендується зовнішнє живлення. Працює при температурах від -30° С до +60° С. SG90 постачається з пластиковими важелями та кріпленнями. Якість залежить від виробника: оригінали від TowerPro надійніші, ніж копії. Може бути модифікований для безперервного обертання (360°), але втрачає точне позиціонування. Не рекомендується для застосувань із високим навантаженням через пластикові шестерні (рис. 2.4).



Рисунок 2.4 – Серводвигун SG90 2кг 180 [22]

Безпаяні макетні плати для тестування схем дозволяють легко збирати та тестувати схеми без використання припою, забезпечуючи швидкий і надійний спосіб створення прототипів.

Електричне з'єднання виконується шляхом вставлення проводу компонента через отвори на відстані 0,1 дюйма до пружинних затискачів, встановлених на металевих смугах у пластиковій платі з АБС. Ці смуги забезпечують спільне електричне з'єднання, щоб можна було легко додавати додаткові компоненти.

Рядки та стовпці макетної плати пронумеровані та позначені літерами, щоб допомогти у розміщенні компонентів і полярності, позначені на кожній рейці живлення.

2.3 Опис апаратної частини

Arduino – це електронна платформа з відкритим кодом, яка базується на гнучкому, простому у використанні апаратному та програмному забезпеченні для розробки інтерактивних пристроїв. Ядром Arduino є мікроконтролер AVR. Перша така платформа була запущена в 2003 році, і її зростання було вражаючим. Як результат, наразі у нього є кілька моделей, доступних для

придбання, знаючи, що всі плати мають характеристики, які дозволяють доступні та гнучкі системи розробки.

Цікавим аспектом цієї платформи є те, що плати можна створити вручну або придбати, а систему можна розширити за допомогою плат для прототипування та стандартного екрану перетворювача. Загалом, можна виконати будь-яке завдання і встановити зв'язки. Загальні характеристики плат Arduino: цифрові входи/виходи (14 контактів), аналогові входи (6 контактів), тактова частота мікроконтролера (наразі 16 МГц). Однак можна знайти відмінні від інших моделей швидкості.

Нижче наведено короткий опис основних елементів стандартної плати Arduino: Гніздо живлення: це гніздо живлення підключає Arduino безпосередньо до штепсельної вилки, напруга якої становить від 7 до 12 вольт. Коли пристрій живиться через це гніздо, регулятор напруги, підключений до вхідного діода, забезпечує регульоване живлення мікроконтролера та інших компонентів, підключених до шини 3,3 В. Ми також можемо жити пристрої через контакт 5 в або 3,3 В, але в цьому випадку прикладена напруга не регулюватиметься. Максимальний струм для цієї плати становить 50 мА. Крім того, контакти GND повинні бути підключені до землі. Джерело живлення гнізд (VIN і GND) є входом регулятора напруги, що дозволяє регульоване джерело живлення з максимальними обмеженнями від 7 в до 12 В. Однак рекомендується використовувати щонайменше 9 В, щоб зменшити вплив тепла регулятора. Нарешті, він доступний на виході МІМО через діод захисту від зворотної полярності, що дозволяє, таким чином, напрузі, вільній від зворотної полярності, надходити на вхід регулятора.

Світлодіод (LED) часто використовується як джерело освітлення в автоматизації, і він недорогий. Його енергоспоживання також відносно низьке. Отже, це робить світлодіод природним вибором джерела освітлення. Поширеною проблемою використання світлодіодів є те, що вони часто можуть спричинити легке мерехтіння через навколишнє магнітне випромінювання пристроїв. Іноді це також може змусити людей почуватися трохи незручно. Це

можна вирішити, забезпечивши світлодіод стабільним джерелом живлення. Застосування високочастотного імпульсного джерела живлення може вирішити цю проблему, забезпечивши світлодіод стабільним освітленням. Як правило, LDR і LED використовуються як світлові ворота. Досконаліші ворота можуть також включати зчитування глибини різкості як показник для визначення присутності людей. Цього можна досягти за допомогою LDR, магніту та магнітного датчика в кронштейні воріт.

Коли людина потрапляє в магнітне поле магніту, це спричиняє провал у показаннях магнітного поля, що, коли його виявляє датчик, може спричинити відкриття воріт. Датчик кольору RGB також можна використовувати для вимірювання інтенсивності світла. Потім дані про світлову енергію перетворюються в цифрове значення. Базуючись на моделі світлової енергії, дослідники використовують процедуру керування, щоб змусити датчик RGB підтримувати яскравість на бажаному рівні.

