

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 АВТОСТАРТУ ДВИГУНА НА БАЗІ
МІКРОКОНТРОЛЕРА**

MICROCONTROLLER-BASED ENGINE AUTO-START SYSTEM

спеціальність 123 Комп'ютерна інженерія
(шифр і назва спеціальності)

освітня програма Комп'ютерна інженерія
(назва освітньої програми)

Виконав: здобувач вищої освіти
групи КІ-41
Чепелінський Орест Андрійович

(підпис)

Керівник:
асистент
Конкевич Людмила Миколаївна

(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

Чепелінському Оресту Андрійовичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи Система автостарту двигуна на базі мікроконтролера

Керівник роботи асистент Конкевич Людмила Миколаївна

затверджені 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

Огляд системи автотранспорту

Апаратне забезпечення системи автотранспорту

Проектування та тестування системи автотранспорту

Висновки

5. Перелік графічного (ілюстративного) матеріалу:

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
<i>Огляд системи автотранспорту</i>	<i>Конкевич Л.М., асистент</i>		
<i>Апаратне забезпечення системи автотранспорту</i>	<i>Конкевич Л.М., асистент</i>		
<i>Проектування та тестування системи автотранспорту</i>	<i>Конкевич Л.М., асистент</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, огляд системи автотранспорту</i>	до 10.02.2025 р.	Виконано
2.	<i>Апаратне забезпечення системи автотранспорту</i>	до 02.03.2025 р.	Виконано
3.	<i>Проектування та тестування системи автотранспорту</i>	до 02.04.2025 р.	Виконано
4.	<i>Висновки та пропозиції</i>	до 10.04.2025 р.	Виконано
5.	<i>Формування списку використаних джерел</i>	до 15.04.2025 р.	Виконано
6.	<i>Формування додатків</i>	до 02.05.2025 р.	Виконано
7.	<i>Оформлення ілюстративного матеріалу</i>	до 10.05.2025 р.	Виконано
8.	<i>Представлення остаточного варіанту кваліфікаційної роботи бакалавра 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 кафедрі</i>	до 10.06.2025 р.	Виконано

Здобувач вищої освіти

(підпис)

Чепелінський О.А.

(прізвище, ініціали)

Керівник кваліфікаційної роботи

(підпис)

Конкевич Л.М.

(прізвище, ініціали)

АНОТАЦІЯ

Чепелінський О.А. Система автостарту двигуна на базі мікроконтролера.
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.

У першому розділі проаналізовано принципи роботи сучасних систем дистанційного запуску двигуна, класифікацію типових рішень, а також особливості застосування бездротових технологій керування.

Другий розділ присвячено технічним складовим системи: характеристикам мікроконтролера Arduino Nano, релейних модулів, RF-приймача та допоміжних елементів.

У третьому розділі описано процес складання прототипу на макетній платі, схему підключення, програмну реалізацію алгоритму автостарту, а також тестування розробленої системи.

Результатом дослідження є працездатна модель системи автостарту, яка може бути використана як доступне рішення для керування запуском двигуна.

Ключові слова: автостарт, Arduino Nano, реле, RF-модуль, двигун, дистанційне керування.

ANNOTATION

Chepelinskyi O. Microcontroller-based engine auto-start system. Manuscript.

Qualifying work of a bachelor of EP «Computer Engineering» specialty 123 Computer Engineering. Lutsk National Technical University. Lutsk, 2025.

Qualification work consists of an introduction, three sections, conclusions, a references.

The first chapter analyzes the principles of modern remote engine start systems, the classification of common solutions, and the features of wireless control technologies.

The second chapter focuses on the technical components of the system: specifications of the Arduino Nano microcontroller, relay modules, RF receiver, and auxiliary elements.

The third section describes the process of assembling a prototype on a breadboard, the connection diagram, the software implementation of the autostart algorithm, and the testing of the developed system.

The result of the research is a workable model of the autostart system, which can be used as an affordable solution for controlling engine start.

Keywords: auto-start, Arduino Nano, relay, RF module, engine, remote control.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 ОГЛЯД СИСТЕМ АВТОСТАРТУ	8
1.1 Історія розвитку систем автостарту двигуна.....	8
1.2 Принцип дії та типи комерційних систем автозапуску.....	9
1.3 Технології бездротового керування	12
1.4 Безпека при автозапуску	14
РОЗДІЛ 2 АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ АТОСТАРТУ	17
2.1 Вибір мікроконтролера та його можливості	17
2.2 Реле для керування ланцюгами запалювання і стартера	19
2.3 Радіочастотний модуль для бездротового керування	21
2.4 Вибір елементів для імітації роботи системи автозапуску.....	23
2.5 Забезпечення електробезпеки та живлення.....	24
РОЗДІЛ 3 ПРОЄКТУВАННЯ ТА ТЕСТУВАННЯ СИСТЕМИ	26
3.1 Загальна архітектура та принцип роботи системи	26
3.2 Схемотехнічне рішення системи	28
3.3 Розробка програмного забезпечення.....	31
3.4 Процес збирання та монтажу прототипу.....	34
3.5 Тестування системи	36
3.6 Аналіз та вдосконалення	39
ВИСНОВКИ.....	41
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	43

ВСТУП

Актуальність теми. Зростаюча популярність систем дистанційного керування автомобілями та потреба в підвищенні комфорту та безпеки експлуатації транспорту зумовлює необхідність розробки доступних і надійних рішень автостарту. Використання мікроконтролерів, зокрема Arduino, дає змогу створити недорогу, гнучку та функціональну систему автозапуску.

Метою роботи є розробка та тестування системи автоматичного запуску двигуна автомобіля на базі мікроконтролера.

Об'єкт дослідження – електронна система дистанційного автостарту двигуна.

Предмет дослідження – апаратна та програмна реалізація системи автостарту із застосуванням мікроконтролера.

Завдання, які необхідно виконати:

- провести огляд систем автостарту;
- визначити вимоги до системи;
- підібрати необхідні апаратні компоненти;
- розробити схему підключення;
- зібрати модель на макетній платі та провести тестування.

РОЗДІЛ 1

ОГЛЯД СИСТЕМ АВТОСТАРТУ

1.1 Історія розвитку систем автостарту двигуна

Передумови виникнення систем автостарту двигуна пов'язані з багатьма чинниками, які охоплюють історичний, технологічний та соціальний аспекти розвитку автомобільної індустрії. Вже на ранніх етапах еволюції транспортних засобів з двигунами внутрішнього згоряння гостро стояла проблема зручного, безпечного та надійного запуску. Механічні стартери, що застосовувалися на початку ХХ століття, потребували фізичних зусиль від водія й несли ризики, пов'язані з травмами через зворотний удар. Це стимулювало пошук нових рішень, які дозволили б автоматизувати процес запуску двигуна [1]. Згодом впровадження електричних стартерів стало важливою віхою, однак потреба у ще більш комфортних та безпечних рішеннях залишалася актуальною, особливо в регіонах із суворими кліматичними умовами.

General Motors (GM) [1] стала одним із перших великих автовиробників, хто почав встановлювати заводські системи дистанційного запуску у моделях Cadillac, Chevrolet. Також компанії Toyota [2], Ford [3] і Nissan [4] згодом інтегрували власні системи Remote Start у штатне обладнання. Функціональність таких систем доповнилася захистом від несанкціонованого доступу, можливістю роботи із сигналізацією, інтеграцією з центральним замком тощо.

Наступним етапом розвитку стало впровадження CAN-шинового інтерфейсу, що дозволило точно взаємодіяти з електронними блоками керування автомобіля без фізичного втручання в систему запалювання. Такі рішення стали поширеними в сучасних цифрових автомобілях, де втручання в електропроводку обмежене або небажане [5].

У 2010-х роках відбулося масове впровадження GSM та GPS технологій. З'явилися мобільні застосунки для смартфонів, за допомогою яких користувачі могли запускати двигун з будь-якої точки світу, контролювати стан авто,

задавати умови запуску (наприклад, температуру в салоні), а також отримувати зворотний зв'язок. Такі системи поєднують функції сигналізації, трекінгу, діагностики та дистанційного керування [6].

Паралельно розвивалася і DIY-сфера – ентузіасти почали створювати власні системи автозапуску на базі відкритих апаратних платформ, таких як Arduino, Raspberry Pi та ESP. Це дало змогу створювати дешевші, але функціональні рішення, адаптовані до конкретних потреб користувачів.

Таким чином, технологія автозапуску еволюціонувала від простих таймерних систем до інтелектуальних комплексів, інтегрованих у загальну інфраструктуру автомобіля, що демонструє високий рівень автоматизації та комфорту в експлуатації сучасного транспорту.

1.2 Принцип дії та типи комерційних систем автозапуску

Системи дистанційного автозапуску двигуна відіграють важливу роль у сучасному автомобілебудуванні та активно використовуються як у серійному виробництві транспортних засобів, так і в умовах вторинного дообладнання. Основна мета цих систем полягає в автоматичному запуску двигуна за допомогою зовнішнього сигналу або заданого алгоритму без фізичної присутності водія. Функціональність автозапуску втілюється через контроль систем запалювання, іммобілайзера та бортової електромережі, забезпечуючи комфортну температуру в салоні або запуск двигуна при низькому заряді акумулятора.

Системи дистанційного запуску двигуна умовно поділяються за архітектурною побудовою на кілька основних типів: однобортові (Stand-Alone), CAN-базовані (інтегровані в шину Controller Area Network) та саморобні (DIY) системи на базі мікроконтролерів, зокрема Arduino. Кожен тип має власні особливості побудови, рівень складності інтеграції з автомобілем, вартість та функціональні можливості. Порівняння типів систем дистанційного запуску двигуна наведені в таблиці 1.1.

Таблиця 1.1 – Порівняння типів систем автозапуску [7-9]

Тип системи	Переваги	Недоліки	Застосування	Приклад
Однороб- това	Простота монтажу, низька вартість, автономність	Обмежена функціональність, складна інтеграція з новими авто	Бюджетні рішення для старих авто	Scher-Khan Magicar A, Tomahawk 9.9, Alligator S-275
CAN- базована	Висока сумісність, багатофункціональ- ність, доступ до штатної електроніки	Висока вартість, складність встановлення	Професійна установка в сучасні авто	GAZER S5
DIY (Arduino)	Гнучкість, навчальний потенціал, низька вартість	Не сертифіковано, потребує програмування, немає техпідтримки	Освітні цілі, прототипи, персональні розробки	Системи на Arduino, Raspberry Pi та ESP

Одноробтові системи – це автономні модулі, які не потребують прямої інтеграції з електронною шиною автомобіля. Вони зазвичай працюють за принципом симуляції сигналів запалювання та стартера шляхом замикання відповідних контактів. Такі системи відносно прості в монтажі, але можуть бути обмежені в сумісності з сучасними транспортними засобами, які мають складну електроніку [7].

