Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційних радіотехнологій і технічного захисту інформації
(повна назва)

Кафедра Радіотехнологій інформаційно-комунікаційних систем
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти **перший (бакалаврський)**

ГЮІК.ХХХХХХ.000ПЗ
(позначення документа)

Цифровий замок для сейфа на базі Arduino
(тема)

Виконав:
студент **4** курсу, групи **ТРРТу-21-1**
Балюк Д.О.
(прізвище, ініціали)
Спеціальність **172 Телекомунікації та радіотехніка**
(код і повна назва спеціальності)
Тип програми **освітньо-професійна**
Освітня програма **радіотехніка**
(повна назва освітньої програми)
Керівник **ст. викл. Ганшин Д.Г.**
(посада, прізвище, ініціали)

Допускається до захисту

В. о. зав. кафедри _____ **Зарудний О.А.**
(підпис) (прізвище, ініціали)

2024 р.

Не містить відомостей заборонених до відкритого публікування

Керівник _____ ст. викл. Ганшин Д.Г.

Студент _____ Балюк Д.О.

Харківський національний університет радіоелектроніки

Факультет Інформаційних радіотехнологій і технічного захисту інформації

Кафедра Радіотехнологій інформаційно-комунікаційних систем

Рівень вищої освіти перший (бакалаврський)

Спеціальність 172 Телекомунікації та радіотехніка
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма радіотехніка
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«____»_____ 2024 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові **Балюку Дмитру Олександровичу**
(прізвище, ім'я, по батькові)

1. Тема роботи **Цифровий замок для сейфа на базі Arduino**

затверджена наказом університету від **27 05 2024 р. № 498 Ст**

2. Термін подання студентом роботи до екзаменаційної комісії **10.06.2024р.**

3. Вихідні дані до роботи
Аналіз технологій цифрових замків.
Технічні вимоги до цифрового замка.
Використання платформи arduino для цифрових замків.
Розробка цифрового замка для сейфу.

4. Перелік питань, що потрібно опрацювати в роботі
Вступ 1. Аналіз технологій цифрових замків. 2. Технічні вимоги до цифрового замка. 3. Використання платформи arduino для цифрових замків. 4. Розробка цифрового замку для сейфу. Висновки. Перелік джерел посилання. Додатки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) _____
Комп'ютерна презентація – слайди у форматі Power Point

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

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

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Вступ	06.05.2024	виконано
2	Аналіз технологій цифрових замків	06.05.2024	виконано
3	Технічні вимоги до цифрового замка	07.06.2024	виконано
4	Використання платформи Arduino для цифрових	07.06.2024	виконано
	замків		
5	Розробка цифрового замка для сейфу	08.06.2024	виконано
6	Висновки	08.06.2024	виконано
7	Оформлення пояснювальної записки	09.06.2024	виконано
8	Оформлення ілюстрацій	09.06.2024	виконано
9	Представлення роботи на кафедру	10.06.2024	виконано

Дата видачі завдання **05 травня 2024 р.**

Студент _____ Д.О. Балюк
(підпис)

Керівник роботи _____ ст. викл. Д.Г. Ганшин
(підпис)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи має: 82 сторінок тексту , 22 рисунків, 3 додатка, 15 джерел.

ЦИФРОВИЙ, ЗАМОК, ARDUINO, СЕЙФ, БЕЗПЕКА, ПРОГРАМУВАННЯ, RFID.

Об'єкт дослідження – цифровий замок.

Мета кваліфікаційної роботи– розробити цифровий замок для сейфу .

Розробка надійного та безпечного цифрового замка для сейфу на базі платформи Arduino, що забезпечує високий рівень захисту від несанкціонованого доступу та безперебійну роботу. Це включає в себе створення апаратної та програмної частин замка, інтеграцію його з іншими системами безпеки, забезпечення резервного живлення та реалізацію функцій моніторингу і управління живленням, а також розробку зручного користувацького інтерфейсу для ефективного управління та використання замка.

ABSTRACT

The explanatory note of the qualification work has: 82 pages of text, 22 figures, 3 appendices, 15 sources.

DIGITAL, LOCK, ARDUINO, SAFE, SECURITY, PROGRAMMING, RFID.

The object of research is a digital lock.

The goal of the qualification work is to develop a digital lock for a safe.

Development of a reliable and safe digital lock for a safe based on the Arduino platform, which provides a high level of protection against unauthorized access and uninterrupted operation. This includes the creation of hardware and software parts of the lock, its integration with other security systems, provision of backup power and the implementation of power monitoring and control functions, as well as the development of a convenient user interface for effective management and use of the lock.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

IoT (Internet of Things) - інтернет речей

RFID (Radio-Frequency Identification) - радіочастотна ідентифікація

AES (Advanced Encryption Standard) - розширений стандарт шифрування

RAM - Оперативна пам’ять

EEPROM (Electrically Erasable Programmable Read-Only Memory) - електрично стирана програмована постійна пам'ять.

КБ - Кілобайт

МГц - Мегагерц

В - вольт

PWM – широко-імпульсна модуляція

NFC (Near Field Communication) - технологія бездротового зв'язку на короткій відстані

IDS - Системи виявлення вторгнень

AREF (Analog REFerence) - аналогова довідка

SDA (Serial Data Line) - серійна лінія даних

SCL(Serial Clock Line) - серійна лінія годинника

С – секунди

SPI (Serial Peripheral Interface) - синхронний протокол передачі даних

ВСТУП

У сучасному світі, де безпека є одним із пріоритетних аспектів повсякденного життя, розробка надійних систем захисту стає все більш актуальною. Однією з таких систем є цифрові замки, які забезпечують високий рівень безпеки та зручності у використанні. Використання цифрових технологій дозволяє створювати замки, які можуть бути легко інтегровані з іншими системами безпеки, такими як системи сигналізації, відеоспостереження та інші.

Темою моєї бакалаврської дипломної роботи є розробка цифрового замка на базі платформи Arduino. Вибір платформи Arduino обумовлений її широкою популярністю, доступністю та гнучкістю, що дозволяє створювати складні проекти навіть з обмеженим бюджетом. Arduino надає можливість інтеграції різноманітних датчиків, дисплеїв та виконавчих механізмів, що дозволяє створити функціональний та надійний цифровий замок.

Метою даної роботи є розробка прототипу цифрового замка, який забезпечує високу надійність, простоту у використанні та можливість розширення функціональності. У процесі роботи будуть розглянуті основні компоненти системи: клавіатура для введення коду, RFID-зчитувач для ідентифікації користувачів, серводвигун для керування механізмом замка, а також дисплей для відображення інформації. Окрім цього, будуть розроблені алгоритми роботи замка, які забезпечують захист від несанкціонованого доступу та можливість додавання нових користувачів.

У даному вступі представлено загальний огляд теми та актуальність дослідження. У наступних розділах роботи будуть детально розглянуті теоретичні аспекти розробки цифрових замків, технічні характеристики використовуваних компонентів, процес створення прототипу та результати тестування розробленої системи.

Сподіваюся, що ця робота стане корисним внеском у розвиток сучасних систем безпеки та знайде своє застосування у реальних умовах.

1 АНАЛІЗ ТЕХНОЛОГІЙ ЦИФРОВИХ ЗАМКІВ

1.1 Принцип роботи цифрових замків

Цифрові замки функціонують за рахунок використання електронних та механічних компонентів, які забезпечують контроль доступу до приміщень або об'єктів. Основний принцип роботи цифрового замка полягає у верифікації користувача за допомогою електронних методів аутентифікації та активації механізму замикання на основі результатів цієї верифікації.

На першому етапі користувач вводить дані для аутентифікації через інтерфейс, який може бути представлений у вигляді клавіатури, сенсорного екрану, біометричного сенсора (сканер відбитків пальців, розпізнавання обличчя тощо), або через бездротове з'єднання (смарт-карти, мобільні додатки). Введені дані передаються на електронний контролер замка, який порівнює їх з попередньо збереженими даними в пам'яті пристрою або в хмарному сервісі для підтвердження особи користувача.

Після цього, якщо введені дані збігаються із збереженими, контролер приймає рішення про надання доступу. Якщо введені дані не збігаються із збереженими, доступ не надається, і користувач може отримати повідомлення про невдалу спробу аутентифікації.

У випадку позитивної аутентифікації, контролер надсилає сигнал на електронний механізм замикання (соленоїд або електромотор), який відкриває замок. Після відкриття або через певний проміжок часу замок автоматично закривається, або користувач може закрити його вручну за допомогою того ж інтерфейсу користувача.

Багато цифрових замків оснащені світлодіодними індикаторами або звуковими сигналами, які повідомляють користувача про успішне або невдале відкриття замка. Деякі цифрові замки підтримують підключення до IoT, що дозволяє здійснювати дистанційне керування через мобільні додатки або інші

пристрої. Інформація про всі спроби доступу може зберігатися у внутрішній пам'яті замка або у хмарному сховищі, що дозволяє власникам перевіряти історію доступу і отримувати повідомлення про несанкціоновані спроби входу.

Цифрові замки поєднують у собі електронні та механічні компоненти для забезпечення високого рівня безпеки і зручності використання. Вони дозволяють легко керувати доступом до приміщень і об'єктів, зменшуючи ризик несанкціонованого проникнення.

1.2 Класифікація цифрових замків

Цифрові замки є важливим елементом сучасних систем безпеки, і їх різноманітність дозволяє вибирати оптимальні рішення для різних потреб. Класифікація цифрових замків може бути здійснена за кількома критеріями, такими як метод аутентифікації, тип живлення, спосіб керування та сфера застосування. Кожна з цих категорій включає різні типи замків, що мають свої унікальні характеристики, переваги та недоліки.