До розеток будинку можна підключати різні типи навантажень залежно від необхідної їм потужності. Перемикачі використовуються для управління ON/OFF, але вони також можуть надсилати більш цікаві команди керування. Їх можна, наприклад, легко використовувати як лампочки для пробудження, повільно вмикаючи світло в певній заздалегідь визначеній послідовності. Як правило, для такої різноманітності комунікацій потрібні датчики та виконавчі механізми, розподілені по всьому будинку.

Ця система використовується для моніторингу та керування предметами автопаркувальної системи за допомогою мікроконтролера, щоб віддавати порядок виконання цих завдань внутрішньо чи зовнішньо через комп'ютер.

2.4 Архітектура програмного забезпечення

Розробка програмного забезпечення передбачає програмування мікроконтролера, який керує екранами. Перший крок полягає в ідентифікації всіх домашніх пристроїв і електроприладів, щоб знайти відповідний вихідний

сигнал мікроконтролера. Потім розробникам потрібно визначити відповідний вихідний контакт транзисторів екрану реле.

Після цього скетч програмується, щоб викласти послідовність логічних рівнів, які надсилаються як вхідні дані для кожного реле. Таким чином було розроблено програмне забезпечення, яке містить перелік пристроїв керування та кнопок. Враховуючи актуальність графічного інтерфейсу для домашньої автоматизації, програмне забезпечення було віддано перевагу, щоб краще задовольняти потреби користувачів, полегшуючи використання пристроїв керування. Діяльність долає аспект інтерфейсу, оскільки всі дошки розпізнаються автоматичною системою.

Після вибору та налаштування комунікаційної плати всі екрани мають однаковий налаштований контакт, який можна відстежити в ескізах, використовуючи файли цих плат, які додаються до оновленого ескізу та коду протоколу. Незважаючи на всі ці проблеми, користувач не має втрат щодо того, що було змінено у його вихідному коді, оскільки система була розроблена звичайним способом, наприклад, бібліотеками, які містять усі функції, необхідні для функціонування плат.

Після кодування апаратного забезпечення необхідно згрупувати Uno та Ethernet, виконавши ще одне налаштування контактів, щоб встановити канали передачі даних. Для нашої реалізації кожен мікроконтролер зв'язку відповідає за до восьми відділів, обговорюючи та ділячи будинок з іншими зв'язками, таким чином збільшуючи здатність контролювати пару реле.

Середовище розробки Arduino дозволяє писати програми та спілкуватися з мікроконтролером на платі Arduino. Програма, яку буде написано, зберігається з розширенням .ino. Після збереження цієї програми Arduino IDE компілює її, перевіряє, чи все в порядку з програмою, а потім завантажує її на плату Arduino. Монітор, вбудований у послідовний порт, дозволяє вам взаємодіяти з цією платою. Arduino IDE по суті є компілятором C/C++ для сімейства мікроконтролерів Atmel AVR.

Вхід/вихід загального призначення діє як цифровий контакт, контрольний контакт під номером 2. Коли кнопку натиснуто, тяговий резистор встановлюється на високий рівень, а відповідний висновок знаходиться на 0 для індикації. Вивід 12 підключається до модуля реле, щоб модуль мав контактувати лише в разі зміни логічного рівня виводу, а вивід № 13 перевіряє тяговий резистор для стандартної перевірки світлозалежного резистора, щоб запобігти нагріванню протягом сонячного дня. Сигнал зміни описує спосіб вмикання та вимикання опалення. Крім того, порівняно деякі методи ввімкнення/вимкнення RA та ПД-регулятора та встановлено, що найкращим методом є ПД-регулятор, який забезпечує зазори між 0,2 та 0,5 градуса та вдвічі менший час роботи порівняно з методом вимкнення/увімкнення. Однак метод вимкнення/увімкнення вимагає менше логічного читання та має інші додаткові недоліки.

2.5 Розробка алгоритмів роботи системи

Автоматизована система паркування може виконувати операції з паркування швидко та безпечно без втрат часу та енергії. Це дослідження описує масштабну модель автоматизованої системи паркування з, яка може регулювати та керувати кількістю транспортних засобів, які можна припаркувати в певній зоні будь-який момент часу залежно від наявності місць для паркування. Використання датчиків під час входу та виходу з автомобіля називається автоматичним паркуванням. Згідно з нашим аналізом кількох існуючих систем, більшість поточних систем не є повністю автоматизованими й потребують певного втручання людини або взаємодії з нею або з нею. Ми перевірили деякі з існуючих систем, і результати вказують на те, що більшість із них не повністю автоматизовані та потребують певної участі людини.