CAN-базовані системи інтегруються в електронну архітектуру автомобіля через шину CAN (Controller Area Network), що дозволяє точно керувати штатними функціями (запалювання, блокування дверей, датчики). Такі системи часто підтримуються офіційними виробниками сигналізацій, мають високу надійність і функціональність, але вимагають більшого досвіду при встановленні [8]. Реалізація таких систем полягає у використанні мікроконтролера, який отримує сигнали з брелока або мобільного застосунку, обробляє їх та керує відповідними реле або транзисторними ключами. Інтерфейс CAN дає змогу взаємодіяти з електронними блоками управління автомобіля без потреби прямого втручання у силову проводку. Такий підхід дозволяє зменшити складність інсталяції та підвищити надійність роботи системи.

На українському ринку присутні як імпортні, так і локальні виробники систем автозапуску. До провідних українських брендів, що пропонують подібні рішення, належать Gazer GSM CAN – один із найсучасніших прикладів систем з підтримкою GSM-керування та інтеграцією з CAN-шиною. Завдяки мобільному застосунку та гнучкому програмному забезпеченню, системи Gazer дозволяють віддалено запускати двигун, контролювати стан дверей, акумулятора, а також налаштовувати сценарії запуску за температурою або напругою [9].

Принцип дії таких систем зводиться до запуску двигуна через замикання ланцюга запалювання або симуляції натискання кнопки «Start/Stop». Для обходу штатного імобілайзера використовуються спеціальні модулі з вбудованими RFID-ключами або програмними алгоритмами, що зчитують код доступу через CAN-шину. Крім того, новітні реалізації інтегрують функції моніторингу температури, напруги бортової мережі, рівня пального та позиціонування, створюючи комплексну систему керування транспортним засобом на відстані.

Перевагами таких систем є зручність експлуатації, можливість запуску двигуна в холодну пору року без виходу з приміщення, а також підвищення рівня безпеки за рахунок інтеграції з охоронними модулями. До недоліків слід віднести можливі труднощі з сумісністю на окремих моделях автомобілів, складність інсталяції без спеціалізованого обладнання, а також потенційні ризики при втраті зв'язку або збої в програмному забезпеченні.

DIY-системи на Arduino або інших мікроконтролерах дозволяють створити бюджетне рішення, адаптоване під конкретні вимоги користувача. Вони є ідеальними для навчання, прототипування або персонального використання, однак потребують базових знань у сфері програмування та електроніки, а також не завжди відповідають сертифікованим стандартам безпеки.

Таким чином, ринок систем автозапуску в Україні динамічно розвивається, демонструючи високий рівень локалізації та відповідності сучасним тенденціям електронного керування автомобілями.

1.3 Технології бездротового керування

Розвиток систем дистанційного запуску двигуна безпосередньо пов'язаний з еволюцією бездротових технологій. Безпроводне керування дозволяє користувачу активувати запуск двигуна автомобіля на відстані без фізичного контакту із засобом пересування. Найпоширенішими каналами зв'язку для таких систем є радіочастотна (RF) комунікація, Bluetooth-з'єднання, а також GSM/мобільні мережі. Кожна з технологій має власну архітектуру, діапазон дії, рівень енергоспоживання та безпеки, що визначає її застосування у різних класах автозапускових систем.

RF-зв'язок є найпоширенішим способом реалізації дистанційного запуску. Комунікація здійснюється за допомогою радіосигналу в діапазоні 315/433 МГц, який передається з пульта керування на приймач, вбудований у систему автозапуску. Дальність дії зазвичай становить від 50 до 1000 метрів, залежно від потужності сигналу, перешкод та характеристик навколишнього середовища [10]. Bluetooth-комунікація використовується у системах, що передбачають керування через мобільні пристрої в межах ближнього радіусу (до 10-50 м). Пристрій користувача підключається до контролера автозапуску через Bluetooth-модуль (наприклад, HC-05), після чого можливе керування запуском через спеціальний мобільний додаток або інтерфейс [11]. GSM-зв'язок передбачає використання SIM-картки, встановленої у GSM-модулі, який підключений до контролера. Користувач надсилає керуючі команди через SMS або мобільний додаток, а модуль активує відповідні функції системи [12].

Разом з тим, у новітніх розробках все ширше впроваджується концепція Інтернету речей (Internet of Things, IoT), яка забезпечує глибшу інтеграцію автомобільних систем у цифрове середовище. На відміну від класичних рішень,

IoT-підхід базується на хмарних технологіях, мобільних застосунках, геолокаційних сервісах та мережевих протоколах (зокрема MQTT, HTTPS), що дозволяє реалізувати не лише дистанційний запуск, а й інтелектуальне керування сценаріями запуску, моніторингом параметрів авто, а також двосторонню синхронізацію з іншими пристроями або сервісами [13].

Мікроконтролери з підтримкою Wi-Fi або GSM, такі як ESP32, STM32 у поєднанні з відповідними IoT-платформами (Blynk, ThingsBoard, AWS IoT Core), здатні не тільки приймати команди запуску, але й аналізувати контекст – температуру повітря, час доби, стан акумулятора, геолокацію користувача [14]. Це дозволяє, наприклад, автоматично активувати попередній прогрів двигуна за певної температури або відправити сповіщення при виявленні аномалій. Усе це відбувається з можливістю віддаленого оновлення прошивки та налаштувань системи через хмару, що значно підвищує масштабованість і безпеку.

У таблиці 1.2 наведено порівняльну характеристику бездротових технологій, що використовуються в системах автозапуску двигуна.

Таблиця 1.2 – Характеристики основних технологій дистанційного керування автозапуском двигуна [10-16]

Характеристика	Радіочастотне (RF)	Bluetooth	GSM	Інтернет речей (IoT)
Канал зв'язку	315/433 МГц	Bluetooth 2.0/4.0	Мобільна мережа (GSM)	GSM, Wi-Fi, LTE, MQTT
Дальність дії	50-1000 м	10-50 м	Необмежена (залежить від покриття)	Необмежена
Переваги	Низька затримка, простота, незалежність від мережі	Енергоефективність, швидке з'єднання, автономність	Глобальне покриття, зворотний зв'язок, стабільність	Хмарна інтеграція, автоматизація, аналітика
Недоліки	Обмежена дальність, перешкоди, можливість перехоплення	Мала дальність, потреба бути поруч, без глобального доступу	Залежність від оператора, енергоспоживання, потреба у SIM	Складність, потреба у захисті, залежність від хмар
Сфера застосування	Бюджетні системи, DIY, базові охоронні комплекси	Особистий транспорт, proximity unlock	Преміум-системи, телематика, вантажні ТЗ	Інтелектуальні авто, Smart City, Smart Car

Таким чином, вибір технології залежить від цілей проєкту: RF підходить для локального простого керування, GSM – для глобального доступу й телематики, а Bluetooth – для зручності в межах безпосередньої близькості. У рамках реалізації DIY-систем на базі Arduino переважно використовуються RF-модулі через простоту інтеграції та доступність компонентів.

1.4 Безпека при автозапуску

Забезпечення безпеки транспортного засобу під час використання функції автозапуску є критичним аспектом розробки будь-якої системи дистанційного запуску двигуна. У зв'язку з потенційними ризиками несанкціонованого доступу, втручання в електроніку автомобіля або запуску у небезпечних умовах, сучасні автозапускові системи інтегрують ряд механізмів безпеки. Основні складові таких механізмів (рис. 1.1) є перевірка правомірності команди запуску (автентифікація користувача), захист від перехоплення та несанкціонованого доступу (шифрування переданих команд), а також перевірка умов перед запуском, таких як закриті двері та капот, нейтральне положення коробки передач тощо [17].



Рисунок 1.1 – Основні складові механізмів безпеки

Автентифікація є базовим етапом у забезпеченні безпеки – вона передбачає підтвердження права користувача на керування системою. У

найпростіших реалізаціях застосовується статичний код у радіобрелоках, проте в сучасних рішеннях все частіше впроваджуються динамічні коди або двофакторна автентифікація через мобільні додатки. У деяких випадках використовується також ідентифікація за Bluetooth-профілем або SIM-карткою користувача [18].

Щоб уникнути перехоплення та несанкціонованого відтворення команд, передача даних між передавачем і приймачем повинна бути захищеною. Для цього використовуються алгоритми шифрування, наприклад AES або власні протоколи, що ускладнюють аналіз сигналу. У сучасних комерційних системах (наприклад, StarLine або Pandora) впроваджуються спеціалізовані криптостійкі протоколи з індивідуальним ключем пристрою [19].

Окрім захисту від зовнішніх втручань, важливо забезпечити технічну безпеку запуску. Система повинна перевіряти такі параметри, як:

- нейтральне положення коробки передач;
- закритість дверей, капоту, багажника;
- наявність і стабільність бортової напруги;
- статус охоронної сигналізації;
- положення стоянкового гальма.

У разі порушення будь-якої з умов система повинна або заборонити запуск, або автоматично зупинити двигун, якщо запуск уже здійснено. Також поширена реалізація зворотного зв'язку (feedback), коли користувач отримує підтвердження про статус запуску, температуру двигуна або активність тривожних датчиків. Ризики при автозапуску та засоби їх усунення наведені в таблиці 1.3.

Багато автозапускових систем працюють у тісній інтеграції із штатною або додатковою сигналізацією. Це дозволяє реалізувати спільну логіку блокувань двигуна, автоматичне увімкнення сирени або повідомлення на телефон у разі несанкціонованого відкриття авто під час запуску. Важливо також враховувати затримку у запуску та перевірку імітації присутності ключа, що необхідно для деяких моделей авто з іммобілайзером.