Класифікация цифрових замків за методом аутифікації розділяються на кодові замки, біометричні замки, RFID, замки дистанційним керування.

Кодові замки бувають двох основних типів: з клавіатурою та сенсорною панеллю.

– Замки з клавіатурою передбачають введення коду на фізичній клавіатурі, що забезпечує простоту використання та доступність, але вони можуть бути вразливими до зломів через підглядання або зчитування відбитків пальців на клавішах;

– Замки з сенсорною панеллю використовують сенсорні екрани для введення коду, що робить їх більш гнучкими та зручними у використанні, але вони також можуть бути вразливими через відбитки пальців на екрані;

Біометричні замки включають кілька типів: замки, що використовують відбитки пальців, розпізнавання обличчя та сканування сітківки ока.

– Відбитки пальців забезпечують високий рівень безпеки та зручність, але можуть бути проблематичними для людей з ушкодженими або брудними пальцями;

– Розпізнавання обличчя використовує камеру для ідентифікації, що забезпечує високу безпеку та зручність, але може мати труднощі при поганому освітленні або зміні зовнішності користувача;

– Сканування сітківки ока є дуже надійним методом, але дорогим і може бути некомфортним для користувачів;

– RFID-системи використовують карткові системи або брелоки і мітки. Карткові системи дозволяють доступ за допомогою RFID-карти, що є зручним, але карти можуть бути втрачені або викрадені. Брелоки та мітки з вбудованими RFID-чіпами також забезпечують зручність, але можуть бути втрачені або викрадені;

– Магнітні картки забезпечують доступ шляхом проведення картки через зчитувач. Цей метод є дешевим і поширеним, але картки легко підробляються;

– Смарт-картки, на відміну від магнітних, мають вбудовані мікропроцесори, які можуть зберігати додаткову інформацію та забезпечувати підвищену безпеку. Вони більш надійні, але й дорожчі;

– Замками дистанційного керування може здійснюватися за допомогою Bluetooth або Wi-Fi. Bluetooth-замки відкриваються через мобільний додаток за допомогою Bluetooth-з'єднання, що дуже зручно, але вразливе до атак, якщо зловмисник знаходиться поблизу. Wi-Fi-замки можуть керуватися віддалено через Інтернет, що дозволяє контролювати доступ на великій відстані. Це зручно, але потребує стабільного Інтернет-з'єднання і є вразливим до кібер-атак.

Замки класифікуються за типом живлення: автономні, мережові, комбіновані.

– Автономні живляться від батарей або акумуляторів, що дозволяє їх використовувати без підключення до електромережі. Це забезпечує мобільність та незалежність, але потребує регулярної заміни батарей;

– Мережеві підключені до електричної мережі, що забезпечує постійне живлення. Вони не залежать від заміни батарей, але можуть бути непридатні у випадку відключення електрики;

Комбіновані можуть працювати як від батарей, так і від електромережі, забезпечуючи резервне живлення у випадку відключення електрики. Це поєднує переваги обох попередніх методів, але ускладнює конструкцію замка;

Замки котрі класифікуються за способом керуванням бувають: механічні, електронні та комбіновані(механічно-електронні).

– Механічні цифрові замки поєднують механічний механізм з цифровою системою аутентифікації. Вони надійні і можуть працювати без електрики, але менш гнучкі у налаштуванні;

– Електронні цифрові замки повністю електронні, без механічних частин. Вони дозволяють легко змінювати налаштування і програмувати різні режими роботи, але залежать від джерела живлення;

– Механічно-електронні замки мають як механічні, так і електронні компоненти, забезпечуючи додаткову надійність. Вони поєднують переваги обох типів, але можуть бути більш складними в обслуговуванні;

Класифікуються замки також за сферою їх застосування: домашні фоісні, сейфові, автомобільні, промислові. Та за середовищем використання : внутрішні, зовнішні, замки для транспортних засобів.

– Домашні замки використовуються для захисту житлових приміщень. Вони зазвичай прості у використанні і мають середній рівень безпеки;

– Офісні замки застосовуються в офісних будівлях для захисту робочих приміщень та конфіденційних документів. Вони мають вищий рівень безпеки і можуть бути інтегровані з іншими системами контролю доступу;

– Сейфові замки спеціально розроблені для використання в сейфах, забезпечуючи високий рівень безпеки для цінних речей. Вони можуть мати додаткові функції, такі як захист від вибуху чи механічного впливу;

– Автомобільні замки використовуються для захисту автомобілів. Вони можуть бути інтегровані з іншими системами безпеки автомобіля, такими як сигналізація та іммобілайзер;

– Внутрішні замки призначені для використання всередині приміщень. Вони зазвичай мають менший рівень захисту від впливу зовнішнього середовища, але забезпечують високу зручність та естетичний вигляд;

– Зовнішні замки призначені для використання на відкритому повітрі або в умовах підвищеної вологості і температурних коливань. Вони мають захист від пилу, вологи та інших несприятливих умов;

– Замки для транспортних засобів спеціально розроблені для захисту автомобілів, мотоциклів та інших транспортних засобів. Вони повинні бути стійкими до вібрацій, ударів та інших механічних впливів.

Замкі поділяються на три рівня захисту:

– Базовий рівень замки, що забезпечують мінімальний рівень захисту, підходять для приміщень з невисокими вимогами до безпеки. Вони часто використовуються в домашніх умовах

– Середній рівень замки, які забезпечують помірний рівень захисту, підходять для офісів та житлових приміщень. Вони мають додаткові функції безпеки, такі як додаткове шифрування або контроль доступу;

– Високий рівень замки, які забезпечують максимальний рівень захисту, підходять для сейфів, банківських установ та інших об'єктів з високими вимогами до безпеки. Вони часто використовують біометричні методи аутентифікації та мають багаторівневі системи захисту.

1.3 Порівняння замків різних виробників

Ринок цифрових замків на сьогоднішній день пропонує широкий спектр продуктів від різних виробників, які відрізняються за функціональністю, надійністю, ціною та іншими характеристиками.

Компанія August виробляє цифрові замки, які підтримують інтеграцію з мобільними додатками та системами розумного дому, надаючи функції віддаленого доступу та моніторингу. До ключових моделей входять August Smart Lock Pro та August Wi-Fi Smart Lock. [12]

Schlage, компанія спеціалізується на виробництві цифрових та смарт-замків, які оснащені клавіатурою, біометричними датчиками, та підтримують мобільні додатки. До основних моделей належать Schlage Encode, Schlage Sense, та Schlage Connect. [13]

Компанія Yale виробляє цифрові замки з функцією віддаленого керування, біометричними датчиками та інтеграцією з системами розумного дому, доступні у моделях Yale Assure Lock, Yale Real Living та Yale Conexis L1. [14]

Компанія Kwikset виготовляє цифрові замки, які можна керувати через мобільні додатки, клавіатуру, із функцією автоматичного блокування. Серед їх моделей - Kwikset Halo, Kwikset Premis та Kwikset Obsidian. [15]

Компанія Samsung виробляє цифрові замки, що мають біометричні датчики, карткові системи, PIN-коди та можливість інтеграції з розумними домами. До їх моделей належать Samsung SHS-3321, Samsung SHP-DP609 та Samsung SHP-DH538. [16]

Можемо порівняємо замки зазначених виробників за такими критеріями: методи аутентифікації, функціональні можливості, інтеграція з іншими системами, ціна та надійність.

Методи аутентифікації:

– August: мобільний додаток, Bluetooth, Wi-Fi;

– Schlage: клавіатура, Bluetooth, Wi-Fi;

– Yale: клавіатура, мобільний додаток, Bluetooth, Wi-Fi, біометрія;

– Kwikset: клавіатура, Bluetooth, Wi-Fi, Z-Wave;

– Samsung: PIN-код, RFID-карта, біометрія (відбиток пальця), мобільний додаток;

Функціональні можливості:

– August: віддалене управління, автоматичне блокування, історія доступу;

– Schlage: віддалене управління, автоматичне блокування, можливість інтеграції з системами безпеки;

– Yale: віддалене управління, інтеграція з розумним домом, біометрія, можливість використання одноразових кодів;

– Kwikset: віддалене управління, голосовий контроль через смарт-колонки, автоматичне блокування;

– Samsung: автоматичне блокування, інтеграція з системами розумного дому, віддалене управління, багатофакторна аутентифікація;

Інтеграція з іншими системами:

– August: сумісність з Alexa, Google Assistant, Apple HomeKit, Z-Wave;

– Schlage: сумісність з Alexa, Google Assistant, Apple HomeKit, SmartThings;

– Yale: сумісність з Alexa, Google Assistant, Apple HomeKit, SmartThings, Z-Wave;

– Kwikset: сумісність з Alexa, Google Assistant, Apple HomeKit, Z-Wave;

– Samsung: сумісність з SmartThings, Samsung Smart Home;

Ціна:

– August: середній ціновий діапазон ($200-$250);

– Schlage: середній ціновий діапазон ($150-$300);

– Yale: середній до високий ціновий діапазон ($200-$350);

– Kwikset: середній ціновий діапазон ($150-$250);

– Samsung: середній до високий ціновий діапазон ($200-$400);

Надійність:

– August: висока надійність, постійні оновлення програмного забезпечення, підтримка користувачів;

– Schlage: відмінна репутація, надійність механізмів, довговічність;

– Yale: висока надійність, особливо у біометричних моделях, довговічність;

– Kwikset: надійність, добрі відгуки користувачів, довговічність;

– Samsung: висока якість збірки, надійність електроніки, гарні відгуки;

Порівняння цифрових замків різних виробників показує, що кожен з них має свої сильні та слабкі сторони. Вибір конкретного замка залежить від індивідуальних потреб користувача, бюджету та вимог до функціональності.