У нашій пропозиції модель автоматизованої системи паркування, яка обмежує кількість транспортних засобів, які можуть бути припарковані в

певному місці будь-який момент залежно від наявності місця для паркування. Цей проєкт спрямований на автоматизацію паркінгу, щоб автомобілі могли заїжджати та паркуватися.

Дійшовши до в'їзду, автомобіль зупиниться над доріжкою, що веде до стоянки. Автомобіль буде чекати на білій смузі за межами паркувального майданчика, коли він наближається до воріт і шукає вільне місце. Автомобіль продовжить свій шлях до вільного місця для паркування після того, як йому буде призначено вільне місце. Дані поступають на світлодіоди оновляться автоматично після успішної спроби паркування. Інтерфейс мікроконтролера з звуковим сенсором та сервомотором використовувався як фундаментальний модуль у реалізації цієї моделі. Щоб дістатися до воріт паркування, автомобіль їде по стежці. Блок паркування та мікроконтролери автомобіля з'єднуються біля воріт, коли перевіряється наявність вільного паркувального місця. Якщо виявляється вільне місце, воно призначається, і автомобіль слідує вказаним шляхом, щоб припаркуватися там.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ

3.1 Програмування мікроконтролера Arduino Uno

Автоматична паркувальна бар'єрна система з Arduino – це система, яка використовує мікроконтролер Arduino для керування відкриттям і закриттям шлагбауму, щоб обмежити або дозволити транспортним засобам в'їзд або виїзд з стоянки. Arduino – це недорогий мікроконтролер із відкритим вихідним кодом, який можна запрограмувати на взаємодію з різними датчиками та приводами для керування шлагбаумом.

Щоб створити автоматичну шлагбаумну систему за допомогою Arduino, вам спочатку потрібно буде підключити шлагбаум до Arduino за допомогою серводвигуна. Сервопривід двигуна контролює потужність, що подається на двигун шлагбаума, і дозволяє Arduino контролювати рух шлагбаума.

Далі потрібно підключити датчики HC-SR04 до Arduino. Ці датчики можуть виявити присутність транспортного засобу та визначити, чи слід відкрити чи закрити шлагбаум.

Тоді потрібно буде написати програму для Arduino, використовуючи мову програмування Arduino. Ця програма буде відповідати за обробку вхідних даних від датчиків і прийняття рішення про відкриття чи закриття шлагбаума.

Нарешті, також можете включити світлодіод RGB для візуальної індикації стану шлагбаума (відкрито/закрито).

Важливо зазначити, що для створення автоматичної паркувальної бар'єрної системи за допомогою Arduino потрібні передові знання в області електроніки та програмування мікроконтролерів, а також належне планування та проектування, щоб забезпечити безпеку паркувального місця. Крім того, система потребуватиме регулярного обслуговування та оновлень, щоб забезпечити її належне функціонування.

У цьому проєкті змодельюємо автоматизований паркувальний бар'єр за допомогою Arduino. Ця модель відновлює загальну роботу автоматизованих систем, які дозволяють отримати доступ до громадських парків, які можна знайти на вокзалах, в аеропортах, кінотеатрах, супермаркетах тощо.

Шлагбаум відкривається за допомогою серводвигуна, коли звуковий датчик HC SR04 виявляє транспортний засіб, і автоматично закривається, якщо ні.

Програма для Arduino на тему Система автоматичного пропуску транспортних засобів на Arduino Uno, яка використовує ультразвуковий датчик для вимірювання відстані до об'єкта. Якщо об'єкт знаходиться на відстані 10 см або ближче, спрацьовує певна реакція: включається зумер, вмикається зелений світлодіод, сервопривід відкриває «бар'єр» на 90 градусів. Якщо об'єкт віддалений – система повертається у вихідний стан. Далі розглянемо детально, що робить кожна частина коду.

На початок підключається бібліотека Servo.h, яка дозволяє керувати сервоприводами. Потім оголошуються піні для ультразвукового датчика: trigPin (вивід для запуску імпульсу) і echoPin (вивід для прийому сигналу, що відбився). Далі створюється об'єкт myServo – це наш сервопривід.

Після цього визначаються пін для зумера (buzzerPin) і два піні для RGB-світлодіода – червоний (redPin) та зелений (greenPin).

Також створюються змінні duration та distance. Перша зберігає тривалість імпульсу, друга – розраховану відстань до об'єкта.

У функції setup () відбувається початкова ініціалізація: відкривається серійний порт для виводу в монітор, налаштовуються піні ультразвукового датчика (один як вихід, інший як вхід), підключається сервопривід до порта 8, задаються піні зумера та світлодіодів як виходи. Початково вмикається червоний світлодіод, щоб сигналізувати про «закритий стан».