Таблиця 1.3 – Ризики при автозапуску та засоби їх усунення [17-19]

№	Можлива загроза	Причина	Засіб захисту / запобігання
1	Несанкціонований запуск двигуна	Використання підробленого сигналу або втраченого брелока	Унікальний код, динамічне шифрування, автентифікація користувача
2	Запуск на ввімкненій передачі (МКПП)	Відсутність перевірки положення коробки передач	Датчик нейтралі, програмна перевірка стану
3	Запуск із відкритими дверима, капотом або багажником	Немає контролю стану кінцевиків	Додаткові кінцевики, перевірка в логіці програми
4	Перехоплення бездротового сигналу	Слабке або відсутнє шифрування	Rolling code, 128/256-бітне шифрування (RF, GSM, BLE)
5	Витрата заряду акумулятора	Надмірне енергоспоживання в режимі очікування	Автоматичне відключення, сплячий режим мікроконтролера
6	Відмова запуску в холодну пору року	Недостатній заряд АКБ або густа олива	Контроль напруги, затримка запуску, підігрів перед стартом
7	Відсутність зворотного зв'язку про запуск	Користувач не знає, чи відбувся запуск	Звуковий/світловий сигнал, GSM-повідомлення, індикація на брелоку
8	Викрадення авто після запуску	Двигун працює, але двері не заблоковані	Автоматичне блокування керма/дверей, таймер блокування, PIN-код
9	Помилковий запуск при випадковому натисканні кнопки	Відсутність подвійного підтвердження	Комбінація кнопок, подвійне натискання, тайм-аут
10	Збої при поганому сигналі GSM	Перерваний зв'язок під час передачі команди	Повтор команди, підтвердження прийому, журнал подій

Таким чином, реалізація безпечної системи автозапуску вимагає комплексного підходу: від базового шифрування сигналів до повного моніторингу технічного стану автомобіля. Особливу увагу слід приділяти ергономіці та взаємодії з користувачем, щоб забезпечити як зручність керування, так і високу ступінь захисту від загроз несанкціонованого використання. У саморобних (DIY) системах, таких як Arduino-рішення, важливо ретельно продумати алгоритми контролю стану і, за можливості, реалізувати базові елементи автентифікації.

РОЗДІЛ 2

АПАРATНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ АТОСТАРТУ

2.1 Вибір мікроконтролера та його можливості

Проектування системи дистанційного автозапуску двигуна автомобіля вимагає ретельного підбору мікроконтролера, який є центральною обчислювальною одиницею та забезпечує взаємодію з периферійними пристроями. Мікроконтролер має обробляти сигнали від бездротового модуля, керувати релейними виконавчими елементами, здійснювати індикацію станів системи та контролювати логіку запуску двигуна відповідно до встановлених умов безпеки.

При виборі мікроконтролера враховувались такі технічні та експлуатаційні критерії:

- достатня кількість цифрових входів/виходів для підключення реле, RF-приймача та індикаційних елементів;
- компактні габарити плати, що дозволяє зручно інтегрувати систему у відсік автомобіля;
- низьке енергоспоживання, важливе для роботи в умовах живлення від акумулятора;
- наявність зручного середовища розробки та доступної документації;
- підтримка поширених протоколів зв'язку (UART, SPI, I²C);
- низька вартість та доступність на ринку.

На основі цих параметрів було обрано Arduino Nano – одну з найпоширеніших платформ для побудови компактних вбудованих систем. Arduino Nano побудований на базі мікроконтролера ATmega328P, що працює на частоті 16 МГц і має 32 КБ flash-пам'яті для збереження програми. Крім того, плата містить 2 КБ оперативної пам'яті (SRAM), що є достатнім обсягом для обробки простих логічних операцій, керування реле та обробки бездротових команд [20].

Серед ключових технічних характеристик Arduino Nano:

- 14 цифрових входів/виходів, з яких 6 підтримують ШІМ (PWM) – зручно для керування реле або сигналів індикації;
- 8 аналогових входів – дозволяє в майбутньому додавати контроль датчиків, наприклад температури двигуна;
- UART, SPI та I²C – підтримка основних протоколів, що забезпечує можливість розширення функціональності, наприклад, додавання GSM-модуля;
- живлення від 5 В або 7-12 В через VIN – гнучкість у виборі джерела живлення;
- малі габарити: 45×18 мм – дозволяють розмістити плату в обмеженому просторі.

На відміну від інших мікроконтролерних платформ, Arduino Nano (рис. 2.1) поєднує у собі зручність використання, простоту програмування (через середовище Arduino IDE), достатню обчислювальну потужність та широке ком'юніті підтримки. В таблиці 2.1 наведено порівняння основних характеристик Arduino Nano, Arduino Uno [22], ESP8266 (NodeMCU) [23] та Raspberry Pi Pico [24].

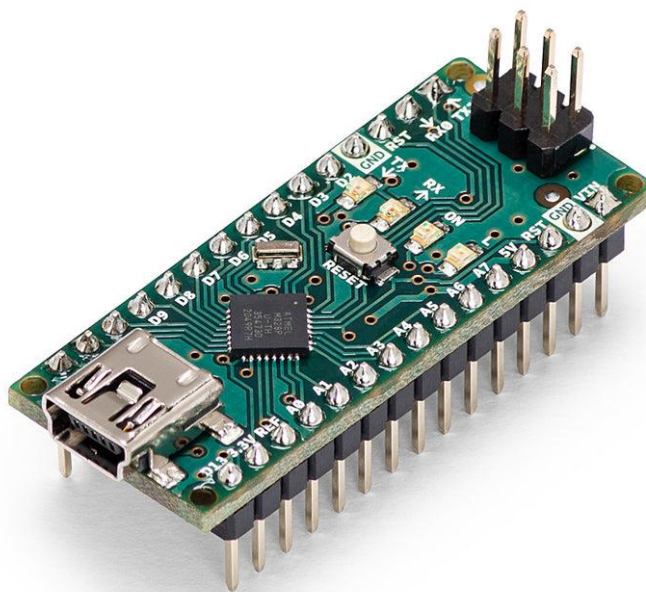


Рисунок 2.1 – Arduino Nano [21]

Таблиця 2.1 – Порівняльна характеристика популярних мікроконтролерів [20, 21-24]

Параметр	Arduino Nano	Arduino Uno	ESP8266 (NodeMCU)	Raspberry Pi Pico
Мікроконтролер	ATmega328P	ATmega328P	ESP8266	RP2040
Тактова частота	16 МГц	16 МГц	80-160 МГц	до 133 МГц
Flash-пам'ять	32 КБ	32 КБ	4 МБ	2 МБ
SRAM	2 КБ	2 КБ	80 КБ	264 КБ
Кількість цифрових входів	14	14	17	26
Аналогові входи	8	6	1	3
Інтерфейси	UART, SPI, I ² C	UART, SPI, I ² C	UART, SPI, I ² C, Wi-Fi	UART, SPI, I ² C
Підтримка Wi-Fi	Ні	Ні	Так	Ні
Напруга живлення	5 В / 7-12 В (VIN)	5 В / 7-12 В (VIN)	3.3 В	3.3 В
Розміри	45 × 18 мм	68.6 × 53.4 мм	48 × 25 мм	51 × 21 мм
Орієнтовна вартість	\$3-5	\$5-8	\$4-7	\$4-6
Сфера застосування	Компактні системи, DIY	Прототипування	Інтернет речей (IoT)	Освітні та DIY проекти

Загалом, вибір Arduino Nano обґрунтований балансом між продуктивністю, вартістю та доступністю. Для задачі реалізації автозапуску двигуна ця плата є оптимальним варіантом, що дозволяє створити компактну, функціональну та надійну систему з можливістю подальшого масштабування.

2.2 Реле для керування ланцюгами запалювання і стартера

Ефективне функціонування системи дистанційного запуску двигуна безпосередньо залежить від правильного підбору виконавчих елементів, які здійснюють фізичну реалізацію керуючих сигналів, сформованих мікроконтролером. У проєкті використовується Arduino Nano, що забезпечує вихідний струм не більше 40 мА на кожному цифровому піні, чого недостатньо для прямого керування силовими елементами автомобільної електросистеми. У зв'язку з цим доцільним є використання електромагнітних реле з керуванням від 5 В, здатних комутувати струми до 10 А і більше.

Для реалізації керування ланцюгами запалювання та стартера у даній системі використано два окремих модулі 1-канального електромагнітного реле на 5 В, які забезпечують гальванічну розв'язку завдяки наявності вбудованої оптопари. Кожен релейний модуль приймає цифровий сигнал з виходу Arduino Nano i, у разі активації, замикає чи розмикає відповідні силові контакти [25].

При виборі реле було враховано параметри, критичні для автомобільного середовища: номінальну комутаційну напругу, максимальний струм навантаження, наявність захисту (діод, оптопара), компактність модуля та легкість інтеграції з Arduino. Узагальнені характеристики найпоширеніших реле, придатних для подібних застосувань, подано у таблиці 2.2. На рисунку 2.2 представлено зовнішній вигляд одного з таких модулів.

Таблиця 2.2 – Приклади 5-вольтових електромагнітних реле [25]

Назва / Модель	Основні характеристики	Особливості застосування
Songle SRD-05VDC-SL-C	Напруга керування – 5 В DC, комутація – до 10 А / 250 V AC або 10 А / 30 V DC	Дуже популярне реле, часто встановлене на модулях з оптопарою
HK4100F-DC5V-SHG	5 В, 10 А / 125 V AC або 28 V DC, компактне реле з низьким енергоспоживанням	Зручно для обмеженого простору, добре підходить для DIY
JQC-3FF-S-Z	5 В, 7-10 А, типове міні-реле	Використовується у модульних платах, має хорошу сумісність із Arduino
Omron G5LE-1-DC5	Високонадійне реле, 5 В керування, 10 А / 250 V AC	Від бренду з високою якістю, надійне для тривалого використання
Finder 40.52.9.005.0000	Промислове реле, 5 В, комутація до 8 А	Часто використовується в промислових панелях і проектах
HW-316 Relay Module	Напруга – 5 В, вхід: з TTL-рівнем, LED-індикатор, оптопара	Має захисні елементи, зручне підключення до Arduino. Рекомендовано для автосистем середнього рівня складності.
SSR 5V-DC (твердотіле)	Напруга керування – 3-32 В DC, комутація: до 2 А при 240 В AC, без механіки, безшумне	Застосовується в системах, де важлива швидкість, довговічність, безшумність. Не підходить для високих струмів стартера.