– August підходить для тих, хто цінує інтеграцію з мобільними додатками та простоту використання;

– Schlage забезпечує високу надійність і довговічність з великим спектром функцій;

– Yala пропонує широкі можливості інтеграції з системами розумного дому і біометричні рішення;

– Kwikset відомий своєю сумісністю з голосовими асистентами та віддаленим управлінням;

– Samsung пропонує інноваційні рішення з біометричними методами аутентифікації та високою надійністю;

1.4 Інтеграція цифрових замків з іншими системами безпеки

Інтеграція цифрових замків з іншими системами безпеки є важливим аспектом для забезпечення комплексного підходу до захисту приміщень. Така інтеграція дозволяє створювати синергічний ефект, підвищуючи загальний

рівень безпеки та зручність управління системою. Розглянемо основні аспекти інтеграції цифрових замків з іншими системами безпеки.

Інтеграція цифрових замків з іншими системами безпеки забезпечує кілька важливих переваг:

– Централізоване управління: можливість контролювати та керувати всіма аспектами безпеки з одного централізованого інтерфейсу. Це дозволяє операторам швидко реагувати на будь-які інциденти;

– Підвищення рівня безпеки: об'єднання різних систем дозволяє підвищити загальний рівень безпеки. Наприклад, інтеграція з системами відеоспостереження дозволяє фіксувати всі спроби несанкціонованого доступу;

– Зручність використання: інтегровані системи дозволяють користувачам використовувати одну платформу для управління всіма аспектами безпеки, що спрощує їх використання та знижує кількість помилок;

– Автоматизація процесів: інтеграція дозволяє автоматизувати багато процесів, таких як автоматичне блокування дверей у разі тривоги або надання доступу до приміщень у певний час;

Існують кілька основних способів інтеграції цифрових замків з іншими системами безпеки:

– Інтеграція з системами контролю доступу: цифрові замки можуть бути інтегровані з системами контролю доступу, які використовують різні методи аутентифікації (RFID-карти, біометрія, коди). Це дозволяє створювати єдину систему управління доступом до всіх приміщень;

– Інтеграція з відеоспостереженням: замки можуть інтегровані з системами відеоспостереження для фіксації всіх подій, пов'язаних із відкриттям і закриттям дверей. Це допомагає оперативно реагувати на спроби несанкціонованого доступу.

– Інтеграція з пожежною сигналізацією: у разі пожежної тривоги цифрові замки можуть автоматично розблокуватися для забезпечення швидкої

евакуації людей із приміщення. Така інтеграція значно підвищує рівень безпеки під час надзвичайних ситуацій;

– Інтеграція з системами домашньої автоматизації: цифрові замки можуть бути частиною системи розумного будинку, що дозволяє керувати ними через мобільний додаток або голосові асистенти. Це підвищує зручність використання та дозволяє інтегрувати замки з іншими пристроями, такими як освітлення, термостати тощо;

Інтеграція цифрових замків з іншими системами безпеки також пов'язана з певними викликами, такими як сумісність різних систем, забезпечення кібербезпеки та надійність зв'язку між компонентами. Для вирішення цих викликів необхідно:

– Використання стандартів: застосування стандартних протоколів зв'язку, таких як Z-Wave, Zigbee або Wi-Fi, забезпечує сумісність між різними системами;

– Кібербезпека: забезпечення надійного шифрування даних і захисту від кіберзагроз є критично важливими для інтегрованих систем безпеки. Використання сучасних методів аутентифікації та регулярне оновлення програмного забезпечення допомагають запобігти несанкціонованому доступу;

– Надійність зв'язку: Забезпечення стабільного та надійного зв'язку між компонентами системи є важливим для її безперебійної роботи. Використання резервних каналів зв'язку та дублювання критичних компонентів допомагає підвищити надійність системи;

Цифрові замки мають значні переваги, зокрема зручність використання, можливість інтеграції з іншими системами та підвищений рівень безпеки. Однак, існують і певні виклики, такі як необхідність забезпечення надійного захисту від кібератак та фізичних зломів, а також потенційні проблеми з електроживленням і стабільністю роботи програмного забезпечення.

Інтеграція цифрових замків з іншими системами безпеки є важливим кроком для створення ефективної та надійної системи захисту приміщень. Вона забезпечує централізоване управління, підвищує рівень безпеки, зручність використання та автоматизацію багатьох процесів. Незважаючи на певні виклики, такі як забезпечення сумісності та кібербезпеки, сучасні технології дозволяють створювати інтегровані системи, що відповідають найвищим стандартам безпеки.

1.5 Програмне забезпечення для цифрових замків

Програмне забезпечення для цифрових замків є важливою складовою, що забезпечує їх функціональність, безпеку та інтеграцію з іншими системами. В цій частині розглянемо основні аспекти програмного забезпечення для цифрових замків, зокрема мови програмування, використані бібліотеки, а також основні алгоритми та функції, які реалізуються для забезпечення роботи замка.

Цифрові замки зазвичай базуються на мікроконтролерах, для програмування яких використовуються мови, такі як C або C++. Наприклад, для розробки програмного забезпечення на базі платформи Arduino використовується мова програмування, що базується на C/C++, доповнена спеціальними бібліотеками, що спрощують роботу з апаратними компонентами. Arduino IDE є популярним середовищем розробки для написання, налагодження та завантаження коду на мікроконтролери.

Використані бібліотеки

Для забезпечення функціональності цифрового замка використовується низка спеціалізованих бібліотек. Основні з них включають:

– LiquidCrystal для роботи з РК-дисплеями, що відображають статус замка та інші повідомлення;

– Keypad для обробки введення даних з клавіатури;

– Servo для керування серводвигуном, що фізично відкриває або закриває замок;

– EEPROM для збереження паролів та інших важливих даних, які мають зберігатися навіть після вимкнення живлення;

Програмне забезпечення цифрового замка включає декілька ключових алгоритмів та функцій:

– Ініціалізація системи. Під час запуску програми відбувається налаштування всіх необхідних компонентів, таких як дисплей, клавіатура та серводвигун. Встановлюються початкові значення для змінних та конфігурацій.

– Введення та перевірка пароля. Користувач вводить пароль за допомогою клавіатури. Програма перевіряє правильність введеного пароля, порівнюючи його із збереженим у пам'яті пристрою. Якщо пароль правильний, замок відкривається або закривається відповідно до обраної логіки.

– Управління серводвигуном. Серводвигун відповідає за фізичне переміщення механізму замка. Програмне забезпечення контролює кут повороту серводвигуна, забезпечуючи надійне відкривання та закривання замка.

– Збереження даних. Використання EEPROM дозволяє зберігати паролі та інші конфігурації, які повинні залишатися доступними після вимкнення живлення. Це забезпечує стабільність роботи системи та зручність для користувачів.

– Безпека та захист. Програмне забезпечення повинно включати механізми для захисту від несанкціонованого доступу. Наприклад, після кількох невдалих спроб введення пароля система може блокувати замок на певний час, або вимагати додаткову автентифікацію.

Сучасні цифрові замки можуть інтегруватися з іншими системами безпеки та автоматизації, такими як системи розумного будинку, системи відеоспостереження, та системи управління доступом. Це дозволяє

створювати комплексні рішення, які підвищують загальний рівень безпеки та комфорту.

Завдяки програмному забезпеченню цифрові замки стають гнучкими та функціональними пристроями, здатними забезпечити надійний захист та зручність у використанні. Правильна розробка та налаштування програмного забезпечення є ключем до успішної роботи та довговічності таких систем.

2 ТЕХНІЧНІ ВИМОГИ ДО ЦИФРОВОГО ЗАМКА

2.1 Основні технічні характеристики

Для розробки надійного та функціонального цифрового замка необхідно визначити основні технічні характеристики, які будуть відповідати вимогам безпеки, зручності використання та ефективності роботи. Ключові технічні параметри, що визначають функціональність і надійність замка:

- Мікроконтролер. ;
- Живлення;
- Вхідні та вихідні порти;
- Безпека;
- Периферійні пристрої;
- Надійність та довговічність;

Основні технічні характеристики цифрового замка повинні забезпечувати високу надійність, безпеку та зручність використання, що робить його ефективним рішенням для захисту сейфів та інших об'єктів.

2.1.1 Програмування та налаштування

Середовище програмування Arduino IDE надає інтуїтивно зрозумілий інтерфейс для написання, компіляції та завантаження коду на мікроконтролер. У ньому доступна велика кількість готових бібліотек, що спрощує розробку з використанням різних датчиків і модулів.

Під час написання коду для цифрового замка в Arduino IDE спочатку ініціалізуються компоненти, проводиться підключення і налаштування датчиків, модулів і виконавчих механізмів. Далі відбувається обробка введення з клавіатури, RFID-зчитувача або сканера відбитків пальців, після

чого відбувається аутентифікація користувача, перевірка введених даних або зчитаних ідентифікаторів з базою даних.

Після успішної аутентифікації здійснюється управління замком, активація сервомотора або соленоїда для відкриття або закриття замка залежно від результату перевірки. На завершення, виконується індикація стану, виведення інформації на дисплей або використання світлодіодних індикаторів для інформування користувача про статус замка.

2.2 Резервне живлення та аварійне вимкнення

Забезпечення безперебійної роботи цифрового замка є критично важливим для його надійності та функціональності. Важливим аспектом цього є наявність резервного живлення та механізмів аварійного вимкнення.