У функції loop () система постійно повторює цикл дій. Спочатку очищається пін trigPin, потім на дуже короткий час на ньому подається сигнал після чого створюється імпульс, який запускає вимірювання. Відбитий сигнал

приймається піном echoPin, і за допомогою pulseIn () визначається тривалість цього імпульсу. З неї обчислюється відстань в сантиметрах, яка виводиться у серійний монітор.

Далі йде перевірка: якщо відстань менше або дорівнює 10 см, виконується серія дій – вмикається зумер (низький сигнал на пін, бо активний LOW), вмикається червоний світлодіод, вмикається зелений, сервопривід відкриває прохід (поворот на 90 градусів), очікується 5 секунд, після чого зелений світлодіод вмикається.

Якщо ж відстань більша за 10 см – сервопривід повертається у початкове положення (0 градусів), зумер вмикається (високий сигнал на пін), червоний світлодіод знову загоряється.

В кінці циклу встановлюється коротка затримка у 500 мс перед новим вимірюванням.

3.2 Робота сервоприводу для відкриття/закриття шлагбаума

Шлагбаум, також відомий як автомобільний обмежувач, був запозичений з-за кордон. Англійська назва – Barrier Gate. Це каналне обладнання для управління в'їздом і виїздом, яке обмежує рух автотранспорту на дорогах. Зараз він широко використовується на платних автомагістралях, автостоянках, у громад, на підприємствах та в установах. Біля воріт, для управління в'їздом і виїздом транспортних засобів. Електричний бар'єр може бути реалізований лише за допомогою пульта дистанційного керування, а також може автоматично управлятися системою управління автостоянкою (тобто системою управління картками IC). Відповідно до місця використання шлагбаумів, стовпи воріт можна розділити на прямі стовпи, вигнуті на 90 градусів, складені на 180 градусів стовпи та огорожі.

Шлагбаум складається з коробки передач, двигуна, (або гідравлічного) механізму передачі, балансувального пристрою, шасі, кронштейна гальмівної тяги і гальмівної тяги.

З постійним розвитком автомобільної промисловості і постійно зростаючою кількістю автомобілів, попит на шлагбауми також зростає, і висуваються більш високі вимоги до експлуатаційних характеристик шлагбаумів. Такі як інтегрований рух дорожніх воріт, пристрій зчеплення, інтелектуальна функція антипідйому, пристрій повернення в разі опору, функція обігріву (для забезпечення використання в середовищі мінус 40 градусів), система вентиляційного охолодження (для своєчасного зниження температури двигуна), пристрій автоматичного зчеплення, пристрій для запобігання зіткненням De-rod тощо. Швидкість підйому та опускання штанги становить від 0,8 секунди до 6 секунд. Довжина стовпа становить 0-7 метрів, залежно від фактичної ширини каналу на місці.

Шлагбауми все більше розвиваються до високотехнологічних, щоб задовольнити ринковий попит на продукцію, і стають ключовим обладнанням смуги руху в управлінні системами паркування.

Після вимкнення живлення шлагбаума з різних причин важіль гальма можна вручну підняти або опустити за допомогою ключа перемикача розподіленого зчеплення, щоб забезпечити нормальну роботу функції управління каналом транспортного засобу навіть у разі відключення електроенергії.

У процесі відчинення або зачинення воріт електрична частина керування воріт автоматично реєструє час роботи двигуна. Коли час роботи двигуна перевищує задане значення, вихідний сигнал керує зупинкою двигуна, щоб запобігти руху важеля воріт через несправність фотоелектричного пристрою виявлення Перевищення граничного положення може призвести до розбиття транспортного засобу або пошкодження шлагбаума, що підвищує надійність роботи системи.

Під час роботи шлагбаума, незалежно від того, чи перебуває важіль воріт у процесі відкриття, закриття або зупинки, якщо подається сигнал відкриття воріт, ворота негайно виконують дію відкриття (вгору), щоб запобігти розбиттю транспортного засобу з різних причин.

Налаштувавши спеціальний потрійний електронний вимикач та електронний вимикач дистанційного керування, вимикачем та зупинкою шлагбаума можна керувати дистанційно та віддалено.

Електрична частина управління шлагбаумом може бути оснащена стандартним інтерфейсом R485(у нашому випадку запрограмована плата ардуіно), а перемикачем і зупинкою шлагбаума можна керувати за допомогою віддаленого комп'ютера або іншого обладнання для реалізації дистанційного автоматичного управління.