Рисунок 2.2 – Модуль реле 1-канальний 5 В для Arduino [26]

Основні технічні характеристики реле:

- тип керування – цифровий;
- тип реле SRD-05VDC-SL-C;
- кількість каналів керування – 1;
- живлення 5 В;
- робочий струм реле: 15-20 mA
- максимальний комутований струм – макс. 10 А;
- максимальна комутована напруга, що комутується – 250 В;
- розміри – 43 x 17 мм [26].

Стандартне рішення для низьковольтних Arduino-проектів. Підходить для керування запалюванням і стартером авто.

2.3 Радіочастотний модуль для бездротового керування

Для дистанційної передачі команд потрібно обрати радіочастотний приймач, що працює на частоті 433 МГц, який є стандартним для недорогих побутових бездротових систем. Приймач інтегрується з пультом дистанційного

керування та забезпечує передачу логічних сигналів на цифрові входи мікроконтролера. В таблиці 2.3 наведені приклади RF-модулів.

Таблиця 2.3 – Приклади RF-модулів для дистанційного керування [27]

Модель	Частота	Кількість каналів	Тип передавання	Живлення	Особливості
XY-FST + XY-MK-5V	433 МГц	4	Односторонній	5 В	Простий приймач/передавач, зручний у використанні з Arduino
FS1000A + MX-05V	433 МГц	1	Односторонній	3.3-5 В	Найдешевший варіант, мінімум компонентів
WL102-341 + WL102-342	433 МГц	4	Односторонній	5 В	Часто продається з пультами в комплекті
XD-RF-5V	433 МГц	до 4	Односторонній	5 В	Сумісний із пультами на PT2262
RF-315 / RF-433 комплект	315 або 433 МГц	1-4	Односторонній	5 В	Простий монтаж, сумісність з Arduino IDE

На рисунку 2.3 зображено RF-модуль XY-MK-5V для системи автостарту двигуна.



Рисунок 2.3 – RF-модуль XY-MK-5V [28]

Основні параметри RF-модуля:

- частота прийому – 433 МГц;
- кількість каналів – 4 (дозволяє реалізувати кілька функцій керування);
- робоча напруга – 5 В;
- тип сигналу – дискретний (високий/низький рівень логіки);

– радіус дії – до 100 м за умови прямої видимості [28].

Завдяки простоті інтерфейсу RF-модуль безпосередньо підключається до цифрових входів Arduino Nano, що значно спрощує програмну реалізацію функціоналу автозапуску.

2.4 Вибір елементів для імітації роботи системи автозапуску

Було прийняте рішення застосувати сервоприводи (рис. 2.4), як виконавчі елементи, завдяки своїй здатності точно відтворювати задані кути повороту. У контексті реалізації макетної системи автозапуску двигуна на основі Arduino Nano доцільним є використання мікросервоприводів типу SG90, які поєднують у собі компактність, низьке енергоспоживання та простоту підключення. Ці пристрої працюють при напрузі 5 В і можуть отримувати живлення безпосередньо від лінії живлення контролера, без потреби у додаткових перетворювачах чи драйверах [29]. Завдяки інтерфейсу керування на основі широтно-імпульсної модуляції (PWM) SG90 легко інтегрується у цифрові системи, де керування здійснюється через бібліотеки Arduino, зокрема Servo.h, що дозволяє задавати точний кут обертання в межах від 0 до 180 градусів.

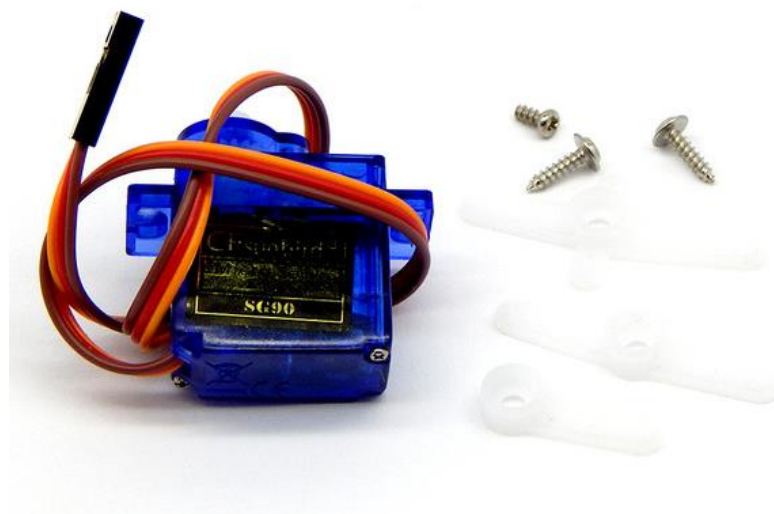


Рисунок 2.4 – Сервопривод Arduino SG90 [29]

У даному проєкті передбачено застосування сервоприводів для наочної імітації дій, які відповідають включенню запалювання та запуску стартера. Це забезпечує більш реалістичну демонстрацію принципу роботи системи автозапуску порівняно зі звичайними світлодіодними індикаторами, оскільки дозволяє фізично відтворити процес обертання ключа запалювання. З технічної точки зору, сервопривід SG90 демонструє стабільні показники навіть при живленні від портативного USB-джерела, а також характеризується помірним струмом споживання, що робить його сумісним із іншими елементами системи. Низький рівень споживання у статичному режимі та помірне навантаження в динаміці дозволяють експлуатувати SG90 без загрози перевантаження виходів Arduino Nano.

Таким чином, обґрунтоване використання SG90 у проєкті полягає не лише у його доступності й популярності серед розробників, а й у здатності ефективно виконувати функції виконавчого елемента у контексті моделювання автозапуску. Враховуючи простоту реалізації, надійність та високу повторюваність позиціонування, застосування саме цього типу сервоприводу повністю відповідає технічним завданням проєкту.

Вибір виконавчих елементів у даній роботі здійснено з урахуванням вимог до надійності, доступності та сумісності з обраною мікроконтролерною платформою. Реле забезпечують безпечне керування силовими колами, радіочастотний модуль дозволяє реалізувати бездротове керування запуском, а сервоприводи – забезпечують інтуїтивно зрозумілу індикацію стану системи. Такий підхід дозволяє реалізувати повнофункціональний прототип системи автозапуску.

2.5 Забезпечення електробезпеки та живлення

Одним із важливих аспектів при проектуванні даної системи на основі мікроконтролера є забезпечення електробезпеки, стабільного живлення та захисту компонентів від пошкоджень, спричинених перевантаженням або

короткими замиканнями. Це особливо актуально, бо система взаємодіє з електромеханічними елементами, такими як реле, сервоприводи та використовує зовнішнє живлення.

У розробленій системі автозапуску використовується плата Arduino Nano, що функціонує при напрузі 5 В, яку можна подавати як через порт USB, так і через зовнішнє стабілізоване джерело живлення. При живленні від USB доцільно використовувати джерело з обмеженням по струму (наприклад, повербанк), що допомагає уникнути перегріву або пошкодження елементів при короткому замиканні.

Сервоприводи типу SG90, які використовуються в цій системі для імітації роботи вузлів запалювання та стартера, також працюють від напруги 5 В, проте мають імпульсне енергоспоживання, що потребує врахування при проектуванні живлення. У зв'язку з цим рекомендовано підключати їх через окрему шину живлення з електролітичним конденсатором. Керування сервоприводами здійснюється безпосередньо з цифрових виходів Arduino, однак важливо забезпечити, щоб загальний струм через порти не перевищував рекомендованого значення (не більше 40 мА на пін і 200 мА сумарно).

Модулі реле та RF-приймач також працюють від 5 В і підключаються до тієї ж шини живлення, що й мікроконтролер. Для захисту Arduino від зворотних струмів або перенапруги, модулі реле мають вбудовану гальванічну розв'язку через оптрони, що суттєво знижує ризик пошкодження мікроконтролера.

Для надійної роботи системи необхідно забезпечити спільний «нуль» (GND) для всіх елементів. Порухення цієї умови може призвести до нестабільності сигналів або повного виходу системи з ладу. Рекомендується використання макетної плати з перевіреними контактами, а також регулярна діагностика цілісності ланцюгів живлення та заземлення.

РОЗДІЛ 3

ПРОЄКТУВАННЯ ТА ТЕСТУВАННЯ СИСТЕМИ

3.1 Загальна архітектура та принцип роботи системи

Розробка системи автостарту двигуна на базі мікроконтролера передбачає створення архітектури, що забезпечує надійну та контрольовану взаємодію між основними функціональними блоками. Основними елементами такої системи є мікроконтролер Arduino Nano, модулі реле, сервоприводи, RF-приймач XY-МК-5V і передавач 433 МГц. Усі ці компоненти об'єднані в єдину інформаційно-керуючу систему, яка має забезпечувати точну і безпечну імітацію замикання контактів запалювання та стартера двигуна, а також виконання додаткових операцій, що підвищують рівень автоматизації процесу запуску.

Принцип роботи системи базується на послідовному виконанні алгоритмічних етапів, кожен із яких реалізується через електронні сигнали, керовані мікроконтролером. На вхід системи надходять сигнали від RF-приймача, який приймає бездротові команди з пульта-передавача. Прийняті сигнали передаються на Arduino Nano, де відбувається їхня обробка згідно із запрограмованою логікою. У залежності від команди, мікроконтролер активує релейні модулі, які забезпечують комутацію електричних ланцюгів, необхідних для імітації роботи ключа запалювання та стартера.

RF-модуль XY-МК-5V, що використовується у системі, функціонує на частоті 433 МГц, забезпечуючи прийом сигналів від бездротового пульта керування. Завдяки своїй простоті та сумісності з Arduino, цей модуль дозволяє розширити функціональність системи без суттєвого ускладнення апаратної частини. З технічної точки зору, він забезпечує передачу цифрових сигналів на один із цифрових входів мікроконтролера, що дозволяє використовувати стандартні засоби Arduino для обробки вхідних подій. Такий підхід забезпечує масштабованість архітектури, відкриваючи можливості для розширення

функціоналу системи шляхом додавання нових типів команд або інтеграції додаткових пристроїв.

