Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інфокомунікацій
(повна назва)

Кафедра інформаційно-мережної інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіті перший (бакалаврський)

Розробка електронного цифрового замка на базі Arduino
(тема)

Виконав:
здобувач 4 року навчання,
групи ТРІМІ-21-2
Юрій Мац
(власне ім'я, прізвище)

Спеціальність 172 Телекомунікації та радіотехніка
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Інформаційно-мережна інженерія
(повна назва освітньої програми)

Керівник: ст. викл. Олександр Бибка
(посада власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри _____ Валерій Безрук
(підпис) (власне ім'я, прізвище)

2025 р.

Не містить відомостей заборонених до відкритого публікування

Здобувач _____ / Юрій Мац /
(підпис) (прізвище та ініціали)

Керівник _____ / Олександр Бибка /
(підпис) (прізвище та ініціали)

Харківський національний університет радіоелектроніки

Факультет інфокомунікацій

Кафедра інформаційно-мережної інженерії

Рівень вищої освіти перший (бакалаврський)

Спеціальність 172 Телекомунікації та радіотехніка
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма інформаційно-мережна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«____»_____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Мацу Юрію Євгеновичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка електронного цифрового замка на базі Arduino

затверджена наказом університету від 23 травня 2025 р. № 410 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 24 червня 2025 р.

3. Вихідні дані до роботи _____
Провести аналіз особливостей цифрових замків, їхні переваги та недоліки
Розробити прототип електронного цифрового замка на базі плати Arduino.

4. Перелік питань, що потрібно опрацювати в роботі _____
Вступ
1. Аналіз цифрових замків
2. Розробка прототипу
Висновки

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

Слайди у форматі Power Point (назва та мета роботи, компоненти прототипу, принцип роботи, вдосконалення прототипу, висновки)

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Ознайомлення із завданням. Уточнення ТЗ.	23.05.2025	вик.
2	Підбір літератури за темою роботи	24.05.-28.05.2025	вик.
3	Аналіз цифрових замків	29.05.-02.06.2025	вик.
4	Розробка прототипу	03.06.2024-19.06.2025	вик.
5	Оформлення презентаційного матеріалу,	20.06.-21.06.2025	вик.
	підготовка до захисту у ЕК		

Дата видачі завдання 23 травня 2025 р.

Здобувач _____ Юрій Мац
(підпис)

Керівник роботи _____ ст. викл. Олександр Бибка
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 49 с., 22 рис., 2 табл., 5 джерел, 2 додатки.

Мета роботи – розробка прототипу електронного замка, який відкривається введенням правильного коду через клавіатуру, з використанням плати Arduino.

Об'єкт дослідження – система керування доступом на базі мікроконтролера Arduino Uno з електронним замком.

У даній роботі розглянуто процес розробки електронного цифрового замка на базі Arduino. Такий замок забезпечує доступ до об’єкта (кімнати, сейфа, пристрою тощо) лише після введення правильного пароля. Проєкт включає використання клавіатури для введення коду, рідкокристалічного дисплея для відображення інформації, а також сервомеханізму, що імітує механізм замикання. Особливу увагу приділено логіці програмного забезпечення, енергоефективності, зручності використання та можливості подальшого розширення функціоналу.

Актуальність теми полягає у зростаючій потребі в доступних і персоналізованих системах безпеки. Цифрові замки на базі Arduino мають значний потенціал для інтеграції у системи «розумного дому», навчальні проєкти, а також для побутового застосування.

СИМУЛЯТОР, ПЛАТА, ARDUINO, LCD ДИСПЛЕЙ, СЕРВОДВИГУН, КЛАВІАТУРА, ЗАМОК

THE ABSTRACT

Explanatory note: 49 p., 22 fig, 2 tabl., 5 sources, 2 app.

The purpose of the work is to develop a prototype of an electronic lock that opens by entering the correct code through the keyboard, using the Arduino board.

The object of research is an access control system based on the Arduino Uno microcontroller with an electronic lock.

This paper examines the process of developing an Arduino-based electronic digital lock. Such a lock provides access to an object (room, safe, device, etc.) only after entering the correct password. The project includes the use of a keyboard for entering a code, an LCD for displaying information, and a servomechanism that simulates the locking mechanism. Particular attention is paid to software logic, energy efficiency, ease of use, and the possibility of further expanding the functionality.

The relevance of the topic lies in the growing need for affordable and personalized security systems. Arduino-based digital locks have significant potential for integration into smart home systems, educational projects, and household applications.

SIMULATOR, BOARD, ARDUINO, LCD DISPLAY, SERVO MOTOR, KEYBOARD, LOCK

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

AES – Advanced Encryption Standard – розширений стандарт шифрування;

API – Application Programming Interface – інтерфейс програмування застосунків;

IoT – Internet of Things – інтернет речей;

LCD – Liquid Crystal Display – рідкокристалічний дисплей;

NFC – Near Field Communication – технологія бездротового зв'язку на короткій відстані;

PWM – Pulse-Width Modulation – широтно-імпульсна модуляція;

RAM – Random Access Memory – оперативна пам’ять;

RFID – Radio-Frequency Identification – радіочастотна ідентифікація;

SCL – Serial Clock Line – серійна лінія годинника;

SDA – Serial Data Line – серійна лінія даних;

SPI – Serial Peripheral Interface – синхронний протокол передачі даних;

СКД – системи контролю доступу;

ШІМ – широтно-імпульсна модуляція.

ВСТУП

У сучасному світі безпека є однією з найважливіших складових як у побуті, так і в професійному середовищі. Стрімкий розвиток електроніки, мікроконтролерів та автоматизованих систем дав змогу створювати ефективні, недорогі та гнучкі засоби контролю доступу. Одним із таких рішень є електронний цифровий замок, який можна реалізувати за допомогою мікроконтролера Arduino.

Arduino – це платформа для створення прототипів, що поєднує простоту апаратного забезпечення з гнучкістю програмування. Вона активно використовується в освітніх, дослідницьких та прикладних проєктах. Завдяки широкій спільноті, доступності датчиків та модулів, Arduino дозволяє швидко створювати пристрої з різними функціональними можливостями.

У даній роботі розглянуто процес розробки електронного цифрового замка на базі Arduino. Такий замок забезпечує доступ до об’єкта (кімнати, сейфа, пристрою тощо) лише після введення правильного пароля. Проєкт включає використання клавіатури для введення коду, рідкокристалічного дисплея для відображення інформації, а також сервомеханізму, що імітує механізм замикання. Особливу увагу приділено логіці програмного забезпечення, енергоефективності, зручності використання та можливості подальшого розширення функціоналу.

Актуальність теми полягає у зростаючій потребі в доступних і персоналізованих системах безпеки. Цифрові замки на базі Arduino мають значний потенціал для інтеграції у системи «розумного дому», навчальні проєкти, а також для побутового застосування.

1 АНАЛІЗ ЦИФРОВИХ ЗАМКІВ

1.1 Основні відомості

Цифрові замки працюють завдяки поєднанню електронних і механічних елементів, які дозволяють контролювати доступ до приміщень або об’єктів. Основна ідея їхньої роботи полягає в ідентифікації користувача за допомогою електронних методів та подальшому увімкненні механізму замикання залежно від результату перевірки [1].

На першому етапі користувач проходить процедуру аутентифікації, використовуючи певний інтерфейс, наприклад: клавіатуру, сенсорний екран, біометричний сканер (відбитки пальців, розпізнавання обличчя), або бездротові засоби – такі як смарт-карти чи мобільні додатки. Ці дані надсилаються на контролер замка, який порівнює їх із заздалегідь збереженими у пам’яті пристрою або в хмарному сервісі [1].

Якщо верифікація проходить успішно, система приймає рішення про надання доступу, і користувач отримує дозвіл на відкриття. У разі невідповідності введених даних – доступ блокується, а система може повідомити про невдалу спробу.

Після підтвердження особи, контролер активує виконавчий механізм (електромотор або соленоїд), що відмикає двері. Замок може автоматично зачинятися після певного часу або за бажанням користувача вручну через той самий інтерфейс.

Сучасні цифрові замки часто мають візуальні або звукові індикатори, що сигналізують про стан замка (успішне або невдале відкриття). Часто вони також підтримують підключення до Інтернету речей (IoT), що дає можливість керування замком дистанційно за допомогою смартфона чи іншого пристрою. Вся інформація про спроби входу може зберігатися у вбудованій пам’яті або в

хмарному сховищі, що дозволяє власнику переглядати історію доступу та отримувати сповіщення про підозрілі дії [1].

У підсумку, цифрові замки поєднують зручність та надійність, забезпечуючи ефективний контроль доступу та знижуючи ризик несанкціонованого проникнення.

Цифрові замки є невід'ємною складовою сучасних систем безпеки, забезпечуючи зручний і надійний контроль доступу до приміщень, об'єктів чи пристроїв. Їхня широка функціональність та технологічна гнучкість дозволяють обирати найбільш ефективні рішення залежно від конкретних потреб користувача або умов експлуатації.

Класифікація цифрових замків може здійснюватися за кількома основними критеріями: за методом аутентифікації, типом живлення, способом керування та сферою застосування. Кожна з цих категорій включає в себе різноманітні типи замків, які мають свої унікальні особливості, переваги та недоліки. Такий підхід дозволяє систематизувати існуючі рішення і полегшує вибір найбільш відповідного варіанту для конкретного завдання [1].

1.1.1 Класифікація за методом аутентифікації

Метод аутентифікації визначає, яким чином користувач підтверджує свій доступ до системи. Основні типи включають [1]:

– кодові замки (PIN-код) – найпоширеніший варіант, що вимагає введення певної комбінації цифр на клавіатурі;

– біометричні замки – використовують відбитки пальців, розпізнавання обличчя або сітківки ока;

– RFID/NFC замки – дозволяють ідентифікувати користувача за допомогою безконтактних карток або міток;

– Bluetooth/Wi-Fi замки – керуються зі смартфона або іншого пристрою через мобільні додатки;

– гібридні системи – поєднують кілька способів аутентифікації для підвищення безпеки.