3.3 Відправка даних на сервер або ПК

Запропонована система містить три основні частини: центр виявлення паркування, центр моніторингу паркування та глобальний центр управління інформацією.

Центр виявлення паркування складається в основному з вузлів гібридних датчиків які встановлюються на кожному паркувальному місці в кожній зоні. Ці сенсорні вузли утворюють бездротову сенсорну мережу (WSN), що дозволяє збирати стани всіх паркувальних місць (доступних або зайнятих) для надсилання їх на шлюз (приймач) цієї зони, ця інформація потім буде надіслана на центральний сервер для збереження в глобальній базі даних.

Центр моніторингу паркування відповідає за ідентифікацію та перевірку автомобілів, які щойно припаркувались на зарезервованому або вільному місці. Цей центр використовує технологію RFID для контролю та моніторингу, з одного боку, припаркованих автомобілів, а з іншого боку, для ідентифікації та керування оплатою часу паркування.

Глобальний центр управління інформацією – це база даних, де вся інформація, виявлена та зібрана з усіх автостоянок у місті, записується та використовується в режимі реального часу за допомогою веб або мобільних додатків. Таким чином, водії матимуть всю інформацію про вільні місця на

всіх автостоянках міста, ознайомлюватися з цими місцями відповідно до їх призначення та оплачувати паркування.

На рисунку 3.1 показана архітектура запропонованої системи розумного паркування:

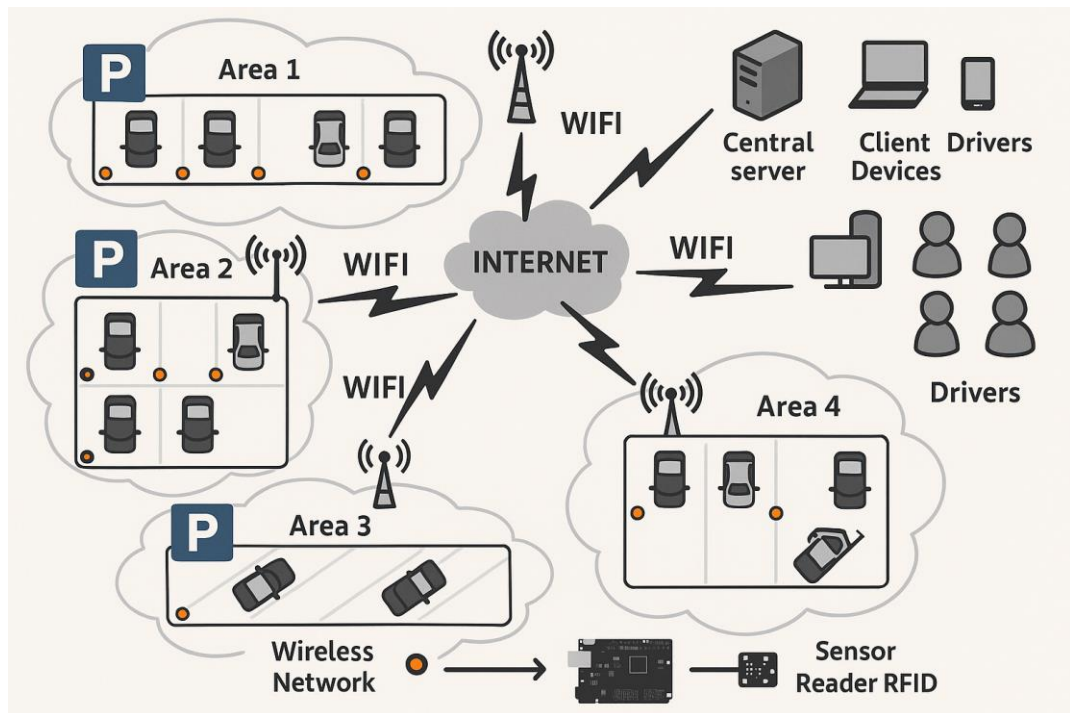


Рисунок 3.1 – Пропонована архітектура система розумного паркування на основі WSN

Центр виявлення паркування використовує дві останні технології: бездротові сенсорні мережі (WSN) і технологію RFID. Формування сенсорної мережі змінюється відповідно до типу паркування на території. Для лінійних автостоянок у мережі буде сформовано ланцюгову топологію, а з іншого боку, мережева топологія у формі кластера буде створена на масових автостоянках. Формування різних топологій базується на виконанні гібридного алгоритму самоорганізації, який адаптується до типу та структури паркування, що

дозволяє формувати або ланцюг (рис. 3.2), або кластери (рис. 3.3), відповідно до розподілу вузлів і того, як вони розкидані в зоні паркування.