Важливим компонентом архітектури є релейні модулі, які виконують роль електричних комутаторів. Вони дозволяють мікроконтролеру, що працює на низьковольтних логічних рівнях, керувати високострумowymi ланцюгами, які зазвичай використовуються для запуску автомобільного двигуна. Конструктивно реле містить електромагнітний елемент, що приводиться в дію подачею сигналу з Arduino, замикаючи або розмикаючи контакти силового ланцюга. Завдяки цьому реалізується імітація замикання ключа запалювання та стартера без необхідності фізичної присутності користувача.

Сервоприводи Arduino SG90 виконують у системі допоміжні функції, пов'язані з механічними діями, наприклад, керуванням засувками або іншими дрібними механізмами, що можуть бути задіяні під час автозапуску. Вони підключаються безпосередньо до мікроконтролера та керуються за допомогою широтно-імпульсної модуляції (PWM), яка дозволяє точно задавати кут повороту серво. Використання сервоприводів у системі автостарту дозволяє підвищити ступінь автоматизації та забезпечити виконання кількох операцій одночасно, що є важливим для реальних застосувань.

Загальна логіка роботи системи передбачає, що після отримання команди з пульта керування, мікроконтролер аналізує, яка саме дія має бути виконана. Якщо отримана команда відповідає увімкненню запалювання, Arduino активує реле, що замикає відповідний контакт. У разі команди на запуск двигуна додатково активується реле стартера. Якщо ж надходить команда на зупинку двигуна, мікроконтролер розмикає обидва реле, перериваючи ланцюги. Усі дії супроводжуються контролем часу виконання та перевіркою станів, що дозволяє уникнути збоїв у роботі та підвищує загальну надійність системи.

Інтеграція всіх компонентів системи відбувається через макетну плату, яка забезпечує надійність підключень під час розробки та тестування прототипу. Система живиться від джерела постійного струму 5 В, що забезпечує необхідні умови для роботи Arduino, реле, RF-приймача та

сервоприводів. Окрему увагу приділено електричній сумісності компонентів, зокрема захисту мікроконтролера від струмів, що виникають під час перемикання реле, для чого використовуються захисні діоди або транзисторні буфери.

Таким чином, загальна архітектура системи автостарту є збалансованою комбінацією апаратних та програмних засобів, що забезпечує безпечну, надійну та масштабовану роботу. Вона враховує специфіку автомобільних систем, вимоги до надійності комутації, точність виконання механічних дій і можливості бездротового керування. Ця архітектура є основою для подальшого проектування електричних схем, розробки програмного забезпечення та проведення тестування прототипу

3.2 Схемотехнічне рішення системи

Схемотехнічне рішення системи автостарту двигуна на базі мікроконтролера Arduino Nano ґрунтується на використанні низки електронних компонентів, які забезпечують як електронну комутацію, так і механічне керування. Загальна схема включає Arduino Nano, два релеїні модулі на 5 В, два сервоприводи SG90, а також макетну плату для зручності збирання. Електричні з'єднання організовані таким чином, щоб забезпечити надійність роботи, мінімізацію перешкод і можливість оперативного тестування прототипу.

Центральним елементом системи є Arduino Nano, який виконує функції приймання, обробки сигналів та видачі керуючих команд. До нього підключено цифрові виходи, через які здійснюється керування реле, а також PWM-виходи для управління сервоприводами. Енергопостачання Arduino Nano забезпечується від джерела живлення 5 В, яке також використовується для живлення інших компонентів системи.

У схемі (рис. 3.1) застосовано два модулі реле, кожен з яких має три основні контакти: IN (сигнальний вхід), VCC (живлення модуля), GND

(загальне заземлення). Сигнальні входи IN обох реле підключено відповідно до цифрових пінів D5 та D3 на Arduino Nano. Це дозволяє мікроконтролеру вмикати або вимикати реле шляхом подачі високого або низького логічного рівня. Контакти VCC та GND реле підключено до відповідних шин живлення на макетній платі – червоної для +5 В і синьої для GND. Важливо зазначити, що кожне реле має вихідні контакти NO (нормально розімкнутий), NC (нормально замкнутий) та COM (спільний контакт), які забезпечують комутацію зовнішнього навантаження. У поточному проєкті використовується NO-контакт для замикання схеми запуску.

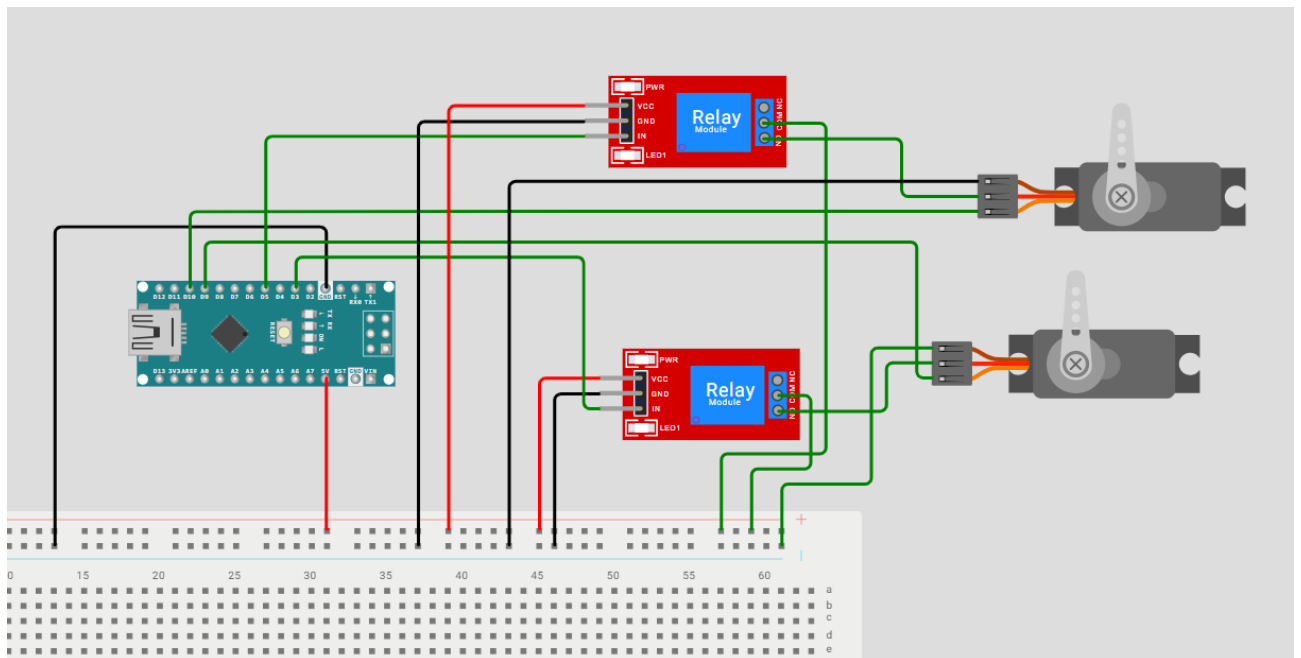


Рисунок 3.1 – Схематехнічне рішення системи автостарту двигуна

Сервоприводи SG90 підключені окремими лініями до Arduino Nano: серво №1 до PWM-піну D9, серво №2 – до PWM-піну D10. Крім того, обидва сервоприводи підключені до спільного джерела живлення 5 В та до загальної шини GND. Сигнальні дроти сервоприводів дозволяють надсилати керуючі імпульси, за допомогою яких мікроконтролер задає потрібний кут повороту вала сервопривода. Сервоприводи забезпечують виконання допоміжних

механічних функцій, наприклад, імітацію повороту ключа або перемикання важелів.

Макетна плата у схемі відіграє роль з'єднувальної основи, яка дозволяє акуратно організувати проводи та створити наочне компонування системи. Лінії живлення (+5 В та GND) проведено по всій довжині макетної плати, забезпечуючи легке підключення компонентів. На схемі видно, що Arduino Nano підключено до макетної плати через шини живлення, а від макетної плати відходять провідники до модулів реле та сервоприводів.

Усі дроти на схемі можна розділити на кілька категорій за функціональним призначенням:

- червоні – лінії живлення (+5 В);
- чорні – лінії заземлення (GND);
- зелені – сигнальні лінії для управління реле та сервоприводами.

Таке розділення дозволяє знизити ризик переплутування під час збирання та спрощує діагностику схеми. Для підключення модулів реле використано по одному проводу для сигналу, одному для живлення та одному для заземлення. Для кожного сервопривода передбачено три дроти: сигнал, живлення та земля, що є типовим варіантом підключення для компонентів цього класу.

Завдяки детальному опрацюванню схеми було забезпечено оптимальне розміщення компонентів, мінімізацію довжини проводів та зниження ймовірності виникнення електромагнітних перешкод. Особливу увагу під час розробки схемотехніки було приділено правильній організації живлення, щоб уникнути провалів напруги при одночасному увімкненні реле та сервоприводів, які можуть споживати значні струми.

Таким чином, схемотехнічне рішення складається з узгодженого набору апаратних модулів: Arduino Nano як центрального керуючого пристрою, двох релейних модулів для комутації електричних ланцюгів та двох сервоприводів для виконання механічних дій. Усі вони інтегровані на макетній платі з дотриманням основних принципів надійності, безпеки та модульності, що забезпечує ефективну роботу системи автостарту двигуна.

3.3 Розробка програмного забезпечення

Розробку програмного забезпечення для системи автостарту двигуна було здійснено у середовищі Arduino IDE, яке є офіційним інтегрованим середовищем розробки для мікроконтролерів Arduino. Arduino IDE забезпечує зручний інтерфейс для написання, компіляції та завантаження скетчів (програмного коду) безпосередньо на мікроконтролер. Версія середовища, що використовувалася під час реалізації, була Arduino IDE 2.3.2, яка підтримує сучасні функції, включаючи підсвічування синтаксису, автодоповнення, інтеграцію з бібліотеками та моніторинг послідовного порту.

Було використано стандартну бібліотеку Servo.h (рис. 3.2), що входить до комплекту поставки Arduino IDE. Ця бібліотека дозволяє керувати сервоприводами за допомогою широтно-імпульсної модуляції (PWM), забезпечуючи точний контроль кута повороту.

```
#include <Servo.h>
```

Рисунок 3.2 – Підключення бібліотеки

У проєкті було зроблено ініціалізацію двох сервоприводів (рис. 3.3).

```
Servo servo1;  
Servo servo2;
```

Рисунок 3.3 – Ініціалізація сервоприводів

Це дозволяє у подальшому здійснювати адресне управління кожним з них, задаючи потрібні значення кута повороту.