Резервним живленням служать літій-іонні або літій-полімерні акумулятори, що забезпечують автономну роботу у разі відключення основного джерела живлення. Забезпечать роботу замка протягом 24-48 годин у разі відключення основного живлення. Використання спеціалізованих контролерів для керування зарядом акумуляторів і запобігання їх перезаряду або глибокому розряду.

Моніторинг стану живлення є важливою складовою забезпечення безперебійної роботи цифрового замка. Він дозволяє своєчасно виявляти проблеми з живленням і вживати необхідних заходів для підтримки працездатності системи.

Завдяки комплексному підходу до контролю та управління живленням, система забезпечує надійну та безперебійну роботу навіть у випадку аварійних ситуацій, що гарантує високу надійність і безпеку експлуатації замка.

2.4 Вимоги до цифрового замка для сейфу

Цифровий замок для сейфу має забезпечувати високий рівень безпеки і надійності, враховуючи специфічні вимоги до захисту цінностей і важливих документів. Ось основні вимоги, які слід враховувати при виборі цифрового замка для сейфу:

- Високий рівень безпеки;
- Надійність та довговічність;
- Зручність використання;
- Автономність;

Вибір цифрового замка для сейфу повинен базуватися на поєднанні високого рівня безпеки, надійності, зручності використання і додаткових функцій, що підвищують загальну ефективність та надійність системи. Дотримання цих вимог допоможе забезпечити захист ваших цінностей і важливих документів на найвищому рівні.

3 ВИКОРИСТАННЯ ПЛАТФОРМИ ARDUINO ДЛЯ ЦИФРОВИХ ЗАМКІВ

3.1 Обгрунтування вибору платформи Arduino та переваги

Arduino є інноваційним засобом розробки програмованих електронних систем, який відрізняється від звичайних комп'ютерів тим, що акцентує на тісній взаємодії з фізичним середовищем. Це відкрита платформа, яка поєднує апаратне забезпечення (мікроконтролери) з власним середовищем розробки програмного забезпечення. За допомогою Arduino можна створювати різноманітні інтерактивні системи, що реагують на зовнішні впливи через датчики та управляють актуаторами. Вона дає можливість розробляти проекти, які можуть працювати автономно або взаємодіяти з ПК та іншими пристроями.

Незалежно від рівня досвіду, Arduino доступна для всіх, оскільки має просту мову програмування, що базується на "Wiring" та "Processing". Це дозволяє швидко освоїти основи програмування для створення власних електронних проектів.

Крім того, Arduino відкрита для всіх користувачів, оскільки має відкритий вихідний код та безкоштовне середовище розробки. Це сприяє активному обміну знаннями та розвитку великої спільноти користувачів, що надають підтримку одне одному у вирішенні різних завдань та проблем.

Існує безліч інших мікроконтролерів і мікропроцесорних пристроїв, призначених для програмування різних апаратних засобів: Parallax Basic Stamp, Netmedia's BX-24, Phidgets, Handyboard від MIT і багато інших. Усі ці пристрої пропонують подібну функціональність і призначені звільнити користувача від необхідності занурюватися в дрібні деталі внутрішнього устрою мікроконтролерів, надаючи йому простий і зручний інтерфейс для їх програмування. [8]

Arduino також спрощує процес роботи з мікроконтролерами, але, на відміну від інших систем, надає низку переваг для викладачів, студентів і аматорів радіолюбителів:

- Низька вартість;
- Кросплатформенність;
- Просте і зручне середовище програмування;
- Розширюване програмне забезпечення з відкритим вихідним кодом;
- Розширюване відкрите апаратне забезпечення;

3.2 Приклади реалізації цифрових замків на базі Arduino

Цифровий замок з кодовою клавіатурою. Складається з плати Arduino Uno, матрична клавіатура, сервомотор, LCD-дисплей. Принцип роботи дуже простий, користувач вводить код на клавіатурі. Якщо код вірний, сервомотор повертається для відкриття замка. Стан відображається на дисплеї.

RFID-замок складається з кількох основних компонентів, включаючи Arduino Nano, RFID-зчитувач, електромагнітний замок і світлодіодний індикатор. Принцип роботи цього замка наступний: користувач підносить RFID-карту до зчитувача. Зчитувач перевіряє ідентифікатор карти з базою даних. Якщо ідентифікатор підтверджується, електромагнітний замок розблоковується, а світлодіодний індикатор змінює колір, сигналізуючи про успішний доступ.

Біометричний замок, створений на базі Arduino Mega, включає в себе такі компоненти як сканер відбитків пальців, соленоїд та OLED-дисплей. Принцип роботи цього замку полягає в наступному: користувач прикладає свій палець до сканера відбитків пальців. Сканер аналізує відбиток і порівнює його з збереженими записами у пам'яті. Якщо відбиток палець визнається як валідний, соленоїд активується і відкриває замок. Результат перевірки, а також статус замка відображаються на OLED-дисплеї.

Замок з розпізнаванням обличчя, реалізований на платформі Arduino, включає такі компоненти як камера, модуль Wi-Fi та електромеханічний замок. Робота цього замку відбувається наступним чином: камера захоплює образ обличчя користувача і відправляє його для обробки. Алгоритми розпізнавання обличчя аналізують зображення, порівнюючи його з даними у базі даних. У випадку успішного розпізнавання обличчя, електромеханічний замок активується і відкривається. Окрім того, система має змогу відправляти сповіщення на мобільний телефон адміністратора через Wi-Fi, забезпечуючи додатковий контроль і безпеку доступу.

3.3 Переваги та обмеження використання Arduino для цифрових замків

Використання платформи Arduino в цифрових замках має як переваги, так і обмеження. Зокрема, вона відрізняється модульністю, що дозволяє легко додавати та модифікувати компоненти відповідно до вимог проекту. Завдяки доступності великої кількості бібліотек та прикладів коду, значно скорочується час, необхідний на розробку. Платформа також забезпечує велику гнучкість у налаштуванні, що дозволяє створювати унікальні рішення для специфічних потреб користувачів.

Проте, серед обмежень Arduino обмежена обчислювальна потужність, яка може бути недостатньою для реалізації складних алгоритмів або обробки великих обсягів даних. Крім того, базові засоби безпеки, що інтегровані у платформу, можуть бути недостатніми для захисту від складних атак, що змушує розробників вдаватися до додаткових методів шифрування та захисту даних. Також слід зазначити залежність системи від якості зовнішніх компонентів, як-от датчиків та інших електронних елементів, що можуть вплинути на надійність і довговічність замка.

Незважаючи на зазначені обмеження, Arduino залишається популярним вибором для розробки надійних та функціональних систем контролю доступу,

пропонуючи розробникам можливість створення ефективних рішень, адаптованих під конкретні потреби користувачів.

3.4 Технічні характеристики платформи Arduino

Платформа Arduino стала основою для численних проектів у сфері електроніки, робототехніки та автоматизації завдяки своїй гнучкості, доступності та широкому спектру можливостей. Технічні характеристики основних моделей Arduino, які найчастіше використовуються для розробки цифрових замків.

3.4.1 Arduino Uno

Arduino Uno – це популярна мікроконтролерна плата, широко використовується для створення електронних проектів та прототипів. Вона базується на мікроконтролері ATmega328PP і є частиною екосистеми Arduino, яка включає апаратне та програмне забезпечення з відкритим вихідним кодом. Arduino Uno розроблена для зручного використання навіть початківцями, пропонуючи просту у засвоєнні IDУ та обширну спільноту, яка підтримує користувачів через форуми, навчальні матеріали та бібліотеки коду. Плата має багато цифрових та аналогових входів/виходів, що дозволяє підключати різноманітні сенсори, двигуни, світлодіоди та інші компоненти, роблячи її універсальним інструментом для вивчення та реалізації різноманітних електронних проектів. Приклад плати рисунок 3.1. [1]

Основні технічні характеристики включають:

– Мікроконтролер: ATmega328P;

– RAM: 2 КБ;

– Флеш-пам'ять: 32 КБ;

– EEPROM: 1 КБ;

– Кількість цифрових входів/виходів: 14 (з них 6 можуть використовуватися як PWM-виходи);

– Аналогові входи: 6;

– Тактова частота: 16 МГц;

– Робоча напруга: 5 В;

– Вхідна напруга: 7-12 В;

– Максимальна вхідна напруга: 6-20 В;

– Порти для підключення: USB, ICSP, роз'єм живлення;

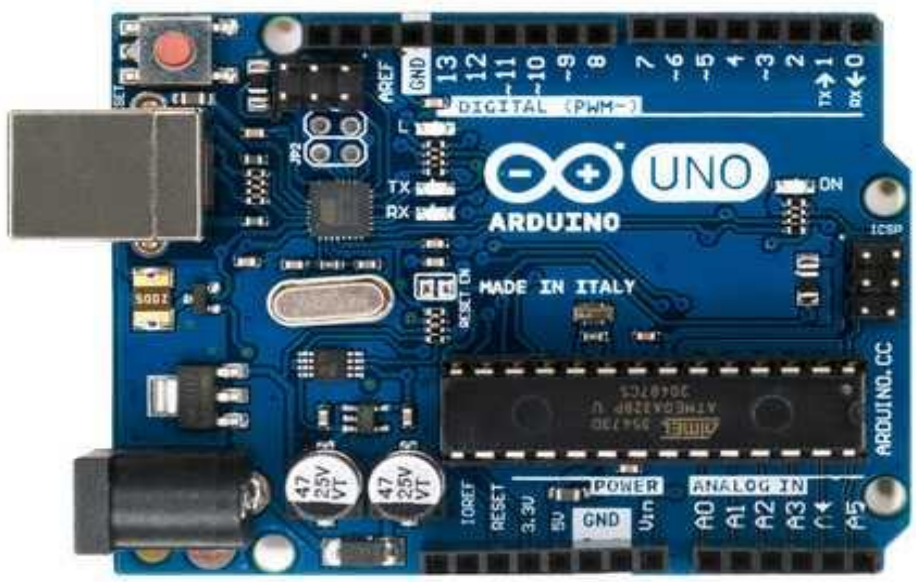

Рисунок 3.1- Arduino Uno

3.4.2 Arduino Nano

Arduino Nano – це компактна мікроконтролерна плата, призначена для створення електронних проектів, де важливі малі розміри та низьке