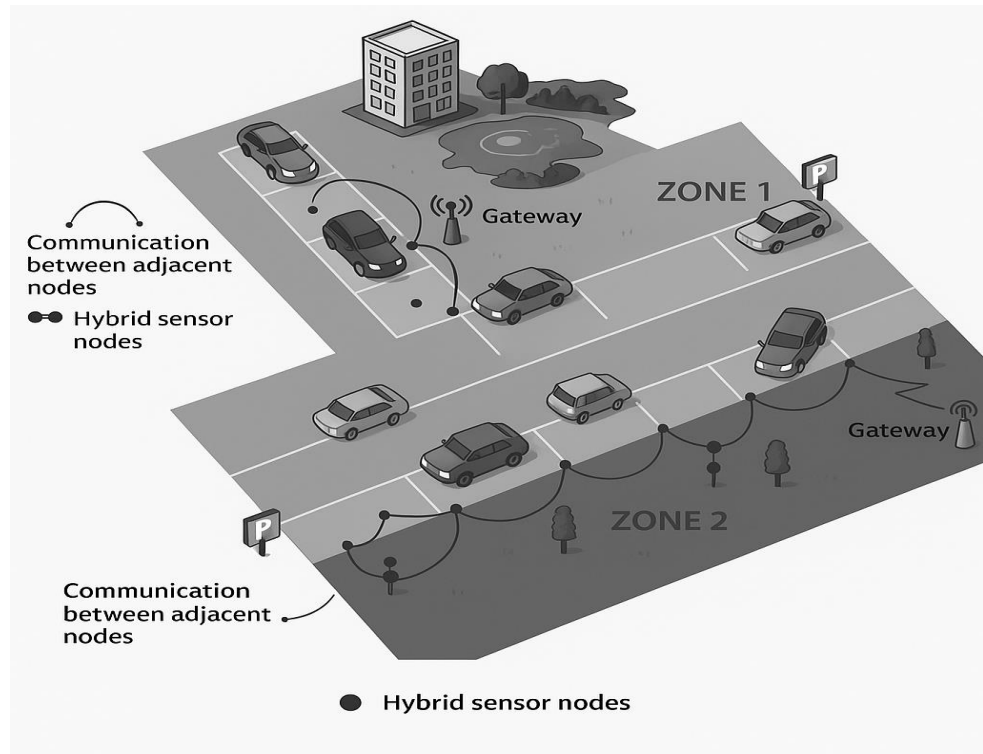


Рисунок 3.2 – Формування топології ланцюга в лінійній парковці

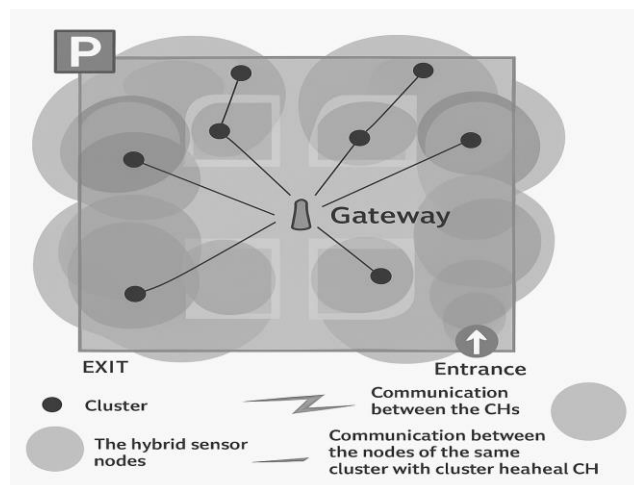


Рисунок 3.3 – Формування топології кластера в зоні масового паркування

Щоб сформувати топологію мережі за допомогою вузлів датчиків на відкритій парковці, усі вузли надсилають свої координати на шлюз, який виконує алгоритм 1, що дає змогу обчислити та визначити тип топології, яка має бути сформована. У випадку лінійного паркування шлюз виявляє, що всі вузли мають однакову координату X або однакову координату Y як функцію розподілу вузлів у зоні паркування. У цьому випадку шлюз просить вузли створити ланцюгову топологію в мережі, щоб мінімізувати споживання електроенергії між вузлами за допомогою багатопрохідного зв'язку зі шлюзом (рис. 3.4).