Для взаємодії з релейними модулями та RF-приймачем було визначено пін-конфігурацію мікроконтролера (рис. 3.4).

```

#define RELAY1_PIN 5
#define RELAY2_PIN 3
#define SERVO1_PIN 9
#define SERVO2_PIN 10
#define RF_INPUT_PIN 2

```

Рисунок 3.4 – Пін-конфігурація мікроконтролера

Було запрограмовано, що піни RELAY1_PIN і RELAY2_PIN працюють як цифрові виходи, а RF_INPUT_PIN – як цифровий вхід, на який подається сигнал від RF-приймача XY-МК-5V.

У функції setup() було зроблено ініціалізацію всіх компонентів (рис. 3.5).

```

void setup() {
  pinMode(RELAY1_PIN, OUTPUT);
  pinMode(RELAY2_PIN, OUTPUT);
  pinMode(RF_INPUT_PIN, INPUT);

  servo1.attach(SERVO1_PIN);
  servo2.attach(SERVO2_PIN);

  digitalWrite(RELAY1_PIN, LOW);
  digitalWrite(RELAY2_PIN, LOW);

  servo1.write(0);
  servo2.write(0);
}

```

Рисунок 3.5 – Ініціалізація всіх компонентів

Це забезпечує налаштування мікроконтролера у початковий стан, коли реле вимкнені, а серводвигуни встановлені у нульове положення. Варто зазначити, що серводвигун SG90 було з'єднано з PWM-каналами Arduino Nano для забезпечення точного регулювання кута, що є критично важливим для правильного виконання механічних дій.

Основна логіка роботи системи було запрограмовано у циклі loop(). Система постійно перевіряє стан вхідного сигналу з RF-приймача (рис. 3.6):

```
int rfSignal = digitalRead(RF_INPUT_PIN);

if (rfSignal == HIGH) {
  // Дії при отриманні сигналу
}
```

Рисунок 3.6 – Перевірка стану вхідного сигналу з RF-приймача

При активному сигналі (натисканні кнопки на пульті) було зроблено послідовну активацію обох реле, затримки для стабілізації живлення, переміщення сервоприводів на заданий кут (30°), а після цього повернення до початкового стану та вимикання реле. Для наочності фрагмент реалізації зображено на рисунку 3.7.

```
digitalWrite(RELAY1_PIN, HIGH);
digitalWrite(RELAY2_PIN, HIGH);
delay(100);

servo1.write(30);
delay(700);
servo2.write(30);
delay(500);

servo1.write(0);
servo2.write(0);
delay(500);

digitalWrite(RELAY1_PIN, LOW);
digitalWrite(RELAY2_PIN, LOW);

delay(3000);
```

Рисунок 3.7 – Послідовна активація обох реле

Було реалізовано затримки між операціями (`delay()`), що гарантують правильну послідовність комутації та рухів сервоприводів. Окремо зазначимо, що всі операції виконуються з урахуванням часу, потрібного для механічного завершення кожної дії, що було обраховано експериментально.

Таким чином, у розробленому програмному забезпеченні реалізовано повний цикл: отримання сигналу, активація електричних ланцюгів, виконання механічних дій та повернення системи до початкового стану. Програмний код було перевірено на відповідність вимогам до безпеки, надійності та стійкості роботи у повторюваних циклах. Розробка програмного забезпечення у середовищі Arduino IDE з використанням стандартних бібліотек забезпечила високу швидкість реалізації, модульність коду та можливість подальшого розширення функціоналу системи.

3.4 Процес збирання та монтажу прототипу

Процес збирання та монтажу прототипу системи автостарту двигуна є одним із ключових етапів реалізації, оскільки саме на цьому етапі перевіряється відповідність розробленого схемотехнічного рішення практичній реалізації. Монтаж було здійснено на макетній платі, яка дозволяє швидко й надійно підключати окремі компоненти без потреби у пайці, що значно прискорює етапи тестування та оптимізації.

Першочергово було розміщено мікроконтролер Arduino Nano, що виступає центральним елементом прототипу. Arduino Nano було закріплено на макетній платі так, щоб забезпечити доступ до всіх його пінів для підключення компонентів керування та живлення. Особливу увагу під час монтажу приділено правильному підключенню ліній живлення (5 В) та загальної шини заземлення (GND), оскільки некоректне підключення цих ланцюгів може призвести до нестабільної роботи або пошкодження компонентів.

Було підключено два релейні модулі, розташовані на макетній платі поруч із мікроконтролером для мінімізації довжини сигнальних дротів.

Сигнальні входи реле підключено до цифрових виходів Arduino Nano (D3 і D5), а лінії живлення VCC і GND – до відповідних шин макетної плати. Було особливо ретельно перевірено, щоб сигнальні лінії не перехрещувалися з силовими ланцюгами, оскільки це може призвести до наведення перешкод і спотворення сигналів.

Два сервоприводи SG90 було розміщено на макетній платі, на деякій відстані від мікроконтролера та релейних модулів, для уникнення механічних завад під час повороту важелів. Серводвигуни було з'єднано з PWM-виходами Arduino Nano (D9 і D10), а також підключено до загальної шини живлення та заземлення. Було перевірено відповідність кольорового маркування дротів, що дозволило уникнути помилок під час підключення.

Усі з'єднання між компонентами було виконано за допомогою м'яких монтажних дротів з наконечниками, які забезпечують надійний контакт із макетною платою. Було зроблено перевірку кожного підключення мультиметром, щоб пересвідчитися у відсутності обривів або коротких замикань. Під час збирання особливу увагу приділено організації проводів, що дозволило зменшити електромагнітні наведення та забезпечити зручність подальшого тестування.

Було проведено початковий запуск системи без підключеного зовнішнього навантаження для перевірки правильності реакції на сигнали керування. Arduino Nano було запрограмовано так, щоб після отримання сигналу від RF-приймача система активувала реле та сервоприводи у заданій послідовності. Перевірка проводилася поетапно: спочатку тестувалися реле, потім окремо – серводвигуни, і лише після цього було здійснено комплексну перевірку усієї системи.

Монтаж прототипу на макетній платі забезпечив гнучкість під час налаштування та внесення змін у схему. Це дозволило ефективно провести серію тестів, виявити та усунути незначні конструктивні недоліки, а також оптимізувати розташування компонентів для забезпечення компактності та надійності. Таким чином, процес збирання прототипу було здійснено

відповідно до сучасних принципів модульності, ремонтпридатності та технологічності, що створює основу для подальших експериментальних досліджень системи.

3.5 Тестування системи

Після завершення етапів розробки та монтажу було проведено комплексне тестування системи автостарту двигуна на базі мікроконтролера Arduino Nano. Основною метою тестування було перевірити відповідність роботи прототипу заданим технічним вимогам, оцінити стабільність виконання команд, час реакції на сигнали керування, а також виявити можливі недоліки як у схемотехнічному, так і в програмному забезпеченні.

У тестуванні використовувалася серія послідовних сценаріїв, які супроводжувалися фіксацією станів релейних модулів та сервоприводів. Було зроблено поетапні знімки системи у різних фазах роботи, щоб візуально підтвердити виконання команд. На рисунках 3.8-3.10 показані три ключові стани системи під час експериментального запуску.

На рисунку 3.8 обидва релейні модулі перебувають у вимкненому стані, індикатори LED не горять, а важелі обох сервоприводів SG90 знаходяться у нульовій позиції. Верхній сервопривід відповідає за імітацію повороту ключа запалювання, а нижній – за імітацію активації стартера. Цей стан системи підтверджує готовність до прийому команди через RF-приймач.

На цьому етапі (рис. 3.9) спостерігається, що перший релейний модуль (відповідає за замикання контактів запалювання) активований, що підтверджується світінням зеленого індикатора LED. Сервопривід 1, який імітує поворот ключа запалювання, починає змінювати своє положення, демонструючи виконання завдання. Сервопривід 2 і друге реле залишаються у початковому стані, очікуючи на подальшу команду.