1.1.2 Класифікація за типом живлення

Живлення цифрових замків може бути [1]:

– батарейне живлення – автономні пристрої, що працюють на акумуляторах або батарейках.

– зовнішнє живлення (через адаптер) – постійне підключення до електромережі.

– гібридне живлення – поєднує батарейне й зовнішнє живлення для підвищеної надійності.

1.1.3 Класифікація за способом керування

Управління цифровими замками здійснюється різними способами [1]:

– локальне керування – через клавіатуру, дисплей або сенсори без підключення до мережі;

– дистанційне керування – через мобільні додатки, Bluetooth або Wi-Fi;

– інтегроване керування – як частина системи «розумного будинку» або охоронної системи.

1.1.4 Класифікація за сферою застосування

Цифрові замки використовуються в різних галузях [1]:

– побутові замки – для квартир, будинків, гаражів.

– промислові та офісні системи – для контролю доступу на підприємствах.

– спеціалізовані рішення – для сейфів, серверних кімнат, електрощитів тощо.

1.2 Аналіз цифрових замків від різних виробників

Сучасний ринок цифрових замків представлений великою кількістю рішень від різних виробників, які варіюються за функціональністю, рівнем безпеки, технологічною складністю, надійністю, ціною та дизайном. Така

різноманітність дозволяє споживачам обирати замки відповідно до своїх потреб – від простих бюджетних моделей до багатофункціональних пристроїв, що підтримують біометричну ідентифікацію та інтеграцію в системи «розумного будинку» [2].

Компанії-виробники цифрових замків [2]:

– Samsung Smart Locks (Samsung SDS, Південна Корея) – виробляє широкий асортимент дверних цифрових замків із PIN-кодами, RFID-картами, біометрією (відбитки пальців), Bluetooth;

– Yale (Assa Abloy Group, Швеція) – один із найстаріших брендів замків у світі, сьогодні активно впроваджує розумні замки з підтримкою Wi-Fi, Bluetooth, додатків та інтеграцією з Google Home, Amazon Alexa;

– August Home (США) – відомі своїми замками, що легко встановлюються на традиційні двері. Підтримують управління зі смартфона, голосове керування та інтеграцію з IoT-системами;

– Schlage (Allegion, США) – пропонує електронні та смарт-замки з клавіатурами, біометрією, підтримкою ZigBee, Z-Wave, Wi-Fi;

– Kwikset (Spectrum Brands, США) – виробляє надійні цифрові замки для дому, серед яких популярна лінійка Kevo з Bluetooth-управлінням;

– Lockly (США) – компанія спеціалізується на біометричних смарт-замках із сенсорними дисплеями, унікальною технологією випадкового розміщення цифр, віддаленим доступом;

– ZKTeco (Китай) – відомий виробник систем контролю доступу з широким асортиментом біометричних цифрових замків;

– Igloohome (Сінгапур) – пропонує замки з PIN-кодами, які працюють без постійного інтернет-з'єднання, створюючи динамічні ключі, які зручно надсилати віддалено;

– Philips Smart Lock (Нідерланди) – виробляє цифрові замки з підтримкою відбитків пальців, паролів і мобільного керування;

– Dessmann (Німеччина/Китай) – спеціалізується на цифрових дверних замках із відбитками пальців, RFID та Wi-Fi/BT-контролем.

Ці компанії активно розвивають технології безпеки, роблячи цифрові замки все більш функціональними, зручними та інтегрованими у сучасні системи «розумного дому». Якщо потрібно, можу надати порівняльну таблицю або приклади конкретних моделей [2].

1.3 Інтегрування цифрових замків в системи безпеки

Інтеграція цифрових замків з іншими системами безпеки є важливим напрямом розвитку сучасних охоронних технологій. Завдяки можливості взаємодії з іншими елементами системи «розумного будинку» або корпоративної системи безпеки, цифрові замки значно підвищують ефективність та функціональність охоронних рішень [3].

Можливості інтеграції цифрових замків з іншими системами безпеки [3]:

– відеоспостереження – цифрові замки можуть бути інтегровані з IP-камерами. При спробі входу (успішній чи неуспішній) камера автоматично активується та фіксує зображення. Відео може зберігатися локально або в хмарі;

– сигналізація – у разі несанкціонованого доступу або багаторазового введення неправильного коду цифровий замок може активувати охоронну сигналізацію. Це дозволяє миттєво реагувати на загрозу;

– системи контролю доступу (СКД) – цифрові замки можуть працювати як частина ширшої системи контролю доступу в офісних будівлях, готелях, навчальних закладах. Вони можуть бути синхронізовані з базами даних, журналами входів/виходів та корпоративними обліковими записами;

– інтелектуальні датчики – цифрові замки можуть взаємодіяти з датчиками руху, відкривання дверей, температури та диму. Наприклад, при пожежі або витоку газу система автоматично розблокує замки для швидкої евакуації;

– голосові помічники (IoT) – багато сучасних замків сумісні з Alexa, Google Assistant або Siri. Користувачі можуть відкривати замок голосовою командою або через мобільний застосунок;

– мобільні застосунки та хмарні сервіси – мобільний додаток дозволяє керувати замками дистанційно, встановлювати тимчасові коди, відстежувати історію доступу. Хмарна інтеграція дозволяє зберігати інформацію та налаштування незалежно від пристрою користувача;

– домашні автоматизовані сценарії – наприклад, відкриття дверей може автоматично вмикати світло, запускати кондиціонер або вітальну музику. Це підвищує рівень комфорту та безпеки одночасно.

Переваги інтеграції [3]:

– централізоване керування безпекою з одного інтерфейсу;

– підвищення рівня захисту завдяки взаємодії з іншими пристроями;

– гнучкість та масштабованість системи під потреби користувача;

– віддалений доступ та моніторинг.

Можливі обмеження [3]:

– необхідність наявності стабільного інтернет-з'єднання;

– складність налаштування для недосвідчених користувачів;

– підвищені вимоги до кібербезпеки.

Інтеграція цифрових замків із системами безпеки дозволяє створити комплексне рішення, що забезпечує не лише надійний фізичний захист, а й розумну взаємодію між усіма компонентами системи безпеки. Це ідеальний варіант для сучасного житла, офісу чи підприємства, де безпека та зручність мають ключове значення [3].

Але існують як переваги, так і певні обмеження у використанні плати Arduino для цифрових замків, і в залежності від сфери використання необхідно про них пам’ятати та враховувати при реалізації цифрового замка на базі плати Arduino.

1.4 Переваги у використанні плати Arduino для цифрових замків

До переваг використання можна віднести [4]:

– простота у використанні – Arduino має зрозуміле середовище розробки (IDE) та велику кількість прикладів, що дозволяє навіть початківцям створювати функціональні проєкти без глибоких знань електроніки;

– гнучкість і модульність – легке підключення додаткових модулів (Bluetooth, RFID, Wi-Fi, дисплеї, датчики тощо) дозволяє створювати системи з різним рівнем складності та функціоналу;

– низька вартість – плата Arduino Uno та сумісні компоненти є відносно дешевими, що робить проєкт доступним для широкого кола користувачів;

– велика спільнота та документація – Arduino підтримується світовою спільнотою розробників: існує безліч готових бібліотек, інструкцій, схем, прикладів коду;

– швидке прототипування – Arduino дозволяє швидко тестувати та вдосконалювати систему завдяки простоті змін як у програмному коді, так і в апаратному забезпеченні.

1.5 Обмеження у використанні плати Arduino для цифрових замків

До обмежень використання можна віднести [4]:

– низький рівень захисту – Arduino сам по собі не забезпечує шифрування чи захист від зловмисного доступу. Без додаткових заходів безпеки цифровий замок може бути вразливим;

– обмежені обчислювальні ресурси – мікроконтролери Arduino Uno мають обмежену оперативну пам’ять і тактову частоту, що ускладнює реалізацію складних алгоритмів безпеки чи обробки даних;

– низька енергоефективність – Arduino не оптимізований для роботи на батарейному живленні тривалий час без спеціальних енергозберігаючих рішень;

– відсутність захисту на рівні обладнання – більшість плат Arduino не мають апаратного захисту від короткого замикання, перенапруги або механічного злому;

– підходить здебільшого для навчальних або побутових проєктів – хоча Arduino і дозволяє створювати робочі пристрої, для промислових або сертифікованих систем безпеки потрібні більш надійні рішення з сертифікованим програмним забезпеченням.

2 РОЗРОБКА ПРОТОТИПУ

2.1 Мета та компоненти проєкту

Розробка прототипу електронного замка, який відкривається введенням правильного коду через клавіатуру, з використанням плати Arduino.

Розроблений прототип складається з (рис. 2.1): плати Arduino Uno, клавіатури 4x4, LCD дисплею (1602) та серводвигуна.