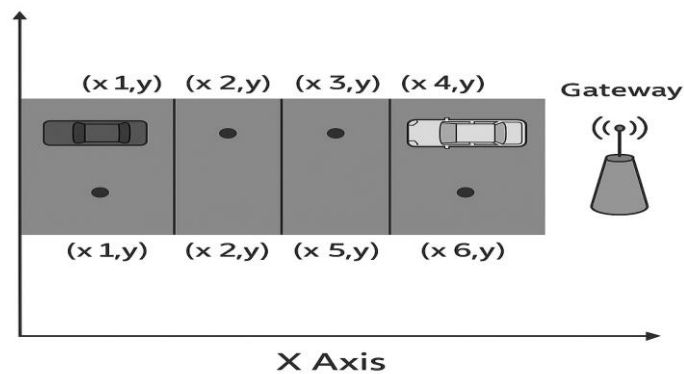


Рисунок 3.4 – Приклад розподілу вузлів датчиків у лінійній зоні паркування

3.4 Аналіз результатів тестування

Аналіз результатів тестування системи, побудованої на основі ультразвукового датчика, сервоприводу, зумера та RGB-світлодіодів, дозволив оцінити її працездатність, чутливість, швидкодію та стабільність функціонування в реальних умовах експлуатації. У ході тестування було перевірено, як система реагує на різні відстані до об'єкта, наскільки точно та швидко визначається момент наближення об'єкта до межі у 10 сантиметрів, і

як злагоджено працюють окремі компоненти у відповідь на зміну вхідних параметрів.

Система автоматичного пропуску транспортних засобів, побудована на базі плати Arduino UNO від Keystudio, призначена для контролю доступу автомобілів через шлагбаум за допомогою сенсорів і виконавчих механізмів. Основним елементом виявлення транспортного засобу є ультразвуковий датчик HC-SR04, який вимірює відстань до об'єкта, надсилаючи звуковий імпульс і фіксуючи час його повернення. Коли датчик фіксує автомобіль на заданій відстані, мікроконтролер обробляє цей сигнал і активує інші модулі системи.

Як виконавчий механізм для підняття та опускання шлагбаума використовується серводвигун SG90. Він отримує керуючий сигнал від Arduino та повертається на потрібний кут, наприклад 90 градусів, що імітує відкриття проїзду. Після проходження транспортного засобу сервомотор повертається у початкове положення, і шлагбаум опускається. Уся система зібрана на макетній безпачній платі MB-102, що дозволяє швидко та зручно з'єднувати всі елементи без потреби пайки, спрощуючи розробку й тестування.

Для візуальної індикації стану системи використовується RGB світлодіодний модуль KY-016. Залежно від ситуації, Arduino змінює його колір: червоний – шлагбаум закрито, зелений – дозвіл на проїзд. Крім того, для звукової індикації застосовується активний модуль з динаміком KY-012, який подає короткий звуковий сигнал при виявленні автомобіля або при зміні стану шлагбаума. Усі сигнали обробляються Arduino, яка керує логікою роботи системи, приймаючи рішення на основі даних з сенсора та подаючи відповідні команди на серводвигун, світлодіод і динамік.

Таким чином, система забезпечує повністю автоматизоване керування доступом транспортних засобів, є зручною у використанні, легко модифікується, а також демонструє ефективне застосування мікроконтролерної техніки у реальних умовах(рис. 3.5-3.6).



Рисунок 3.5 – Загальний вигляд розробленого пристрою



Рисунок 3.6 – Імітація відкриття автоматичного пропуску

Загалом результати тестування підтвердили, що система працює надійно, швидко реагує на зміну відстані, а всі елементи (ультразвуковий датчик, сервопривід, зумер і RGB-світлодіоди) синхронізовані коректно. Отримані результати свідчать про готовність системи до реального застосування, зокрема в системах автоматичного доступу або безконтактного керування об'єктами. За потреби, можливе розширення функціоналу шляхом додавання таймерів, фільтрації шумів чи інтеграції з бездротовими модулями для дистанційного контролю.

ВИСНОВКИ

За результатами проведеного дослідження зроблено відповідні висновки.

Здійснено аналіз існуючих систем автоматичного пропуску транспортних засобів, що працюють на базі мікроконтролерів. Було розглянуто їхні основні функціональні можливості, переваги та недоліки, що дозволило визначити оптимальний підхід для реалізації власної системи.

Виконано проєктування системи автоматичного пропуску транспортних засобів, яка базується на мікроконтролері Arduino Uno. Було розроблено схему підключення ультразвукового датчика, сервоприводу, RGB-світлодіода та звукового модуля, а також визначено алгоритм їхньої роботи.

Спроектовано та реалізовано систему обробки даних з ультразвукового датчика, яка реагує на виявлення транспортного засобу на заданій відстані (наприклад, 20 см). У відповідь сервопривід автоматично відкриває шлагбаум, RGB-світлодіод змінює колір для індикації стану, а звуковий модуль видає сигнал.