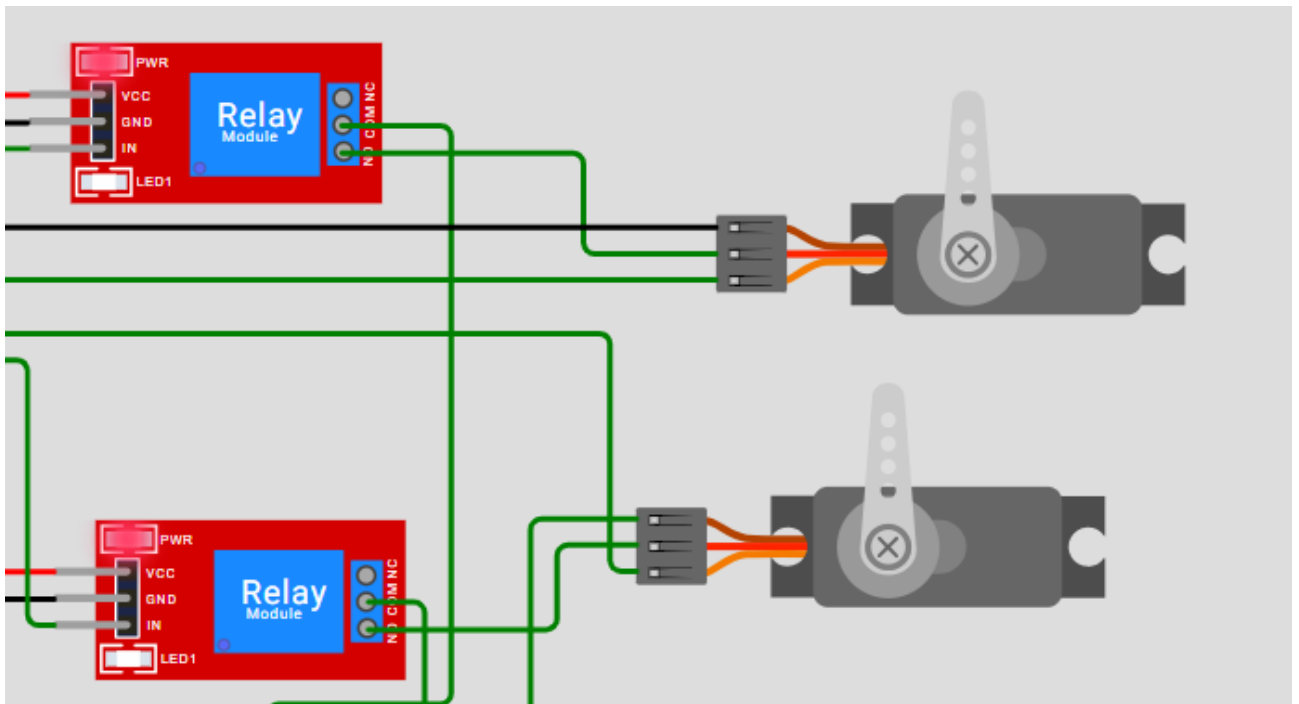


Рисунок 3.8 – Початковий стан системи перед активацією

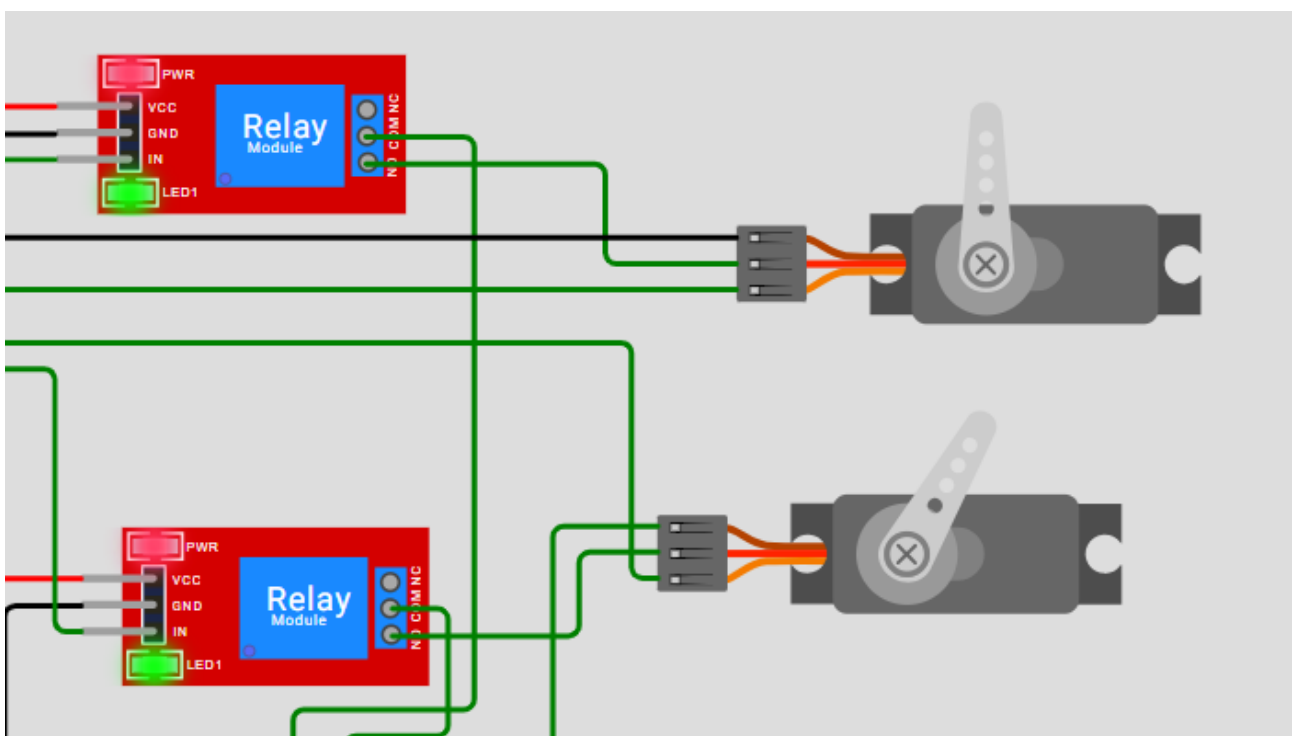


Рисунок 3.9 – Стан після активації першого реле

На рисунку (рис. 3.10) обидва релейні модулі знаходяться у ввімкненому стані: індикатори LED світяться, сигналізуючи про замикання контактів як запалювання, так і стартера. Обидва сервоприводи також виконують поворотні

рухи. Верхній сервопривід завершує імітацію повороту ключа запалювання, а нижній приводить у дію механізм, що імітує запуск стартера. Цей стан підтверджує, що система виконує повний цикл автозапуску згідно із запрограмованим алгоритмом.

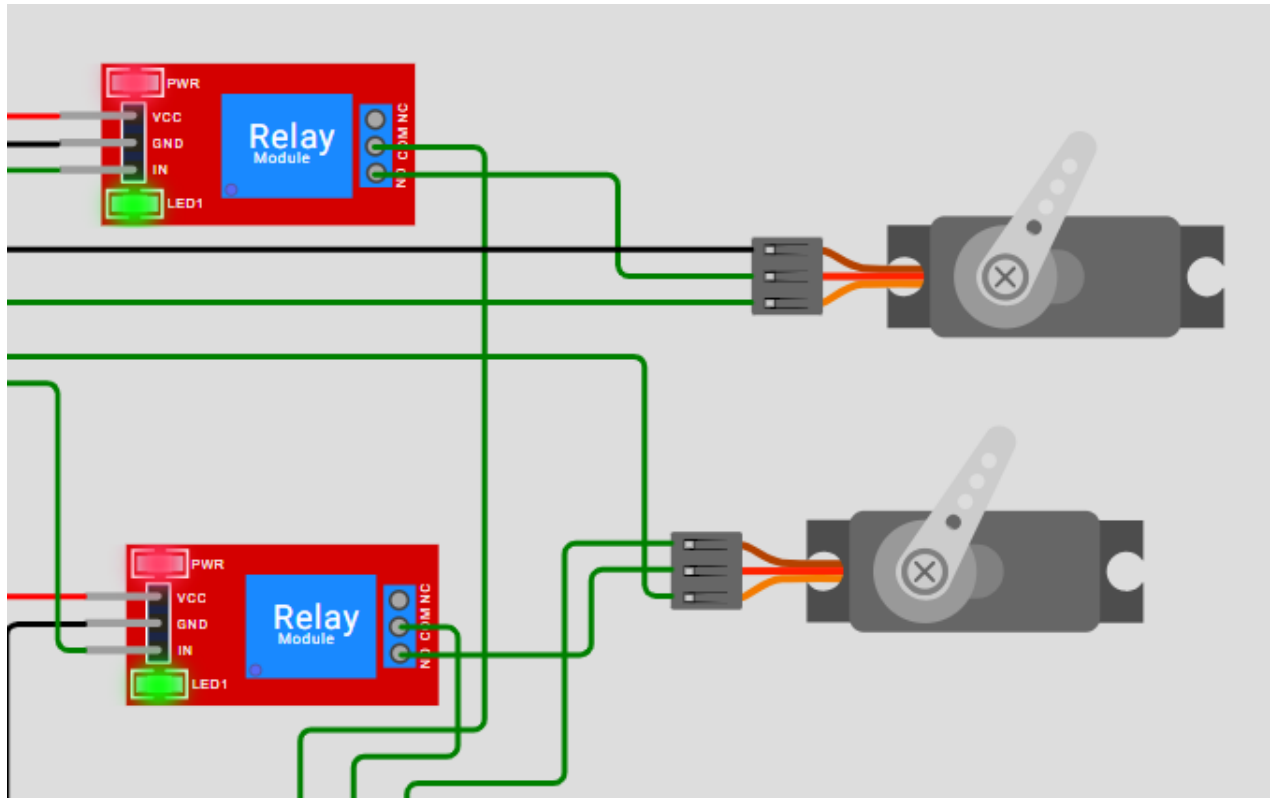


Рисунок 3.10 – Стан після повної активації системи

Завдяки проведеним тестам було підтверджено коректну взаємодію всіх компонентів системи, а також стабільність роботи при послідовному та паралельному увімкненні реле та сервоприводів. Система демонструє передбачувану поведінку: у відповідь на бездротовий сигнал активуються реле, а серводвигуни виконують задані механічні дії, точно імітуючи роботу традиційного замка запалювання та стартера. Експериментально було встановлено, що система здатна безпомилково виконувати команди протягом тривалого часу, що свідчить про її надійність та придатність до практичного використання

3.6 Аналіз та вдосконалення

Після проведення тестування системи автостарту двигуна було здійснено комплексний аналіз отриманих результатів, з метою виявлення сильних і слабких сторін реалізованого прототипу, а також визначення можливостей для його вдосконалення. Було встановлено, що система в цілому виконує поставлені задачі, демонструючи стабільну роботу як апаратної, так і програмної частин, однак експериментальна перевірка виявила низку аспектів, які потребують подальшої оптимізації.

Важливим досягненням реалізованої системи є точне виконання команд, отриманих через RF-приймач, із мінімальними затримками та високим рівнем повторюваності. Це свідчить про коректну інтеграцію бездротового модуля, мікроконтролера Arduino Nano, релейних модулів та сервоприводів. Проте під час багатогодинних тестів було зафіксовано кілька випадків нестабільної роботи при пікових навантаженнях, коли одночасно вмикалися обидва реле й серводвигуни. Аналіз показав, що основною причиною є обмеження потужності живлення макетної плати, що не забезпечує стабільної напруги під час споживання струмів, близьких до граничних значень.

З метою вдосконалення пропонується додатково використовувати окремий стабілізований блок живлення для сервоприводів, розділивши живильні ланцюги мікроконтролера та виконавчих механізмів. Це дозволить зменшити навантаження на спільну шину та запобігти просіданню напруги. Також доцільно застосувати фільтрувальні конденсатори більшої ємності або спеціальні модулі стабілізації, які згладжуватимуть короткочасні коливання під час роботи реле.

Аналіз програмного забезпечення виявив, що основна логіка побудована за послідовним принципом, із використанням фіксованих затримок `delay()`. Хоча такий підхід є достатнім для прототипу, його вдосконаленням може стати впровадження асинхронної обробки подій за допомогою таймерів або прерывань. Це дозволить скоротити час реакції системи, забезпечивши

одночасне виконання кількох завдань без очікування завершення попередніх. Зокрема, можна реалізувати паралельний контроль кількох модулів або додати обробку нових команд під час виконання поточних операцій.