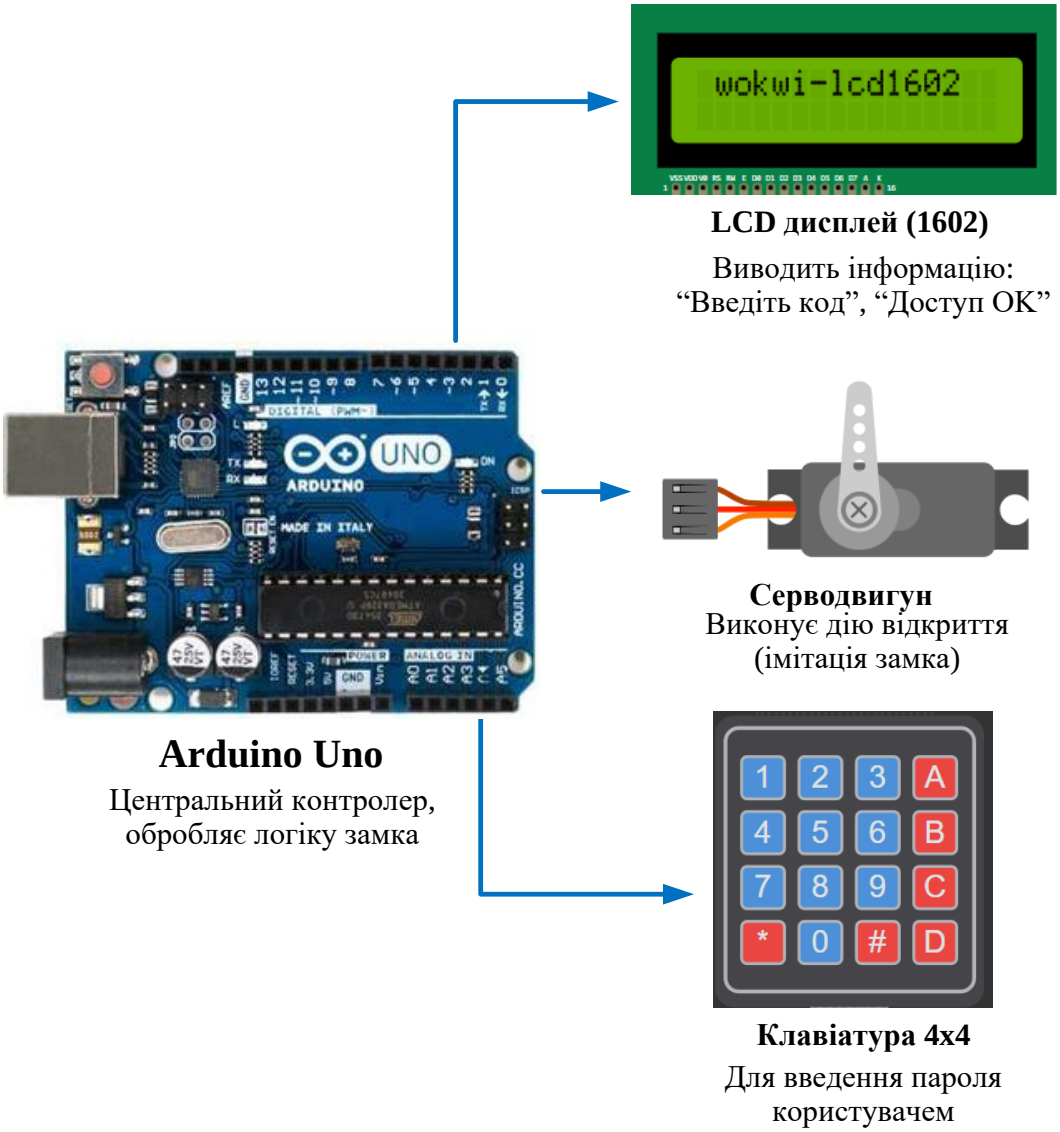

Рисунок 2.1 – Компоненти прототипу

У процесі розробки цифрового замка на базі Arduino Uno до плати були підключені основні елементи: клавіатура 4×4, серводвигун та LCD дисплей 1602, кожен з яких використовує окремі цифрові або спеціалізовані інтерфейси.

Клавіатура 4×4 – під’єднана до аналогових пінів A0-A3 та цифрових пінів D6-D9, які використовуються як звичайні цифрові входи/виходи. Це можливе завдяки тому, що на Arduino Uno аналогові піни A0-A5 також можуть працювати як цифрові (з номерами D14-D19):

– рядки: D6, D7, D8, D9;

– стовпці: A0, A1, A2, A3 (як цифрові піни 14-17).

Серводвигун підключено до PWM-виводу D10, що дозволяє керувати його положенням за допомогою широтно-імпульсної модуляції:

– керуючий пін: D10 (PWM);

– живлення: +5V;

– GND: загальний мінус.

LCD дисплей 1602 підключено до цифрових пінів D2–D5 та D11–D12. Для передачі даних використовується 4-бітний режим, що дозволяє заощадити пінів на Arduino:

– RS: D12;

– EN: D11;

– D4–D7: D5, D4, D3, D2;

– підсвітка: +5V;

– RW: GND (режим запису).

Усі з'єднання здійснено за допомогою стандартних jumper-проводів та макетної плати (breadboard).

Для економії виводів і зручного керування:

– аналогові входи використовуються як цифрові пін для клавіатури;

– PWM-пін задіяно для точного керування серводвигуном;

– LCD дисплей використовує мінімум 6 цифрових пінів в 4-бітному режимі передачі даних.

Система цифрового замка побудована на основі плати Arduino Uno, яка використовує мікроконтролер ATmega328P. Це одна з найпопулярніших платформ для розробки електронних пристроїв у навчальних та хобі-проєктах.

Arduino Uno має такі основні характеристики [5]:

– 14 цифрових входів/виходів, з яких 6 можуть працювати в режимі ШІМ (PWM);

– 6 аналогових входів для зчитування аналогових сигналів (наприклад, з датчиків);

– кварцовий резонатор на 16 МГц;

– USB-порт для програмування та живлення;

– роз’єм живлення (барельний конектор 2.1 мм);

– роз’єм ICSP для прошивки мікроконтролера;

– кнопка скидання (RESET).

Arduino Uno може отримувати живлення двома основними способами:

– через USB-порт – приєднання до комп’ютера або зарядного пристрою;

– через зовнішнє джерело живлення, наприклад: адаптер змінного струму (AC/DC); акумуляторна батарея (наприклад, Li-ion або крона 9 В).

Підключення зовнішнього джерела здійснюється через:

– барельний роз’єм 2.1 мм (центральний контакт – «+»);

– або безпосередньо до пінів Vin (вхід напруги) та GND (земля).

Arduino автоматично перемикається між джерелами живлення залежно від наявності підключень (табл. 2.1).

Таблиця 2.1 – Рекомендації щодо живлення

Напруга живлення	Опис
6 В	Мінімально можлива, але нестабільна
7–12 В	Рекомендований діапазон для стабільної роботи
12–20 В	Можлива, але регулятор може перегріватися
> 20 В	Заборонено – можливе пошкодження плати

Напруга живлення менше 7 В може призвести до нестабільної роботи, оскільки на виході 5V буде недостатня напруга. Понад 12 В може викликати перегрів вбудованого стабілізатора, що небезпечно для плати.

Плата Arduino Uno, як і більшість мікроконтролерів, потребує стабільного джерела живлення для забезпечення належної роботи всіх внутрішніх компонентів та підключених пристроїв. Існує кілька способів подачі живлення, які відрізняються залежно від наявного обладнання та умов експлуатації [5].

Основні способи живлення Arduino Uno. Живлення через USB [5]:

– Arduino може бути підключено безпосередньо до комп’ютера або зарядного пристрою через USB-кабель типу B. Цей спосіб зручний для:

– програмування мікроконтролера;

– тестування пристроїв у процесі розробки;

– живлення пристрою у випадках, коли потужності USB вистачає для всієї системи.

Зовнішнє живлення через роз’єм (DC Jack) – подача напруги через стандартний барельний роз’єм 2,1 мм, який підтримує зовнішні блоки живлення (адаптери) з напругою від 7 до 12 В (оптимально) [5]:

– рекомендований діапазон: 7–12 В;

– мінімальна допустима: 6 В (можливі збої);

– максимальна безпечна: 12 В;

– критична межа: не більше 20 В (вище – ризик пошкодження).

Вивід VIN – на платі передбачений спеціальний пін VIN для підключення нестабілізованого зовнішнього джерела живлення. Саме через цей вивід можна подавати напругу у випадку, якщо не використовується USB або DC-роз’єм [5]:

– напруга: 7–12 В (як для адаптера);

– підключення: + до VIN, – до GND.

Увага: VIN напряму з'єднаний з вхідним стабілізатором напруги, тому подання занадто високої напруги може призвести до перегріву або пошкодження плати [5].

Живлення від батарей – Arduino Uno також може працювати від акумуляторних батарей або блоків живлення з елементами живлення (наприклад, 9 В крона, 2S/3S Li-ion, 6×AA тощо). У цьому випадку джерело підключається або через [5]:

– DC-роз’єм;

– VIN + GND.

Arduino Uno має вбудовану логіку автоматичного вибору джерела живлення. Якщо USB підключено, пріоритет надається йому; якщо ні — живлення надходить через VIN або DC-роз’єм. Це дозволяє гнучко перемикатися між джерелами без ручного втручання [5].

Плата Arduino Uno має декілька спеціалізованих виводів, призначених для подачі живлення, регуляції напруги та програмування мікроконтролера.

Виводи живлення:

– 5V – це регульоване джерело напруги 5 В, яке живить мікроконтролер ATmega328P та інші компоненти на платі. Напруга на цьому виводі може подаватися трьома способами: з USB-порту; через вивід VIN (після стабілізації внутрішнім регулятором); від зовнішнього джерела 5 В (якщо підключено напряму до цього піну) [5];

– 3V3 – це вивід 3.3 В, що регулюється вбудованим лінійним стабілізатором на платі. Застосовується для підключення зовнішніх компонентів, які працюють на 3.3 В. Максимальний допустимий струм: 50 мА. Перевищення струму може призвести до нестабільної роботи або виходу стабілізатора з ладу [5];

– GND (Ground) – загальний провід, або «земля». Усі джерела живлення, модулі та компоненти мають бути з’єднані з цим виводом для забезпечення замкненого електричного кола.