Розроблено механізм керування шлагбаумом за допомогою сервоприводу, який забезпечує швидке та точне відкривання і закривання проїзду. Візуальна та звукова індикація стану системи підвищує зручність її використання.

Реалізовано програмне забезпечення на Arduino Uno, яке забезпечує синхронізацію роботи датчиків, сервоприводу, світлодіодів і звукового модуля. Це дозволяє системі працювати в реальному часі з високою точністю.

Отримано працездатну систему автоматичного пропуску транспортних засобів, яка може бути використана для автоматизації паркувальних зон чи контрольно-пропускних пунктів. Вона забезпечує швидке реагування на виявлення транспортного засобу, надійне керування шлагбаумом та чітку індикацію стану системи.

На основі проведеного тестування можна зробити висновок, що запропонована система є ефективною для автоматизації доступу транспортних засобів. Розроблену систему можна використовувати як окремий пристрій або інтегрувати в більш складні системи управління паркуванням чи контролю доступу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Managing Next-Generation Networks for Intelligent Services and Applications. *CommunicationsSociety*. URL: <https://surli.cc/seduqo> (дата звернення: 10.02.2025).
2. Інтернет речей - нескінченні можливості. *XpertDigital*. URL: <https://surli.li/iuleke> (дата звернення: 16.02.2025).
3. Що таке контроль доступу? *ContinuumGRC*. URL: <https://surli.lu/qbyzev> (дата звернення: 20.02.2025).
4. Role-Based Access Control Models and Implementation Perspectives. *Slide Serve*. URL: <https://surli.cc/ikubqt> (дата звернення: 16.02.2025).
5. Saltzer and Schroeder's design principles. *NoComplexity*. URL: <https://surli.li/fawtwh> (дата звернення: 06.03.2025).
6. A Complete Guide to Parking Access Control System. *Mgiaccess*. URL: <https://surli.lu/hordrr> (дата звернення 10.03.2025).
7. Attribute-Based Access Control (ABAC). *BlueEraSofttech*. for AWS URL: <https://surli.lu/stpyth> (дата звернення 15.03.2025).
8. Cyber-Physical Systems. *NationalScienceFoundation*. (CPS) URL: <https://surli.cc/xppeye>. (дата звернення 15.03.2025).
9. A Privacy-Preserving Architecture and Data-Sharing Model for Cloud-IoT Applications. *IEEEExplore*. URL: <https://surli.li/awhtnt> (дата звернення 20.03.2025).
10. Access Control List (ACL) in cyber security: Beneficial for all, critical for some. *DataGuard*. URL: <https://surli.li/oowkmc> (дата звернення 16.03.2025).
11. IaC Best Practices: Developer Guide. *Daily.dev*. 2024 URL: <https://surli.cc/eleqrl> (дата звернення 25.03.2025).
12. The CE-IoT Framework for Green ICT Organizations. *CircularEconomyAlliance*. URL: <https://surli.gd/icuqft> (дата звернення 05.04.2025).

13. A Smart Real-Time Parking Control and Monitoring System. *MDPI*. URL: <https://surl.li/aviovt> (дата звернення 09.04.2025).
14. 12 Mind-Blowing Arduino AI Projects That Will Transform Your Tech Skills. *Jaycon*. URL: <https://surl.li/urwxlq> (дата звернення 16.04.2025).
15. Automatic Car Parking Assistance using Arduino. *Researchgate*. URL: <https://surl.li/abjtfl> (дата звернення 25.04.2025).
16. Parking sensor. *ProjectHub*. URL: <https://surl.li/czxrlq> (дата звернення 01.05.2025).
17. Key Advantages and Disadvantages of Automation. *SimpliLearn*. URL: <https://surl.li/zweeee> (дата звернення 02.05.2025).
18. Arduino Uno. *Components101*. URL: <https://surl.li/fsjogf> (дата звернення 04.05.2025).
19. Arduino IDE Arduino Integrated Development Environment (IDE) v1. *GeeksForGeeks*. URL: <https://surl.li/ujorjt> (дата звернення 06.05.2025).
20. What power supply can I use with my Arduino board? *Arduino.cc*. URL: <https://surl.li/msupsz> (дата звернення 08.05.2025).
21. Interfacing of ultrasonic Sensor. *Hackster.io*. URL: <https://surl.gd/nbeirh> (дата звернення 09.05.2025).
22. Servo Control. *Armonitor*. URL: <https://surl.li/pbcodv> (дата звернення 14.05.2025).

.