З огляду на результати тестування також доцільно розглянути варіант виготовлення друкованої плати для підвищення надійності з'єднань та компактності конструкції. Макетна плата, яка використовувалася для побудови прототипу, добре підходить для експериментів, проте вона не є оптимальним варіантом для експлуатації в умовах реального автомобіля через підвищений ризик роз'єднання контактів під час вібрацій і механічних навантажень.

Таким чином, проведений аналіз підтвердив, що система автостарту двигуна на базі мікроконтролера Arduino Nano є працездатною та демонструє задовільні експлуатаційні характеристики. Водночас було визначено низку напрямів для подальшого вдосконалення, зокрема в аспектах енергозабезпечення, оптимізації програмного коду, розширення функціоналу та підвищення надійності конструкції. Реалізація цих вдосконалень дозволить наблизити систему до рівня, придатного для використання в реальних умовах експлуатації автомобільного транспорту.

ВИСНОВКИ

У ході виконання роботи було досягнуто основної мети – розробки та тестування системи автоматичного запуску двигуна автомобіля на базі мікроконтролера, що підтвердило актуальність теми, пов'язаної зі зростаючою популярністю систем дистанційного керування автомобілем.

Було проведено ґрунтовний огляд існуючих систем автостарту, що дозволило систематизувати знання про сучасні рішення, архітектури та основні функції таких систем. Завдяки цьому аналізу вдалося визначити загальні принципи побудови, порівняти переваги й недоліки наявних рішень.

Далі було сформульовано вимоги до системи, серед яких ключовими стали: надійність дистанційного запуску, стабільність роботи апаратної частини, точність виконання програмних алгоритмів і простота архітектури для забезпечення доступності й можливості модифікацій.

Під час підбору апаратних компонентів було обрано мікроконтролер Arduino Nano як центральний блок керування, релейні модулі для комутації контактів запалювання й стартера, сервоприводи SG90 для виконання механічних дій, а також RF-приймач XY-MK-5V для прийому бездротових сигналів. Вибір компонентів базувався на їхній сумісності, технічних характеристиках та доступності на ринку.

Було розроблено схему підключення, яка враховує логічні зв'язки між усіма модулями, а також забезпечує коректне підключення живлення, сигнальних ліній та захист від перешкод. Це стало основою для створення апаратної частини системи, дозволяючи побудувати компактну та надійну модель на макетній платі.

Після складання прототипу було проведено його тестування, яке дозволило підтвердити працездатність системи, визначити її технічні характеристики, час реакції та стабільність роботи. Тестування допомогло виявити потенційні проблеми, пов'язані з енергозабезпеченням та

алгоритмічними затримками, що, у свою чергу, стало підґрунтям для формування пропозицій щодо вдосконалення.

У підсумку робота засвідчила, що використання мікроконтролерів Arduino є ефективним підходом до створення недорогих, функціональних та гнучких систем автоматичного запуску автомобільних двигунів. Розроблений прототип продемонстрував хороший потенціал для подальших модифікацій, зокрема у напрямках підвищення надійності, розширення функціоналу та оптимізації апаратної й програмної частини.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. History of General Motors: An American Automotive Legend. *Internetsearchinc.* URL: <https://www.internetsearchinc.com/history-of-general-motors-an-american-automotive-legend/> (дата звернення: 13.01.2025).
2. History | Toyota Industries Corporation. *Toyota Industries Corporation.* URL: https://www.toyota-industries.com/company/history_2/index.html (дата звернення: 13.01.2025).
3. History. *Corporete Ford.* URL: <https://corporate.ford.com/about/history.html> (дата звернення: 13.01.2025).
4. Stories. *Nissan Motor Corporation.* URL: <https://www.nissan-global.com/EN/STORIES/> (дата звернення: 13.01.2025).
5. Що таке CAN-шина? *Microtronic.* URL: <https://microtronic.com.ua/novyny/sho-take-can-shina> (дата звернення: 14.01.2025).
6. Як вибрати кращу автосигналізацію: корисні рекомендації. *130.com.ua.* URL: <https://130.com.ua/uk/luchshaya-avtosyhnalyzatsyya/> (дата звернення: 14.01.2025).
7. Technology Trend in Remote Engine Starter. *Denso Ten.* URL: <https://www.denso-ten.com/business/technicaljournal/pdf/33note1.pdf> (дата звернення: 16.01.2025).
8. Controller Area Network (CAN) Protocol Overview. *Mess und Prüfsysteme, bei Emerson – NI.* URL: <https://www.ni.com/en/shop/seamlessly-connect-to-third-party-devices-and-supervisory-system/controller-area-network--can--overview.html> (дата звернення: 16.01.2025).
9. Українські GSM автосигналізації з CAN шиною Gazer. *Арнаж Інтернет-магазин автотоварів.* URL: <https://arnage.com.ua/ukrainski-gsm-avtosyhnalizatsii-z-can-shynoju-gazer/> (дата звернення: 17.01.2025).
10. Application of Radio Wave Technologies to Vehicles. *SCI-HUB.* URL: <https://sci-hub.se/downloads/2020-02-02/c2/naito2014.pdf> (дата звернення: 17.01.2025).

11. Бiньковська А. Б., Кудирко О. М., Тащиков А. В. Розширення функціональних можливостей системи автосигналізації. *International scientific and practical conference «Science, engineering and technology: global trends, problems and solutions»*: Conference proceedings, September 25-26, 2020. Prague: Izdevnieciba «Baltija Publishing», 2020. Part 2. 208. P. 8-10.

12. Module approach to wireless technology. *Vehicle Electronics*. URL: <https://vehicle-electronics.biz/content/module-approach-wireless-technology> (дата звернення: 19.01.2025).

13. Chang R.-I., Lin T.-C., Lin J.-W. A Vehicle Passive Entry Passive Start System with the Intelligent Internet of Things. *Electronics*. 2024. Vol. 13, no. 13. P. 2506. URL: <https://doi.org/10.3390/electronics13132506> (дата звернення: 19.01.2025).

14. Dzelendzyak U., Fedyniak I., Dziuba Y. SYSTEM OF REMOTE CONTROL AND MANAGEMENT OF VEHICLE FUNCTIONS. *Computer systems and network*. 2024. Vol. 6, no. 2. P. 44-50. URL: <https://doi.org/10.23939/csn2024.02.044> (дата звернення: 20.01.2025).

15. Аулін, В. В., Магопець М. С. Формування мікроклімату в салонах автотранспортних засобів підігрівачами різної конструкції. *Центральноукраїнський науковий вісник*. Технічні науки: зб. наук. пр. Кропивницький: ЦНТУ, 2024. Вип. 9(40). Ч. 2. С. 146-157. URL: <https://dspace.kntu.kr.ua/handle/123456789/16187> (дата звернення: 20.01.2025).

16. Pros and Cons of Using a Remote Starter. *Mobile Install - Car Audio & Remote Start Installation / Minneapolis*. URL: <https://www.mobile-install.com/blog/entryid/257/pros-and-cons-of-using-a-remote-starter>. (дата звернення: 21.01.2025)

17. A Secure Communication Protocol for Remote Keyless Entry System with Adaptive Adjustment of Transmission Parameters. *arXiv.org*. URL: <https://arxiv.org/abs/2504.09527?> (дата звернення: 21.01.2025).

18. This Gadget Hacks GM Cars to Locate, Unlock, and Start Them (UPDATED). *WIRED*. URL: <https://www.wired.com/2015/07/gadget-hacks-gm-cars-locate-unlock-start/>? (дата звернення: 22.01.2025).

19. Microcontroller Selection for Automotive Embedded Systems. *Diva-portal*. URL: <https://www.diva-portal.org/smash/get/diva2:1712830/FULLTEXT01.pdf>. (дата звернення: 12.02.2025).

20. Arduino Nano V3.0 AVR ATmega328P з розпаяними роз'ємами купити в Києві та Україні. *Arduino в Україні*. URL: <https://arduino.ua/prod166-arduino-nano-v3-0-avr-atmega328p-s-raspayannimi-razemami> (дата звернення: 13.02.2025).

21. Arduino Nano. *Store.arduino*. URL: <https://store.arduino.cc/products/arduino-nano>. (дата звернення: 13.02.2025).

22. Arduino UNO – популярна плата розробки. *Itmaster.biz*. URL: <https://itmaster.biz.ua/directory/kits-nabory/arduino-uno.html> (дата звернення: 14.02.2025).

23. NodeMCU v3 на базі ESP-12E. *Mini-Tech*. URL: <https://www.mini-tech.com.ua/ua/nodemcu-esp12-ch340> (дата звернення: 14.02.2025).

24. Raspberry Pi Pico. *Uamper.com*. URL: <https://uamper.com/Raspberry-Pi-Pico-%D0%BE%D0%B4%D0%BD%D0%BE%D0%BF%D0%BB%D0%B0%D1%82%D0%BD%D1%8B%D0%B9-%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80-%D0%BC%D0%B8%D0%BD%D0%B8-%D0%BF%D0%BA> (дата звернення: 15.02.2025).

25. Реле. *Arduino Придбати в Києві, Україна*. URL: <https://arduino.ua/ru/cat67-rele> (дата звернення: 15.02.2025).

26. Модуль реле 1-канальний 5V для Arduino PIC ARM ARM AVR. *Інтернет магазин FlyMod. Продаж FPV дронів і комплектуючих, доставка по Україні*. URL: https://flymod.net/item/1way_relay_module_5v (дата звернення: 16.02.2025).

27. Радіомодулі. *Arduino Придбати в Києві, Україна*. URL: <https://arduino.ua/cat98-radiomodyli> (дата звернення: 16.02.2025).

28. XY-MK-5V – модуль радіоприймача на 433/315 МГц – IT Master – електроніка та програмування. *Головна – IT Master – електроніка та програмування*. URL: <https://itmaster.biz.ua/directory/radio/xy-mk-5v-module.html> (дата звернення: 17.02.2025).

29. Сервопривод Arduino SG90 (Micro Servo motor). *Набори та компоненти для самостійного збирання електроніки*. URL: <https://arduino.com.ua/ua/p1111355106-servoprivod-arduino-sg90.html> (дата звернення: 18.02.2025).