Програмування через порт ICSP (In-Circuit Serial Programming):

– це спеціальний роз’єм, який використовується для низькорівневого програмування мікроконтролера, особливо у випадках, коли: стандартний USB-порт недоступний або не функціонує; необхідно прошити завантажувач

(bootloader); потрібно змінити ф'юзи (fuses) — спеціальні біти налаштувань контролера;

– через ICSP можна: перепрошити мікроконтролер; підключитися до внутрішньої SPI-шини (Serial Peripheral Interface); отримати прямий доступ до внутрішньої пам'яті.

CSP складається з 6 пінів: MISO, MOSI, SCK, RESET, VCC, GND.

Мікроконтролер ATmega328P, що використовується на платі Arduino Uno, має набір виводів, які умовно поділяються на цифрові та аналогові (рис. 2.2). Ці виводи слугують для зчитування сигналів, керування пристроями та зв'язку з іншими модулями.

Рисунок 2.2 – Цифрові та аналогові виводи

У середовищі розробки Arduino (IDE) всі пін-виводи мікроконтролера мають уніфіковану нумерацію, яка представлена в табл. 2.2.

Таблиця 2.2 – Рекомендації щодо живлення

Назва	Логічний номер в коді
D0–D13	0–13
A0	14
A1	15
A2	16
A3	17
A4	18
A5	19

Таким чином, виводи A0–A5 мають подвійну адресуцію: як аналогові (A0–A5) та як цифрові (14–19).

Таким чином, Arduino Uno забезпечує широкі можливості для роботи з вхідними та вихідними сигналами. Аналогові входи дозволяють вимірювати рівень напруги, що відкриває можливість створення навіть простого осцилографа, хоч частота вимірювань і обмежується швидкістю мікроконтролера. Цифрові пін-виводи можуть зчитувати логічні стани або задавати їх, у тому числі генерувати сигнали ШІМ, які зазвичай використовують для керування двигунами або відтворення звуків. Крім того, за допомогою інтерфейсів UART, I2C, SPI та односпрямованих шин, Arduino може взаємодіяти з різноманітними зовнішніми пристроями [5].

Інтерфейси I2C та SPI дозволяють підключати декілька пристроїв до однієї шини, що значно зменшує кількість необхідних пінів мікроконтролера:

– I2C використовується в багатьох датчиках і модулях, оскільки потребує лише два сигнальні пін-виводи – SDA і SCL;

– SPI забезпечує вищу швидкість передачі даних та зазвичай застосовується для пристроїв, які потребують інтенсивного обміну інформацією, таких як Ethernet-модулі, Wi-Fi-модулі або SD-карти.

Більшість сенсорів для Arduino підключаються через [5]:

– аналогові входи (наприклад, потенціометри, фоторезистори);

– односпрямовані цифрові шини (як-от OneWire);

– або інтерфейс I2C.

Для зручності підключення зовнішніх компонентів до Arduino Uno часто використовують:

– Sensor Shield – плату-розширення, яка дублює всі виводи Arduino, додаючи контакти живлення (5V, GND) біля кожного сигналу. Це значно спрощує підключення сенсорів, сервоприводів та інших модулів;

– макетна плата (breadboard) – незамінна при експериментальній розробці, оскільки дозволяє швидко підключати компоненти без пайки, спрощуючи тестування схем.

Плата Arduino Uno оснащена різноманітними засобами зв’язку, що забезпечують взаємодію з комп’ютером, іншими мікроконтролерами або пристроями Arduino. Основний мікроконтролер ATmega328P підтримує послідовний UART TTL інтерфейс (5 В), який реалізується через виводи 0 (RX) та 1 (TX) [5].

Для зручного підключення до ПК цей інтерфейс переадресовується через USB за допомогою мікросхеми ATmega8U2, яка працює як USB-to-Serial перетворювач. Завдяки цьому комп’ютер розпізнає плату як віртуальний COM-порт, що дозволяє легко обмінюватися даними через серійний монітор Arduino IDE або стороннє програмне забезпечення [5].

На платі Arduino Uno передбачена можливість відключення автоматичного перезавантаження (авторесету), яке зазвичай виконується при кожному підключенні до комп’ютера або відкритті серійного порту. Це може бути корисно, наприклад, при роботі з деякими програмами, що постійно тримають порт відкритим [5].

Автоматичне перезавантаження реалізується через окрему лінію «RESET-EN», яка з’єднує мікросхему USB-UART з виводом RESET мікроконтролера.

Щоб відключити авто-ресет, достатньо розімкнути цю лінію, видаливши перемичку або розрізавши доріжку на платі (залежно від серії плати).

Для тимчасового відключення можна підключити резистор номіналом ~110 Ом між виводом RESET та 5V, що блокує короткочасний спад напруги, який викликає перезавантаження.

У разі потреби лінію можна знову замкнути, з'єднавши контакти обох її кінців, відновивши нормальну функцію авто-ресету.

У конструкції кодового замка дисплей (рис. 2.3) використовується для виведення повідомлень, що відображають стан системи (наприклад: "Введіть код", "Доступ дозволено", "Невірний код"), та забезпечення інтерактивної взаємодії з користувачем.

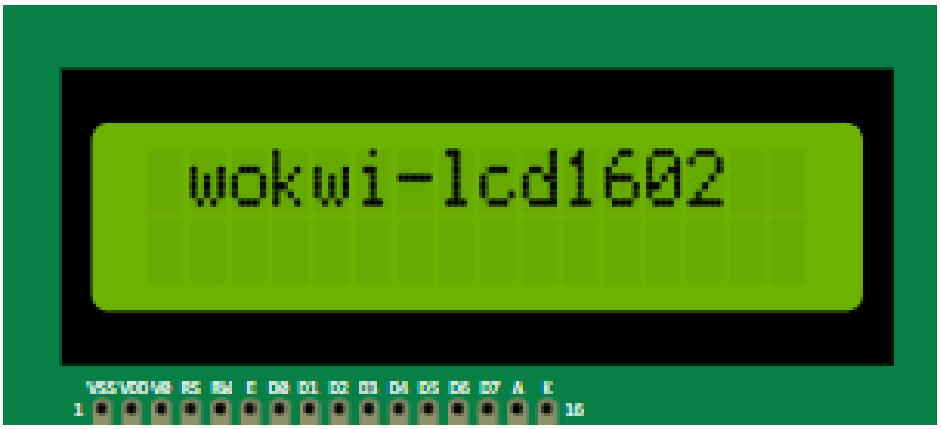

Рисунок 2.3 – LCD дисплей (1602)

Підключення дисплея до плати Arduino. Для передачі даних використовується інтерфейс I2C, який значно спрощує підключення, оскільки замість кількох цифрових пінів використовує лише два сигнальні проводи:

– SDA (Serial Data) – підключається до виводу A4 на платі Arduino Uno;

– SCL (Serial Clock) – підключається до виводу A5.

Крім того, дисплей потребує живлення, яке забезпечується через виводи:

– VCC – підключається до 5V;

– GND – до GND на платі Arduino.

Завдяки використанню I2C-модуля, дисплей не займає багато цифрових виводів, що дозволяє легко інтегрувати його в багатокомпонентні проєкти, як-от кодовий замок.

Клавіатура використовується для введення пароля в системі кодового замка. На рис. 2.4 представлена матрична клавіатура з 16 клавішами (4x4), яка сумісна з платформами Arduino, AVR, PIC, ARM та іншими мікроконтролерами.

Основна перевага матричної клавіатури полягає в можливості підключити велику кількість кнопок, використовуючи обмежену кількість пінів мікроконтролера. Зокрема, для підключення 16 кнопок використовується лише 8 пінів – 4 рядки та 4 стовпці, замість 16 окремих входів.

Клавіатура 4x4 широко застосовується для:

– введення кодів у кодових замках;

– керування пристроями;

– введення числових або символічних даних у різноманітних проєктах.

Рисунок 2.4 – Клавіатура 4x4

Для використання клавіатури потрібно:

– підключити її до Arduino через цифрові пін-виводи;

– ініціалізувати програмну бібліотеку, яка дозволяє зчитувати натискання клавіш (наприклад, Keypad.h);

– забезпечити живлення системи.

Принцип дії: при натисканні клавіші замикається певна пара провідників – один з рядка та один зі стовпця. Програма опитує рядки та перевіряє, у яких точках відбулося замикання, і таким чином визначає, яка саме клавіша була натиснута.

Серводвигун (рис. 2.5) є важливою складовою цифрового замка на базі Arduino, оскільки він виконує фізичне відкривання або закривання механізму замка після введення правильного коду. Це електромеханічний пристрій, призначений для точного керування кутом обертання.

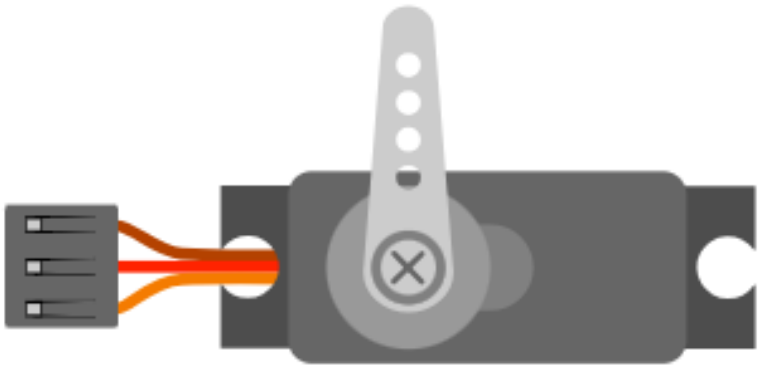

Рисунок 2.5 – Серводвигун

Серводигун поєднує в собі кілька компонентів:

– двигун постійного струму (DC-мотор);

– редуктор, що знижує швидкість обертання і підвищує точність;

– потенціометр, який забезпечує зворотний зв'язок про положення вала;

– вбудований контролер, який аналізує сигнал керування і регулює положення вала відповідно до заданого значення.