споживання енергії. Вона базується на мікроконтролері ATmega328P і є частиною екосистеми Arduino, яка включає апаратне та програмне забезпечення з відкритим вихідним кодом. Arduino Nano має подібні можливості до Arduino Uno, але в більш компактному форм-факторі, що робить її ідеальною для використання в проектах з обмеженим простором. Плата оснащена великою кількістю цифрових та аналогових входів/виходів, що дозволяє підключати різні сенсори, двигуни, світлодіоди та інші компоненти. Завдяки своїм невеликим розмірам і універсальності, Arduino Nano часто використовується в портативних пристроях та інтегрованих системах. Приклад плати рисунок 3.2. [2]

Основні технічні характеристики:

- Мікроконтролер: ATmega328P;
- RAM: 2 КБ;
- Флеш-пам'ять: 32 КБ;
- EEPROM: 1 КБ;
- Кількість цифрових входів/виходів: 14;
- Аналогові входи: 8;
- Тактова частота: 16 МГц;
- Робоча напруга: 5 В;
- Вхідна напруга: 7-12 В;
- Максимальна вхідна напруга: 6-20 В;
- Роз'єм для підключення: Mini-USB;

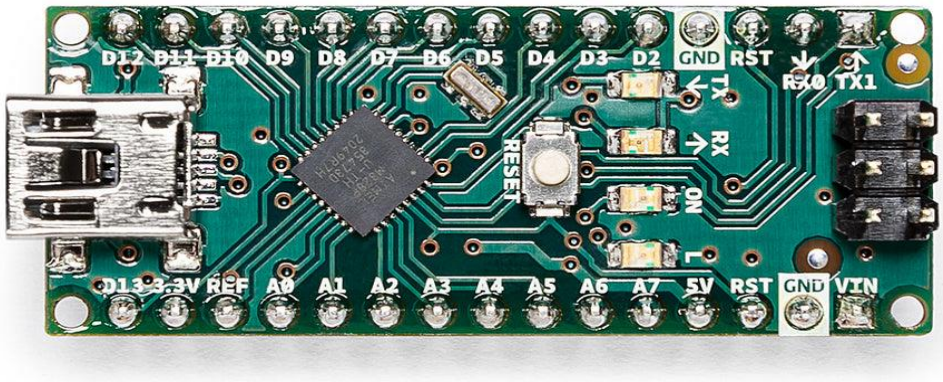

Рисунок 3.2 - Arduino Nano

3.4.3 Arduino Mega

Arduino Mega – це потужна мікроконтролерна плата, призначена для складніших електронних проектів, що потребують більшої кількості входів/виходів та пам'яті. Вона базується на мікроконтролері ATmega2560 і є частиною екосистеми Arduino, яка включає апаратне та програмне забезпечення з відкритим вихідним кодом. Arduino Mega має значно більше цифрових та аналогових входів/виходів у порівнянні з іншими платами Arduino, що дозволяє підключати численні сенсори, дисплеї, двигуни та інші компоненти одночасно. Вона особливо корисна для великих проектів, таких як робототехніка, 3D-принтери та автоматизовані системи. Arduino Mega підтримується тією ж простою у використанні IDE та має велику спільноту, яка надає підтримку через форуми, навчальні матеріали та бібліотеки коду. Приклад плати на рисунку 3.3. [3]

Технічні характеристики:

– Мікроконтролер: ATmega2560;

– RAM: 8 КБ;
– Флеш-пам'ять: 256 КБ;
– EEPROM: 4 КБ;
– Кількість цифрових входів/виходів: 54;
– Аналогові входи: 16;
– Тактова частота: 16 МГц;
– Робоча напруга: 5 В;
– Вхідна напруга : 7-12 В;
– Максимальна вхідна напруга: 6-20 В;
– Порти для підключення: USB, ICSP, роз'єм живлення ;

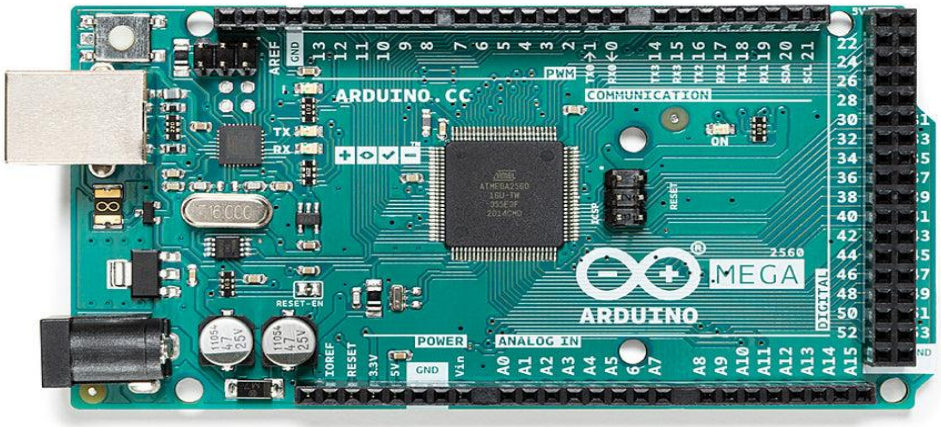

Рисунок 3.3 - Arduino Mega

3.4.4 Arduino Due

Arduino Due – це потужна мікроконтролерна плата, призначена для проектів, що потребують високої продуктивності. Вона базується на 32-бітному мікроконтролері Atmel SAM3X8E ARM Cortex-M3 і є частиною

екосистеми Arduino, яка включає апаратне та програмне забезпечення з відкритим вихідним кодом. Arduino Due має значно більшу обчислювальну потужність та більшу кількість пам'яті в порівнянні з іншими платами Arduino. Вона оснащена великою кількістю цифрових та аналогових входів/виходів, що дозволяє підключати численні сенсори, дисплеї, двигуни та інші компоненти одночасно. Плата також підтримує USB OTG (On-The-Go), що розширює можливості підключення зовнішніх пристроїв. Завдяки своїй високій продуктивності та широким можливостям, Arduino Due підходить для складних проектів, таких як робототехніка, автоматизовані системи управління та інші високопродуктивні застосунки. Вона підтримується тією ж простою у використанні IDE та має велику спільноту, яка надає підтримку через форуми, навчальні матеріали та бібліотеки коду. Приклад плати рисунку 3.5.

Технічні характеристики :

- Мікроконтролер: AT91SAM3X8E (ARM Cortex-M3);
- RAM: 96 КБ;
- Флеш-пам'ять: 512 КБ;
- EEPROM: відсутня (можливе використання зовнішньої пам'яті);
- Кількість цифрових входів/виходів: 54;
- Аналогові входи: 12;
- Аналогові виходи : 2;
- Тактова частота: 84 МГц;
- Робоча напруга: 3.3 В;
- Рекомендована напруга: 7-12 В;
- Максимальна вхідна напруга: 6-16 В;
- Порти для підключення: USB, JTAG, роз'єм живлення;

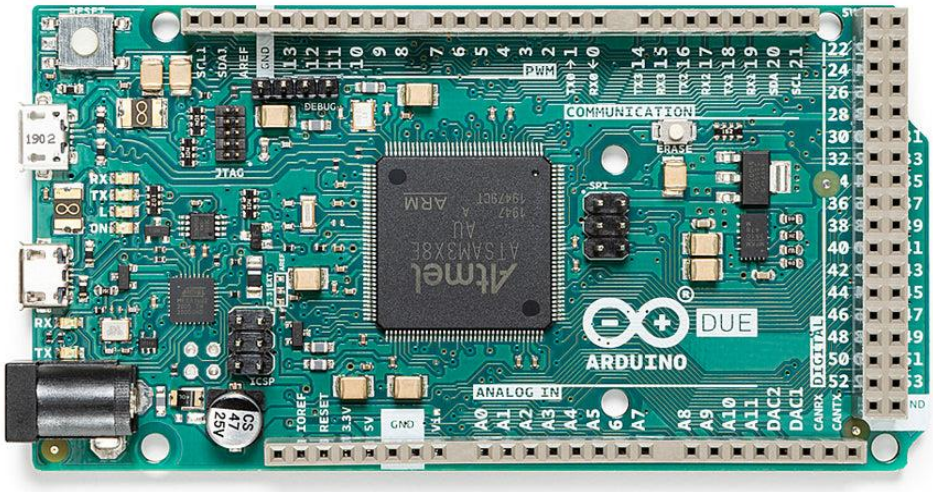

Рисунок 3.4 - Arduino Due

Кожна з моделей Arduino має свої переваги та недоліки, що робить їх придатними для різних типів проектів:

– Arduino Uno: оптимальний вибір для початківців та простих проектів з обмеженою кількістю входів/виходів;

– Arduino Nano: ідеально підходить для компактних проектів, де важливий розмір плати;

– Arduino Mega: забезпечує багато входів/виходів і більше пам'яті для складних проектів з численними компонентами;

– Arduino Due: забезпечує високу обчислювальну потужність для ресурсомістких додатків, таких як обробка даних у реальному часі;

Платформа Arduino пропонує різноманітність моделей, що дозволяє вибрати оптимальну плату для будь-якого проекту, включаючи розробку цифрових замків. Завдяки своїм технічним характеристикам, простоті використання та великій спільноті користувачів, Arduino забезпечує надійну

основу для створення інноваційних рішень у сфері електроніки та автоматизації.