Arduino керує серводвигуном за допомогою ШІМ-сигналу (PWM), передаючи команду на певний кут повороту. Завдяки цьому забезпечується висока точність і повторюваність рухів, що особливо важливо для електронних замків.

2.2 Принцип роботи

Мова програмування Arduino базується на мовах C і C++, з інтеграцією бібліотеки AVR Libc, що дозволяє використовувати її стандартні функції при розробці прошивок для мікроконтролера. Незважаючи на технічну основу, середовище розробки Arduino було створене таким чином, щоб бути простим та інтуїтивно зрозумілим, навіть для новачків.

На сьогодні Arduino є одним із найдоступніших та найзручніших інструментів для розробки програмного забезпечення для мікроконтролерів, особливо в навчальних проєктах та прототипуванні.

Мову Arduino умовно можна розділити на чотири основні компоненти:

– оператори – керування потоком виконання (if, for, while, switch, тощо).

– типи даних – змінні, масиви, структури, тощо.

– функції – вбудовані та користувацькі (наприклад: digitalWrite(), analogRead()).

– бібліотеки – модулі, які додають специфічні можливості (наприклад: LiquidCrystal, Servo, Keypad тощо).

Таке розділення дозволяє швидко орієнтуватися у синтаксисі мови Arduino та ефективно застосовувати її для програмування пристроїв.

Написання коду починається з додавання необхідних бібліотек (рис. 2.6) для кожного з компонент прототипу: клавіатури, дисплею та серводвигуна, а також необхідні бібліотеки, які відповідають з запис паролю до пам’яті та для іконки на самому дисплеї.

```
#include <SPI.h>
#include <MFRC522.h>
#include <LiquidCrystal.h>
#include <Keypad.h>
#include <Servo.h>
#include "SafeState.h"
#include "icons.h"
```

Рисунок 2.6 – Бібліотеки програмного коду

Серводвигун відповідає за керування механізму замка. Під’єднуємо його до виходу – 6. На рис. 2.7 представлений код для ініціалізації серводвигуна, де другий рядок відповідає за поворот в закритому положенні серводвигуна, а третій – за поворот в відкритому положенні.

```
/* Locking mechanism definitions */
#define SERVO_PIN 6
#define SERVO_LOCK_POS   20
#define SERVO_UNLOCK_POS 90
Servo lockServo;
```

Рисунок 2.7 – Ініціалізація серводвигуна

Під’єднавши дисплей до плати, в коді створюємо об’єкт для нього (рис. 2.8) – для симуляції вводу/виводу повідомлень.

```
/* Display */
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);
```

Рисунок 2.8 – Ініціалізація дисплею

Останнім елементом залишається клавіатура. Для її працездатності в коді необхідно створити масив правильного значення кожної клавіші (рис.2.9).

```
const byte KEYPAD_ROWS = 4;
const byte KEYPAD_COLS = 4;
byte rowPins[KEYPAD_ROWS] = {A3, A2, A1, A0};
byte colPins[KEYPAD_COLS] = {A4, A5, A6, A7};
char keys[KEYPAD_ROWS][KEYPAD_COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};

Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, KEYPAD_ROWS, KEYPAD_COLS);
```

Рисунок 2.9 – Масив

Також необхідно створити об’єкт, який зберігатиме пароль користувача. Саме SafeState збереже секретний код користувача в EEPROM (рис. 2.10).

```
/* SafeState stores the secret code in EEPROM */
SafeState safeState;
```

Рисунок 2.10 – Збереження паролю

На рис. 2.11 представлено код початку роботи прототипу електронного цифрового замка.

Запустивши симуляцію проєкту в онлайн симуляторі, на дисплеї з’явиться напис «Welcom!» (рис. 2.12), що підтверджує правильний початок роботи прототипу.

Далі система перебуває в режимі очікування (рис. 2.13), після натискання символу «#»пропонує користувачу ввести чотирьох значний код (рис. 2.14).

```
void lock() {
  lockServo.write(SERVO_LOCK_POS);
  safeState.lock();
}

void unlock() {
  lockServo.write(SERVO_UNLOCK_POS);
}

void showStartupMessage() {
  lcd.setCursor(4, 0);
  lcd.print("Welcome!");
  delay(1000);

  lcd.setCursor(0, 2);
  String message = "ArduinoSafe v1.0";
  for (byte i = 0; i < message.length(); i++) {
    lcd.print(message[i]);
    delay(100);
  }
  delay(500);
}
```

Рисунок 2.11 – Код початку роботи

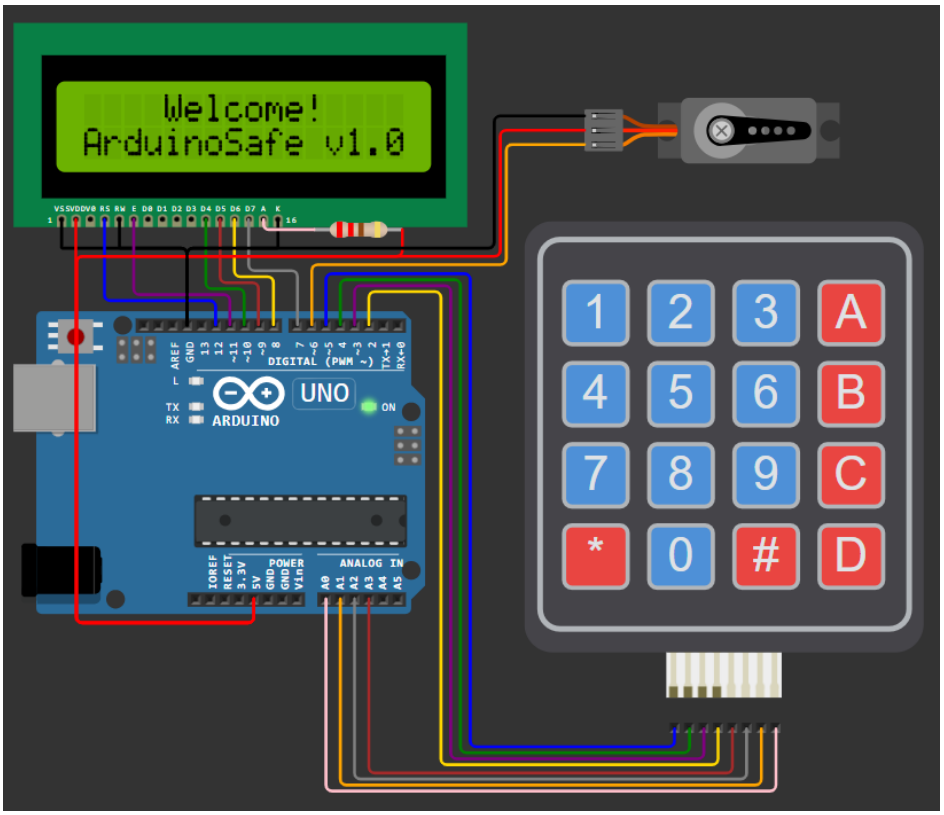

Рисунок 2.12 – Запуск прототипу

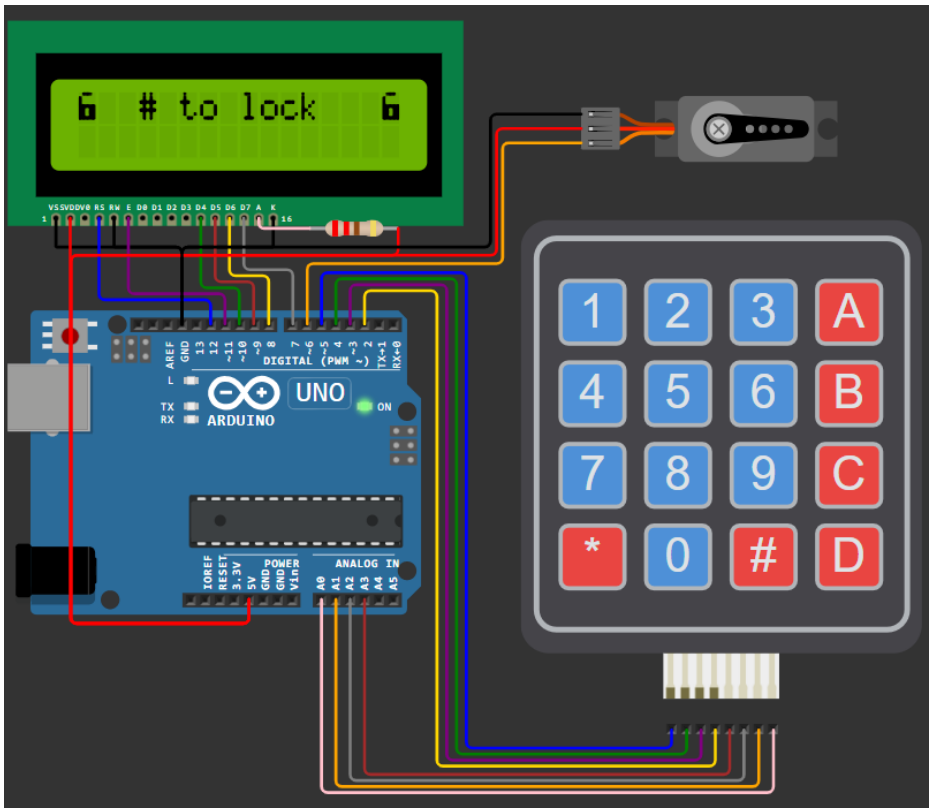

Рисунок 2.13 – Режим очікування

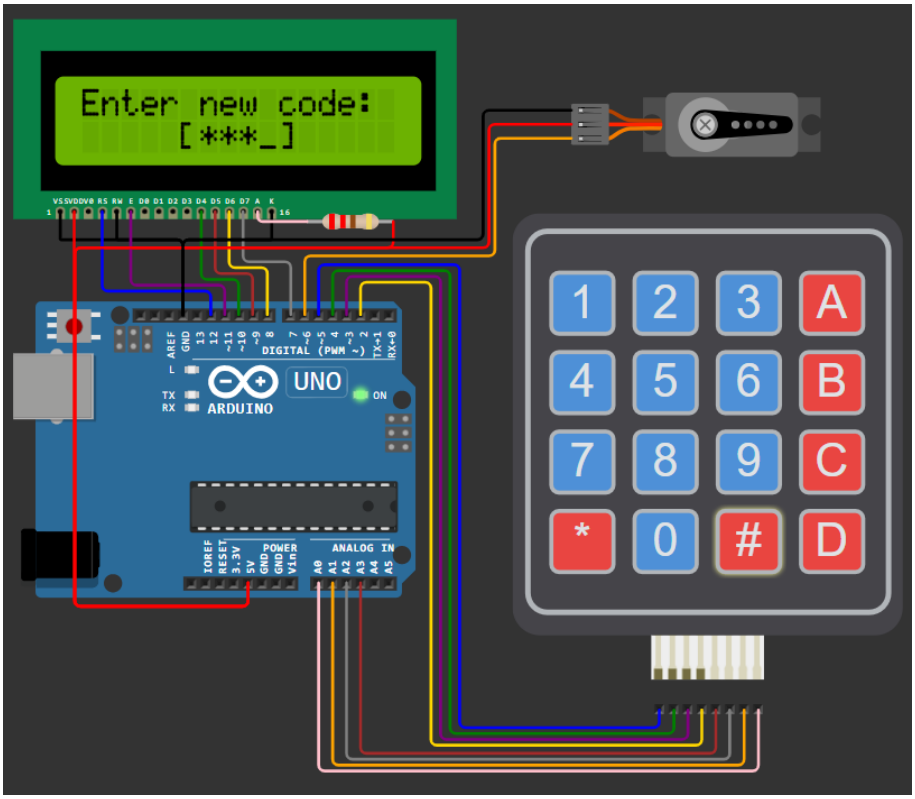

Рисунок 2.14 – Введення коду

Користувач вводить код через клавіатуру. Далі необхідно повторити введення коду (рис. 2.15).

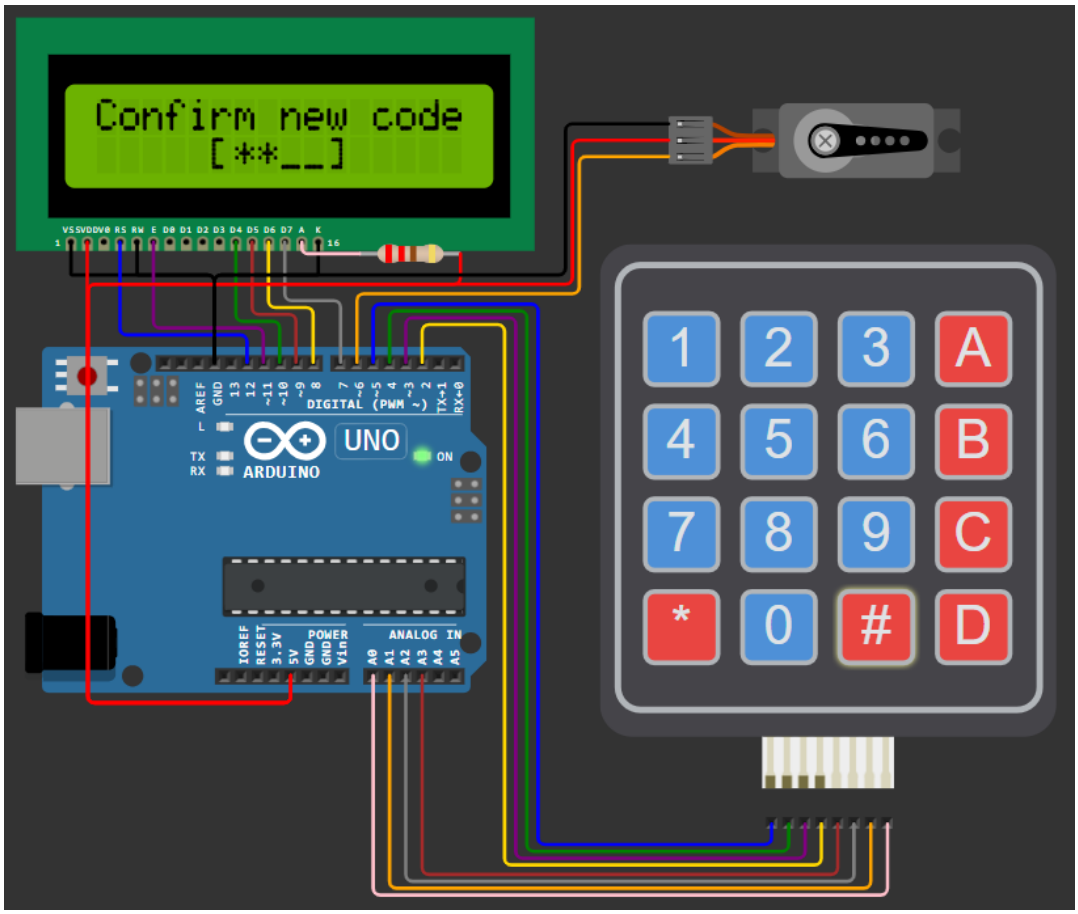

Рисунок 2.15 – Повторне введення коду

Якщо повторно введений правильний код – подається сигнал на серводвигун, що закриває «замок» (рис. 2.16).

Якщо повторно введений код неправильний – з’являється повідомлення про помилку, можна ввести ще раз (рис. 2.17).

Після закриття можемо відкрити «замок». Якщо введено правильний код – подається сигнал на серводвигун, що відкриває «замок», але спершу буде повідомлення системи про це (рис. 2.18). Після успішного відкриття, система дає можливість для нового введення паролю (рис. 2.19).

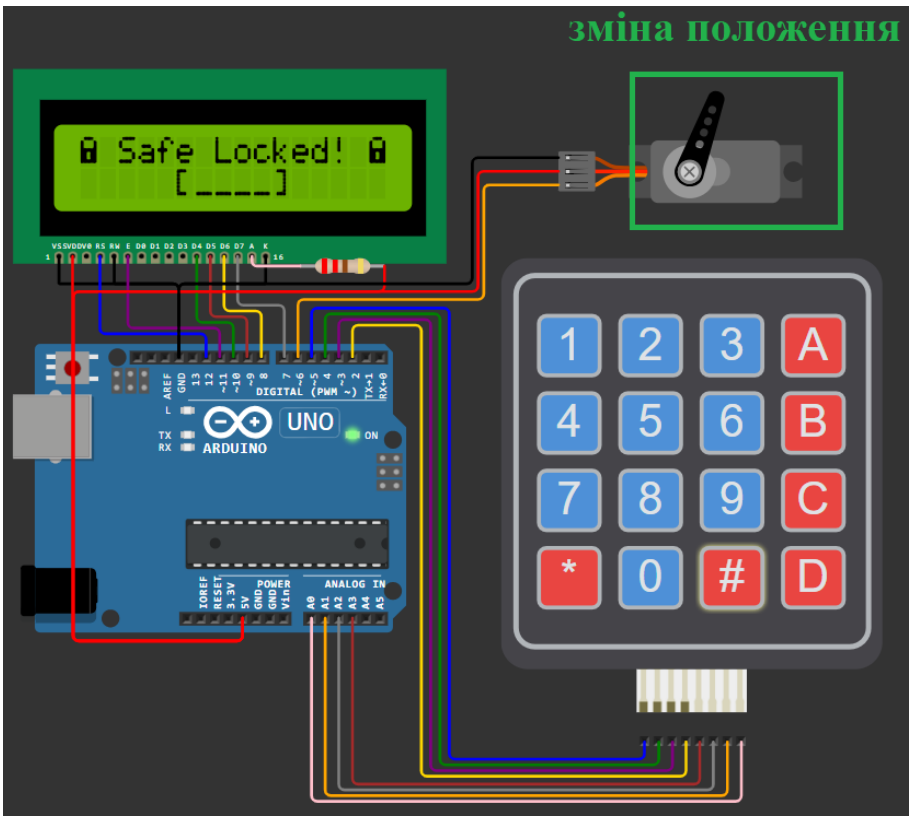

Рисунок 2.16 – Закриття «замку»