3.5 Методи програмування цифрового замка на Arduino

Для розробки цифрового замка на платформі Arduino використовуються різні методи програмування, які дозволяють створювати функціональний та надійний замок з цифровим керуванням. У цьому підрозділі ми розглянемо основні методи програмування, що використовуються при розробці цифрових замків на Arduino.

Програмування за допомогою Arduino IDE є основним інструментом для програмування плат Arduino, включаючи Arduino Uno для розробки цифрових замків. Основні методи програмування в Arduino IDE включають складання коду на мові Arduino: Програмний код для цифрового замка пишеться на мові Arduino, яка базується на мові програмування C/C++ з використанням бібліотек Arduino. Після написання програмного коду, його можна завантажити на плату Arduino Uno через USB-підключення для тестування та виконання функцій замка. Для реалізації функцій цифрового замка на Arduino Uno можна використовувати різні бібліотеки, які допомагають спростити розробку та покращити функціональність замка. Для розробки цифрових замків широко використовуються різноманітні бібліотеки, що дозволяють покращити їх функціональність та надійність. Основні з них включають Keypad Library, яка дозволяє з'єднати клавіатуру з Arduino Uno для введення паролів або кодів доступу; RFID Library, яка призначена для роботи з RFID сенсорами та картками для автоматичного розблокування замка; Servo Library, що використовується для керування серводвигуном та відкривання/закривання замка; LiquidCrystal Library, яка дозволяє виводити інформацію на LCD дисплей про стан замка або процес введення пароля; та EEPROM Library, що забезпечує зберігання даних, таких як паролі та коди

доступу, у вбудованій пам'яті Arduino. Використання цих бібліотек дозволяє покращити функціональність цифрових замків та забезпечити їх ефективну роботу. [5]

3.6 Програмування за допомогою Arduino Sketches

Arduino Sketches - це програмні коди, що виконуються на платі Arduino Uno. Для розробки цифрових замків можна створювати різні скетчі, такі як скетч для керування замком, який містить логіку управління замком на основі введеного коду або RFID, скетч для відображення інформації на дисплеї про стан замка чи введення пароля, а також скетч для збереження інформації про доступні коди, паролі або дані користувачів у внутрішній пам'яті EEPROM плати.[6] [7]

У програмуванні цифрових замків також використовуються графічні інтерфейси для спрощення процесу налаштування та управління:

– MIT App Inventor: інструмент для створення мобільних додатків, що дозволяє керувати цифровим замком через Bluetooth або Wi-Fi;

– Blynk: платформа для створення IoT-додатків з графічним інтерфейсом, що дозволяє швидко створювати мобільні додатки для управління замком;

– Node-RED: інструмент для візуального програмування IoT, який дозволяє створювати потоки даних для керування цифровим замком через веб-інтерфейси;

Методи програмування цифрових замків на базі Arduino пропонують широкий спектр можливостей для розробників, дозволяючи створювати ефективні та надійні рішення для безпеки. Використання стандартних бібліотек, розробка власних алгоритмів, інтеграція з іншими технологіями та інструментами, а також використання передових середовищ розробки

дозволяють створювати високоякісні цифрові замки з розширеним функціоналом.

7 Інтеграція цифрових замків на основі платформи Arduino

Інтеграція замків на платформи Arduino з іншими системами безпеки забезпечує високий рівень безпеки об'єкта та додаткові функціональні можливості. У даному випадку, ми розглянемо основні методи та підходи до інтеграції цифрових замків з іншими системами безпеки.

Інтеграція з системами відеоспостереження полягає у взаємодії з різними типами камер. Arduino може взаємодіяти з IP-камерами через Ethernet або Wi-Fi модулі для запису відео під час спроби доступу. Також, за допомогою додаткових модулів, можлива інтеграція з аналоговими камерами.

Для активізації запису можна використовувати функцію детекції руху або тригери доступу через замок. Це дозволить запускати запис відео при виявленні руху або неправильного вводу коду доступу.

Інтеграція з системами сигналізації також є важливою. Arduino може надсилати сигнали для активації сирен або звукових сповіщувачів при виявленні спроби несанкціонованого доступу. Крім того, можна використовувати GSM-модулі для надсилання SMS або здійснення дзвінків при спробі взлому.

Для зв'язку з централізованою охоронною системою Arduino може інтегруватися через стандартні протоколи зв'язку, такі як RS485 або Modbus, та забезпечувати віддалене управління та моніторинг стану замка через централізовану охоронну панель.

Інтеграція цифрових замків на базі платформи Arduino з біометричними системами відкриває широкі можливості для підвищення рівня безпеки та функціональності. Основні аспекти цієї інтеграції включають використання біометричних сенсорів, обробку та зберігання даних, інтеграцію з системами

розумного дому, синхронізацію з іншими пристроями, інтеграцію з мережевими сервісами та використання IoT.

Спочатку, для використання біометричних сенсорів, можна встановити сканери відбитків пальців для додаткового рівня аутентифікації, а також використовувати сенсори обличчя та голосу для безпечного доступу.

Далі, важливою частиною інтеграції є обробка та зберігання даних. Це включає локальне зберігання біометричних даних у вбудованій пам'яті або на SD-карті, а також захищене зберігання за допомогою алгоритмів шифрування.

Для інтеграції з системами розумного дому, можна використовувати підтримку стандартних протоколів та налаштовувати автоматизовані сценарії, які включають роботу цифрового замка разом з іншими пристроями розумного дому.

Також, можна синхронізувати замок з датчиками дверей та вікон для моніторингу стану, інтегрувати з освітленням для автоматичного увімкнення світла та інших дій для покращення безпеки.

Інтеграція з мережевими сервісами, такими як хмарні сервіси та IoT-платформи, дозволяє здійснювати віддалений моніторинг та керування замком через Інтернет, зберігати дані у хмарі для подальшого аналізу, а також налаштовувати автоматичні сповіщення про стан замка через IoT-платформи.

Ця інтеграція цифрових замків на базі Arduino з іншими системами безпеки дозволяє створювати комплексну та надійну систему захисту, яка забезпечує високий рівень безпеки для користувачів та об'єктів.

4 РОЗРОБКА ЦИФРОВОГО ЗАМКА ДЛЯ СЕЙФУ

4.1 Вибір компонентів

Компоненти:

– Плата Arduino Uno;

– Клавіатура;

– LCD дисплей;

– Серводвигун;

– RFID модуль використовується для альтернативного відкриття замку.

– Динамік (Spiker / Buzzer);

– RGB світлодіод;

На рис. 4.1 наведена структурна схема.

Рисунок 4.1 – Структурна схема підключення компонентів цифрового замку

Відповідно рис 4.1 для користуванням сейфу потрібно ввести пароль або прикласти картку до RFID модуля. Модуль RFID зчитує картку і порівнює її з збереженими даними. Після того користувач вводить код через клавіатуру, який відображається на LCD дисплеї як зірочки. Далі іде перевірка коду у

головному циклі. Порівняння введеного коду з збереженим кодом у системі. Якщо код правильний, замок відкривається, і серводвигун переміщується в позицію відкриття. Успішне або неуспішне введення коду супроводжується звуковими сигналами та зміною кольору RGB світлодіода.

Клавіатура під'єднана до цифрового інтерфейсу на аналогові піни які використовуються як цифрові. Серводвигун та світлодіод до інтерфейсу PWM. RFID під'єднано до SPI. Динамік та LCD дисплей під'єднані до цифрового інтерфейсу.

Підєднується усі компоненти до плати Arduino Uno у такі виходи:

– Клавіатура (ряди) A3, A2, A1, A0;

– Клавіатура (стовпці) A4, A5, A6, A7;

– LCD дисплей (RS) 12;

– LCD дисплей (E) 11;

– LCD дисплей (D4) 5;

– LCD дисплей (D5) 4;

– LCD дисплей (D6) 3;

– LCD дисплей (D7) 2;

– Серводвигун 6;

– RFID модуль (SS) 10;

– RFID модуль (RST) 9;

– Динамік (Buzzer) 13;

– RGB світлодіод (червоний) 7;

– RGB світлодіод (зелений) 8;

4.2 Плата Arduino Uno

Ця система побудована на основі плати Arduino Uno, яка використовує контролер ATmega328P. Платформа має в своєму розпорядженні 14 цифрових

входів/виходів (з яких 6 можуть бути використані як виходи ШІМ), 6 аналогових входів, кварцовий генератор з частотою 16 МГц, USB-порт, силовий роз'єм, роз'єм ICSP і кнопку перезавантаження. Для роботи системи необхідно підключити Arduino Uno до комп'ютера за допомогою USB-кабелю або ж подати живлення через адаптер AC/DC або батарею.

Arduino Uno може отримувати живлення через підключення до USB або зовнішнього джерела живлення, причому вибір джерела живлення відбувається автоматично.

Зовнішнє живлення (окрім USB) може надаватися через перетворювач напруги AC/DC або від акумуляторної батареї. Перетворювач напруги підключається через роз'єм 2.1 мм з центральним позитивним полюсом. Провідники від батареї підключаються до виводів Gnd і Vin роз'єму живлення.

Платформа може працювати з зовнішнім живленням від 6 В до 20 В. З підводом напруги менше 7 В, вивід 5V може надавати менше 5 В, що може вплинути на стабільність роботи. При застосуванні напруги понад 12 В, регулятор напруги може перегрітися і пошкодити плату. Оптимальний діапазон напруги від 7 В до 12 В рекомендується для безпечної роботи платформи.

4.2.1 Виводи живлення

Плата Arduino Uno, як і більшість мікроконтролерів, потребує живлення для своєї роботи. Розглянемо різні види живлення, які можна використовувати для плати Arduino Uno, включаючи внутрішнє живлення, зовнішні джерела енергії та альтернативні методи живлення, на рис. 4.2 приведена схема.

VIN. Цей вивід призначений для підведення живлення з зовнішнього джерела, коли немає доступу до 5 В через USB або іншого регульованого джерела живлення. Подача напруги живлення здійснюється через цей вивід.

5V. Це регульоване джерело напруги, яке використовується для живлення мікроконтролера та компонентів на платі. Напруга може подаватися з виводу VIN через регулятор напруги, через USB або інше регульоване джерело напруги 5 В.

3V3. Напруга на цьому виводі становить 3.3 В і регулюється вбудованим регулятором на платі. Максимальний струм, що споживається, складає 50 мА.

GND. Вивід заземлення.

Порт ISCP призначений для програмування контролера через USB-інтерфейс, зокрема для здійснення операцій, які недоступні через стандартний послідовний інтерфейс, такі як запис завантажувача і зміна фьюз (fuses), що визначають поведінку контролера.

Виводи мікроконтролера розділяються на цифрові (D0-D13) і аналогові (A0-A5), хоча в програмуванні використовується "пряма" нумерація виводів, де, наприклад, вивід 13 = D13, 14 = A0, 15 = A1 і так далі.

Цифрові виводи можуть використовуватися як входи або виходи (режим вибирається в програмі), тоді як аналогові використовуються як входи з можливістю АЦП на 10 біт (від 0 до 1023) і максимальною межею вимірювань 5 В відносно землі або виведенням AREF.

Виводи D0 і D2 призначені для передачі даних через асинхронний послідовний порт і підключені до USB-serial контролера. Вони не можуть бути підключені безпосередньо до порту RS323 через використання TTL-послідовного інтерфейсу, несумісного з RS232 і потребують перетворення.

Виводи D2 і D3 можуть використовуватися для генерації зовнішніх переривань.

Виводи D3, D5, D6, D9, D10 і D11 пов'язані з внутрішніми лічильниками-таймерами мікроконтролера і можуть використовуватися для виведення сигналів ШИМ або як лічильники зовнішніх імпульсів.

Виводи D10-D13 можуть використовуватися для взаємодії з зовнішніми пристроями за допомогою протоколу SPI, з виводом D10 (SS), що використовується у випадку, коли мікроконтролер є приймачем (slave).

Вивід D13 підключений до світлодіоду "L" на платі, що не впливає на його функціональність, але може слугувати для індикації певних подій.

Два виводи I2C в верхньому ряду повторюють A4 і A5 і можуть використовуватися для роботи з зовнішніми пристроями за протоколом I2C, що є додатковою можливістю A4 і A5.

Вивід Vin призначений для подачі зовнішнього живлення, яке потім проходить через регулятор напруги.

Виводи GND, 5V, 3V3 - земля і регульовані напруги 5В і 3,3В відповідно.

Вивід IOREF видає робочу напругу 5 В.

Таким чином, можливості вводу-виводу досить широкі. Аналогові входи дозволяють вимірювати напругу сигналу (можна навіть створити осцилограф, але частота вимірювань буде обмежена швидкістю процесора). Цифрові входи-виходи можуть як зчитувати, так і встановлювати стан, включаючи генерацію ШІМ сигналів, що зазвичай використовуються для управління двигунами або генерації звуку. Крім того, через різні інтерфейси можна взаємодіяти із зовнішніми пристроями: односпрямована шина на основі будь-якого цифрового виводу, асинхронний послідовний порт, I2C, SPI.

I2C та SPI дозволяють підключати багато пристроїв до однієї шини одночасно. Більшість сенсорів для Arduino підключаються через аналогові входи, односпрямовану шину або I2C.SPI зазвичай використовується для пристроїв, що вимагають високої швидкості передачі даних (Ethernet shield, WiFi shield).Для експериментів дуже корисно мати Sensor shield - плату з зручним дублюванням всіх виводів разом із заземленням і живленням для підключення зовнішніх пристроїв. Також зручною є макетна плата для швидкого підключення пристроїв і пасивних компонентів без пайки.

Кожен з 14 цифрових виводів Arduino Uno може бути налаштований як вхід або вихід за допомогою функцій pinMode(), digitalWrite(), і digitalRead(). Ці виводи працюють при напрузі 5 В. Кожен вивід оснащений навантажувальним резистором (за замовчуванням відключений) з опором 20-50 кОм і може пропускати струм до 40 мА. Деякі з виводів мають спеціальні функції.

Послідовна шина 0 (RX) і 1 (TX), використовуються для прийому (RX) і передачі (TX) TTL даних. Ці виводи підключені до відповідних виводів мікросхеми послідовної шини ATmega8U2 USB-to-TTL.

Зовнішні переривання 2 і 3, можуть бути налаштовані на виклик переривання на низькому рівні, передньому чи задньому фронті, або при зміні рівня.

ШІМ 3, 5, 6, 9, 10, і 11. Забезпечують широтно-імпульсну модуляцію (ШІМ) з роздільною здатністю 8 біт за допомогою функції analogWrite().

SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK), використовуються для зв'язку по протоколу SPI за допомогою бібліотеки SPI.

Світлодіод 13, вбудований світлодіод підключений до цифрового виводу 13. Якщо на виводі встановлено високий рівень, світлодіод світиться.

На платі Uno є 6 аналогових входів (позначених як A0 .. A5) з роздільною здатністю 10 біт (тобто можуть приймати 1024 різних значень). Стандартно, виводи можуть вимірювати напругу в діапазоні до 5 В відносно землі, проте верхню межу можна змінити за допомогою виводу AREF і функції analogReference().

Деякі аналогові виводи мають додаткові функції. I2C, 4 (SDA) і 5 (SCL). Використовуються для зв'язку по протоколу I2C (TWI) за допомогою бібліотеки Wire.

Платформа Arduino Uno оснащена кількома засобами зв'язку для взаємодії з комп'ютером, іншими пристроями Arduino або мікроконтролерами. Мікроконтролер ATmega328 підтримує UART TTL (5 В) послідовний

інтерфейс, що реалізується через виводи 0 (RX) і 1 (TX). Цей інтерфейс направляється через USB за допомогою встановленої на платі мікросхеми ATmega8U2, що дозволяє комп'ютерним програмам "спілкуватися" з платою через віртуальний COM-порт.

Прошивка ATmega8U2 використовує стандартні драйвери USB COM, тому сторонні драйвери не потрібні. Однак для підключення на Windows може знадобитися файл ArduinoUNO.inf. Моніторинг послідовної шини (Serial Monitor) в Arduino IDE дозволяє надсилати та отримувати текстові дані при підключенні до платформи. Світлодіоди RX і TX на платі миготять під час передачі даних через мікросхему FTDI або USB, але не при використанні послідовної передачі через виводи 0 і 1.

Бібліотека SoftwareSerial дозволяє створювати послідовну передачу даних через будь-який цифровий вивід на платі Uno.

Мікроконтролер ATmega328 підтримує також інтерфейси I2C (TWI) і SPI. Для зручності використання шини I2C в Arduino включена бібліотека Wire.

Платформа Arduino програмується за допомогою спеціалізованого програмного забезпечення Arduino IDE. Мікроконтролер ATmega328 постачається з попередньо записаним завантажувачем, що значно полегшує процес завантаження нових програм без використання зовнішніх програматорів. Передача даних здійснюється за допомогою оригінального протоколу STK500.

Існує також можливість обійти завантажувач і запрограмувати мікроконтролер безпосередньо через виводи ICSP (внутрішньосхемне програмування).

Плата Arduino Uno спроєктована таким чином, що перезавантаження перед завантаженням нового коду здійснюється автоматично програмним шляхом через Arduino IDE на комп'ютері, а не вручну натисканням кнопки на платформі. Одна з ліній DTR мікросхеми ATmega8U2, яка керує потоком

даних, підключена до виводу перезавантаження мікроконтролера ATmega328 через конденсатор ємністю 100 нФ. Активація цієї лінії, тобто подача сигналу низького рівня, призводить до перезавантаження мікроконтролера. Програма Arduino IDE використовує цю функцію для завантаження коду одним натисканням кнопки Upload у середовищі програмування. Сигнал низького рівня по лінії DTR координується з початком запису коду, що скорочує час очікування завантажувача.

Ця функція також корисна під час роботи з операційними системами Mac OS X або Linux. Кожного разу, коли Arduino підключається до комп'ютера через USB, плата перезавантажується. Після перезавантаження завантажувач активується протягом наступних півсекунди. Під час програмування це забезпечує затримку, щоб уникнути отримання некоректних даних платформою. Якщо потрібне одноразове налагодження записаного скетчу або введення інших даних після першого запуску, програма на комп'ютері повинна почекати близько секунди перед передачею даних.

На платі Uno також є можливість відключити автоматичне перезавантаження шляхом розриву відповідної лінії. Контакти мікросхем з обох кінців цієї лінії можна з'єднати для відновлення функції. Лінія позначена як «RESET-EN». Відключити автоматичне перезавантаження можна також підключивши резистор 110 Ом між джерелом 5 В і цією лінією.