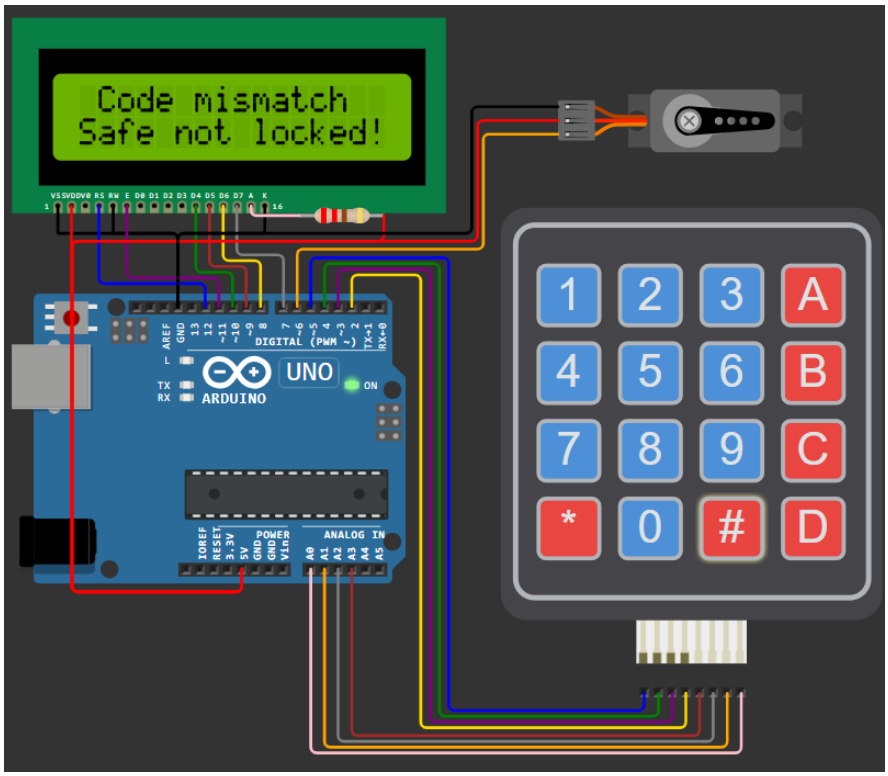

Рисунок 2.17 – Неправильне повторне введення коду

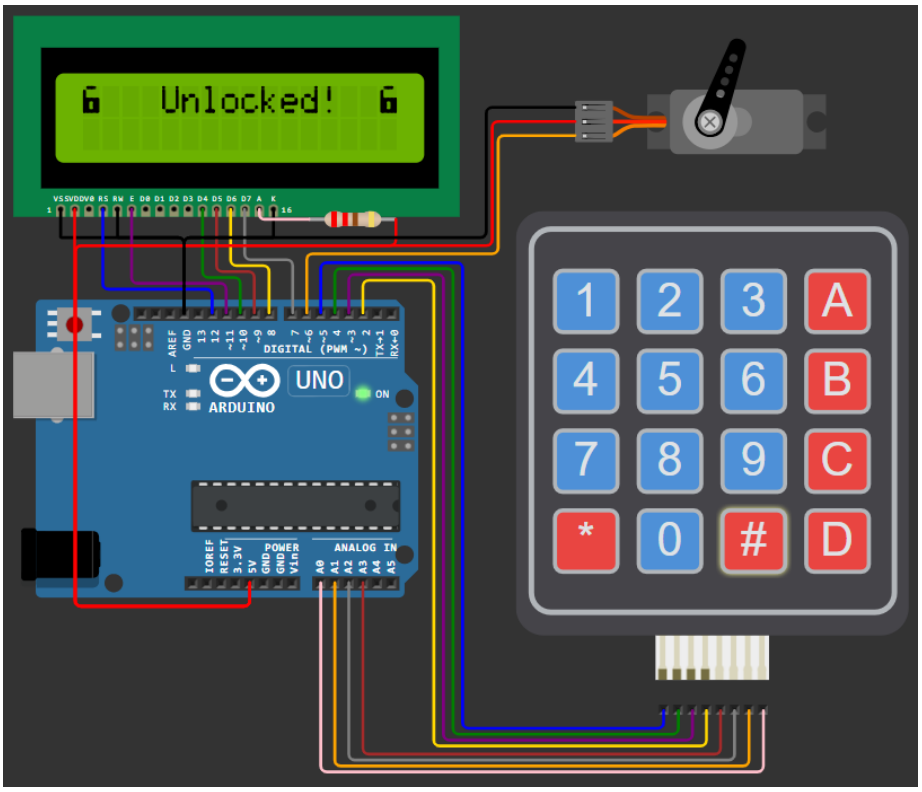

Рисунок 2.18 – Відкриття «замку»

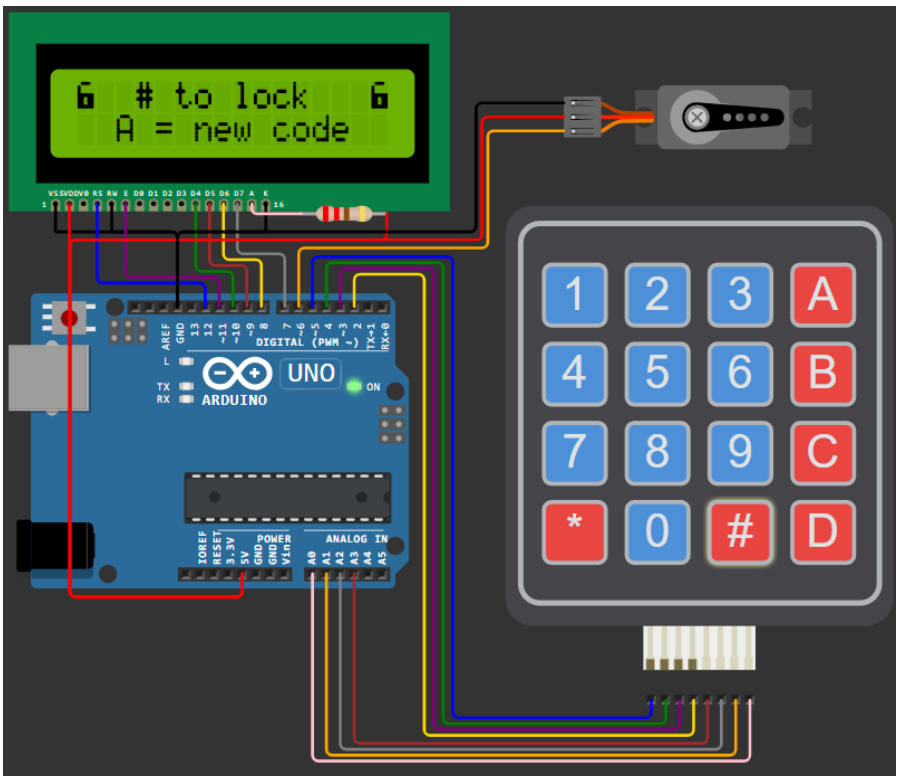

Рисунок 2.19 – Відкриття «замку»

Якщо код неправильний – з’являється повідомлення про помилку, можна ввести ще раз код після 10 секунд (рис. 2.20-2.21).

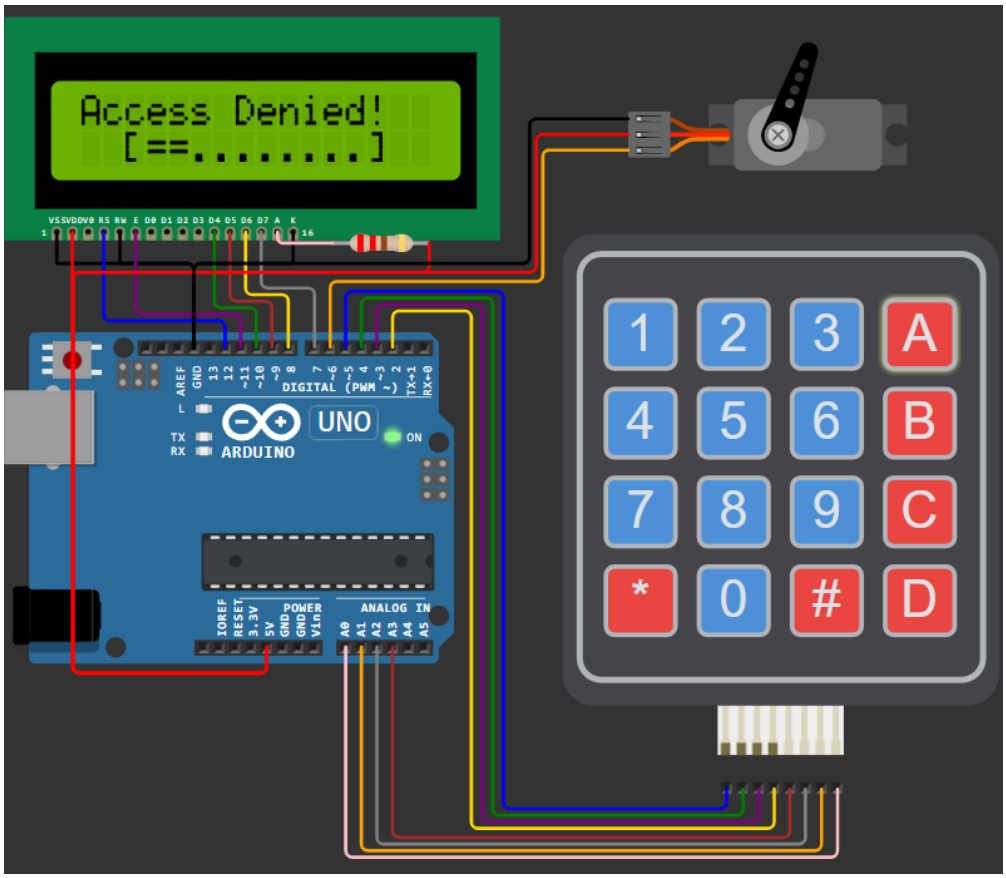

Рисунок 2.20 – Неправильне введення коду

Для додаткового захисту можна:

– обмежити кількість спроб (наприклад, 3);

– додати звукову сигналізацію;

– додати таймер блокування (затримку після помилки).

Переваги системи:

– низька вартість;

– гнучкість і можливість змінити логіку;

– освітнє значення (знайомство з електронікою, програмуванням, безпекою).

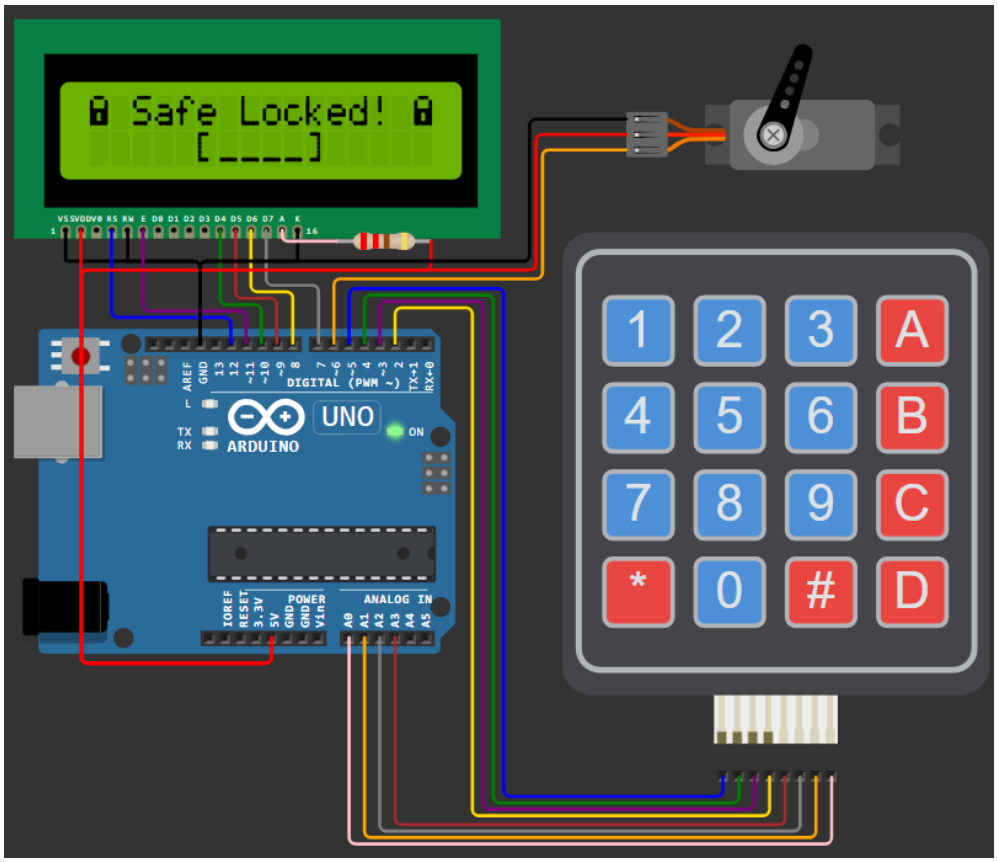

Рисунок 2.21 – Додаткова спроба введення коду

Недоліки системи:

– не гарантує 100% фізичну безпеку (прототип);

– потребує зовнішнього живлення;

– може бути вразливим до злому без належного захисту (наприклад, обхід через Arduino-порт).

Розроблений прототип має свої переваги та недоліки, але якщо базову модель вдосконалити, то в результаті можна отримати надійний та якісний електронний цифровий замок на базі плати Arduino.

Цифровий замок на базі Arduino – це практичний приклад системи доступу, яка може бути використана у побуті, навчанні або як частина більших систем (розумний дім, охорона, IoT).

2.3 Вдосконалення прототипу

Можливості розширення електронного замка представлено на рис. 2.22.

Для підвищення функціональності та зручності використання електронного замка на базі Arduino, систему можна доповнити різними модулями та компонентами:

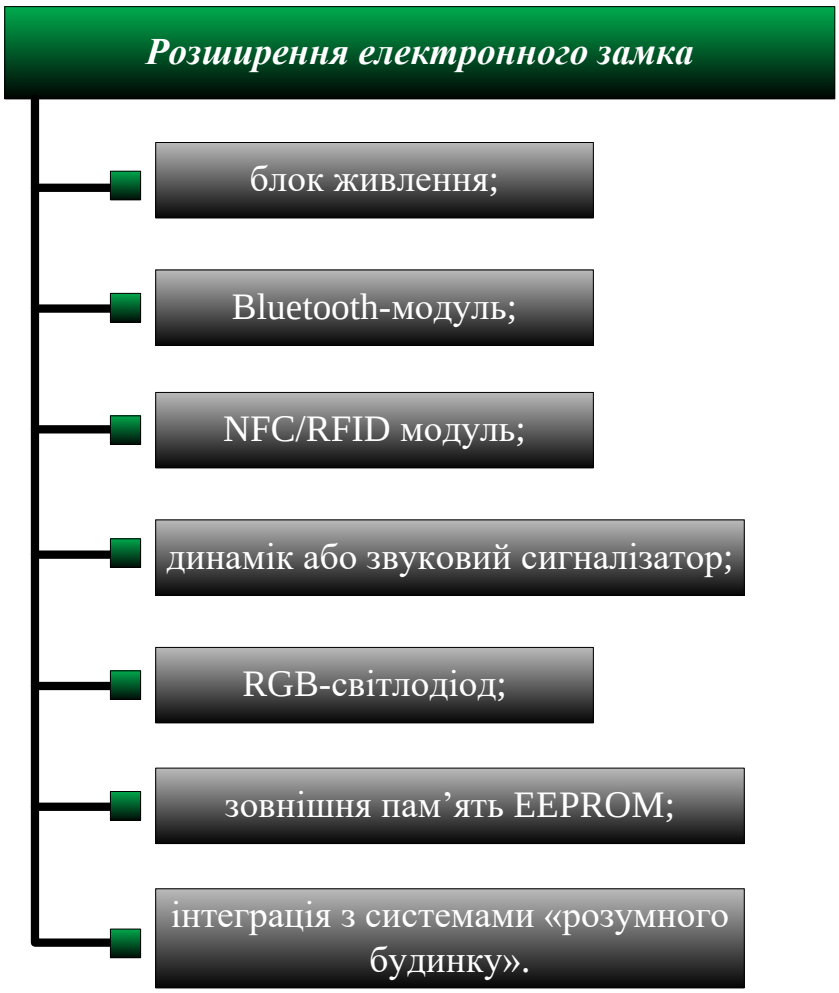

Рисунок 2.22 – Можливості розширення електронного замка

– блок живлення – забезпечує стабільне електроживлення для всієї системи. Може бути реалізований у вигляді адаптера, акумулятора або PowerBank із стабілізатором напруги;

– Bluetooth-модуль (наприклад, HC-05 або HC-06) – дозволяє керувати замком зі смартфона або іншого пристрою через бездротове з'єднання. Це дає можливість відкривати/закривати замок дистанційно через мобільний додаток;

– NFC/RFID модуль (наприклад, RC522) – додає можливість відкриття замка за допомогою RFID-карт або NFC-міток. Це зручно для використання у квартирах, офісах або готельних системах доступу;

– динамік або звуковий сигналізатор (Buzzer) – використовується для подачі звукових сигналів про стан системи — успішний доступ, помилку, введення коду тощо;

– RGB-світлодіод – відображає різні стани замка за допомогою кольорів (наприклад, зелений – доступ дозволено, червоний – заборонено, синій – очікування введення коду);

– зовнішня пам’ять EEPROM – дає змогу зберігати паролі та налаштування навіть після вимкнення живлення. Це дозволяє уникнути втрати даних після перезавантаження пристрою;

– інтеграція з системами «розумного будинку» – замок може бути частиною загальної системи автоматизації будинку (Smart Home), взаємодіяти з іншими пристроями – наприклад, вмикати світло при відкритті дверей або надсилати повідомлення на телефон.

Завдяки використанню мікроконтролера Arduino Uno, серводвигуна, LCD-дисплея та матричної клавіатури, вдалося реалізувати інтерактивну і гнучку у налаштуванні конструкцію. Особливістю цього проєкту є простота його модифікації та розширення — наприклад, за рахунок додавання модулів Bluetooth, RFID/NFC, EEPROM, звукових та світлових індикаторів, а також можливості інтеграції з системами «розумного будинку».

Таким чином, розроблений цифровий замок не лише виконує своє основне завдання – контроль доступу, а й демонструє перспективність використання платформи Arduino для побудови сучасних інтелектуальних електронних систем.

ВИСНОВКИ

У процесі розробки електронного цифрового замка на базі платформи Arduino було створено надійну та доступну систему керування доступом, яка може застосовуватись у побутових, офісних чи навчальних умовах. Система дозволяє відкривати або блокувати механізм замка шляхом введення правильного коду з клавіатури, що забезпечує базовий рівень безпеки.

Завдяки використанню мікроконтролера Arduino Uno, серводвигуна, LCD-дисплея та матричної клавіатури, вдалося реалізувати інтерактивну і гнучку у налаштуванні конструкцію. Особливістю цього проєкту є простота його модифікації та розширення — наприклад, за рахунок додавання модулів Bluetooth, RFID/NFC, EEPROM, звукових та світлових індикаторів, а також можливості інтеграції з системами «розумного будинку».

Таким чином, розроблений цифровий замок не лише виконує своє основне завдання – контроль доступу, а й демонструє перспективність використання платформи Arduino для побудови сучасних інтелектуальних електронних систем.

ПЕРЕЛІК ПОСИЛАНЬ

1. An Overview of Electronic Locks: Technology, Types, and Security Features. [Електронний ресурс] – Режим доступу: https://www.lowratelocksmith.com/locks/electronic-lock/?utm_source=chatgpt.com

2. What are the Different Types of Electronic Locks. [Електронний ресурс] – Режим доступу: https://www.betechiot.com/what-are-the-different-types-of-electronic-locks/?utm_source=chatgpt.com

3. Electronic locking systems & Electronic lock systems. [Електронний ресурс] – Режим доступу: https://cnyglock.com/blog/electronic-locking-systems/?utm_source=chatgpt.com

4. Connecting lock with Arduino Uno. [Електронний ресурс] – Режим доступу: https://forum.arduino.cc/t/connecting-lock-with-arduino-uno/974145?utm_source=chatgpt.com

5. Arduino Uno. [Електронний ресурс] – Режим доступу: https://store-usa.arduino.cc/products/arduino-uno-rev3?srsltid=AfmBOooaWmMLK-ei8BDL4GpHEDutwTQxTTebr1hMB-QDhZlpAfX0uajH&utm_source