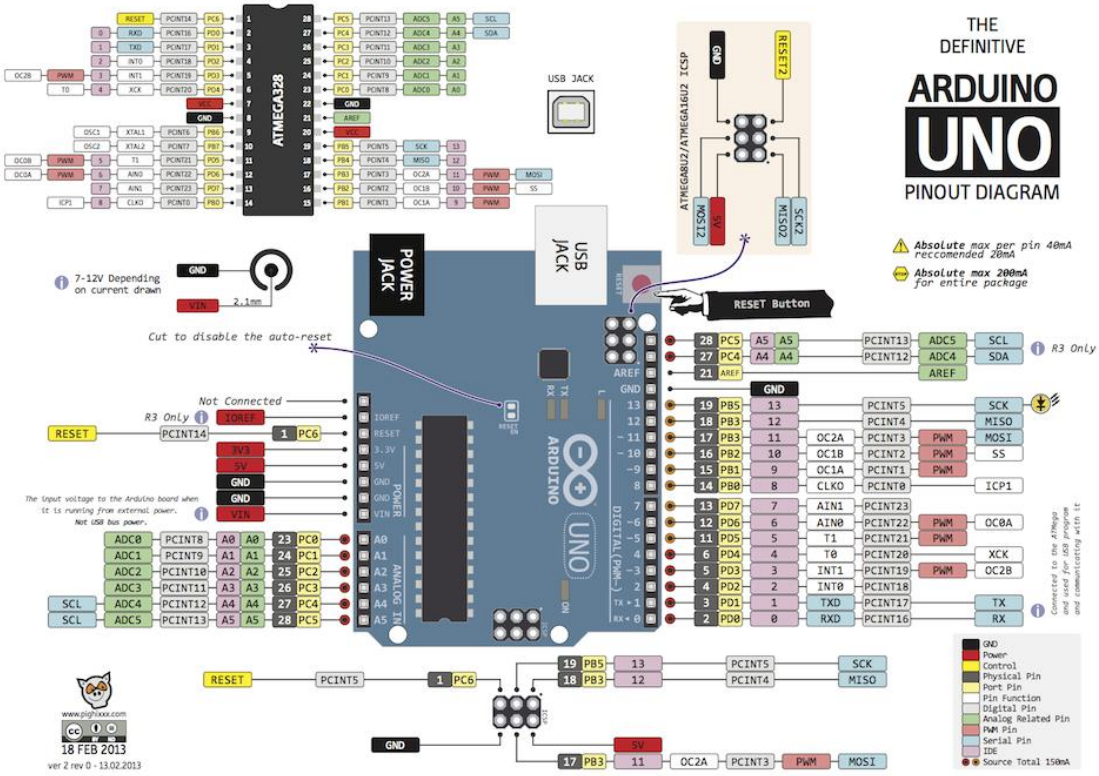

Рисунок 4.2 - Схема виводи плати Arduino

4.3 Дисплей

У кодовому замку дисплей використовується для відображення інформації про стан замка та взаємодії з користувачем. Рисунок 3.2. З'єднання з Arduino здійснюється за допомогою протоколу I2C, який використовує чотири виводи: SDA, SCL, GND, і VCC. Вивід SDA дисплея підключається до виводу A4, а вивід SCL — до A5 на платі Arduino. Інші два виводи забезпечують живлення дисплея. Рисунок зображення дисплея 4.3.

Рисунок 4.3 – LCD дисплей

4.4 Клавіатура

Клавіатура використовується для введення паролю. На рис. 4.4 зображена сама матрична клавіатура 16 клавіш. Вона призначена для роботи з Arduino, AVR, PIC, ARM та іншими мікроконтролерами. Вона дозволяє підключити велику кількість кнопок до пристрою, використовуючи менше портів. Для підключення 16 кнопок матричної клавіатури потрібно лише 8 пінів (4 стовпці та 4 рядки), а не 16.

Клавіатура 4x4 може застосовуватися для управління різними пристроями, введення кодів у кодових замках та програмування. Для її використання необхідно підключити клавіатуру до керуючого пристрою, створити відповідну програму управління та забезпечити живлення. Принцип роботи клавіатури полягає в тому, що при натисканні будь-якої клавіші замикається певна пара проводів.

Рисунок 4.4 – Зовнішній вигляд клавіатури

4.5 Серводвигун

Серводвигун є ключовим компонентом у цифровому замку на базі Arduino, оскільки він відповідає за фізичне переміщення механізму замка. Це електромеханічний пристрій, який може точно контролювати положення, швидкість і прискорення. Серводвигун складається з двигуна постійного струму, редуктора, потенціометра та системи управління.

Основні етапи його роботи включають:

– Сигнал управління: Серводвигун отримує сигнал управління від мікроконтролера (у нашому випадку від Arduino). Цей сигнал зазвичай має форму ШІМ, де тривалість імпульсу визначає положення вала двигуна;

– Перетворення сигналу: Система управління серводвигуна інтерпретує сигнал ШІМ і порівнює його з поточним положенням вала, яке вимірюється потенціометром;

– Регулювання положення: На основі цієї інформації система управління вносить корекції, подаючи відповідний струм на двигун постійного струму. Це змушує двигун обертатися в потрібному напрямку до досягнення заданого положення;

– Зворотний зв'язок: Потенціометр постійно відстежує положення вала і передає ці дані назад у систему управління, яка продовжує регулювати струм до досягнення стабільного положення;

4.6 RFID модуль

Модуль RFID RC522 є одним з найпопулярніших зчитувачів RFID, який використовує радіочастотну ідентифікацію для безконтактного читання та запису даних на RFID-мітках. Модуль RFID RC522 зображен на рис. 4.5

Ініціалізація модуля Підключення модуля RC522 до мікроконтролера за допомогою SPI-інтерфейсу. Зчитувач починає випромінювати радіосигнали для активації міток у своєму діапазоні. Модуль RC522 зчитує дані з RFID-мітки, які містяться у її пам'яті. Отримані дані можна обробити у мікроконтролері для подальших дій, наприклад, відкриття дверей, контролю доступу тощо. У цього модуля є свої переваги та недоліки.[10]

Переваги модуля RC522:

– Можливість зчитувати інформацію з RFID-міток без прямого контакту;

– Швидкий обмін даними між модулем і міткою;

– Легка інтеграція з мікроконтролерами, зокрема з Arduino;

Обмеження модуля RC522:

– Мітки повинні бути відносно близько до зчитувача (до 10 см).

– В деяких випадках існує ризик перехоплення сигналу зчитувача

Модуль RFID RC522 знаходить широке застосування у сферах автоматизації, контролю доступу, систем безпеки та інших областях, де потрібне безконтактне ідентифікування об'єктів чи осіб.

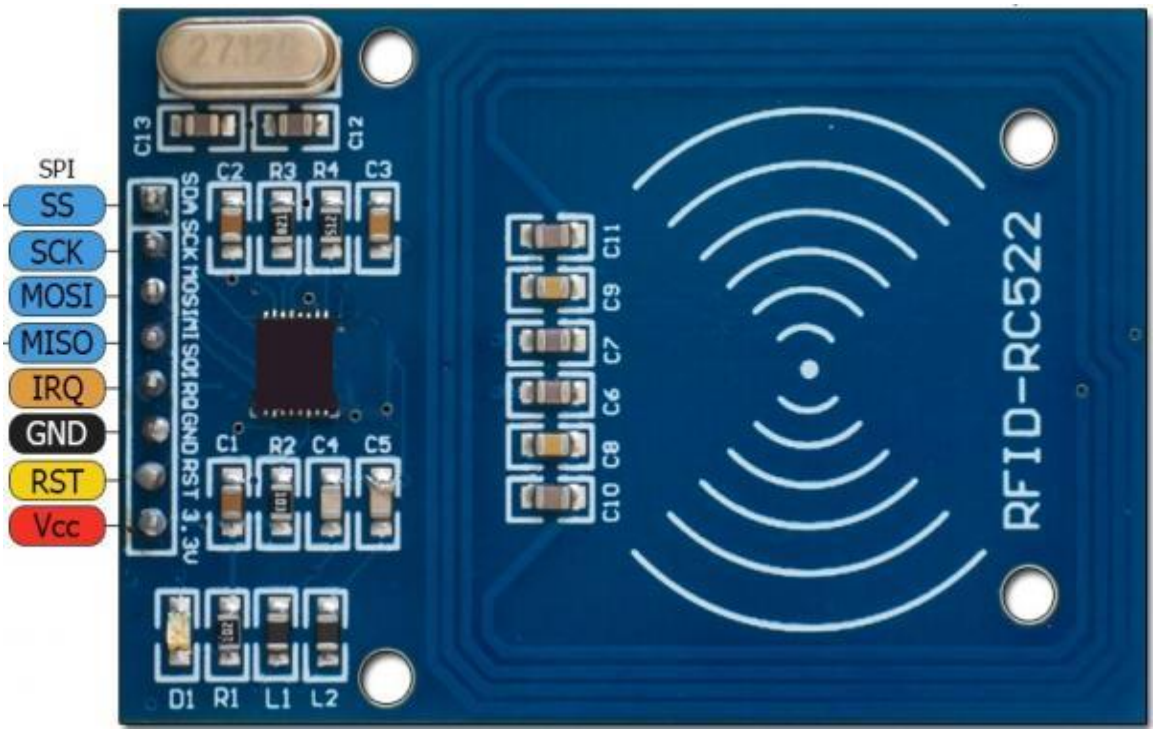

Рисунок 4.5 - Модуль RFID RC522

4.7 Світлодіод та динамік

Світлодіод та динамік відповідають за систему сповіщення відкриття, закриття замка. При відкритті замка горить зелений колір, та спрацьовую сигнал, закритті червоний. Також при помилках чи відмові у доступі, вже буде інший сигнал і світлодіод загорить червоним.

4.8 Програмування цифрового замку

Мова програмування для пристроїв Arduino базується на C/C++ та скомпонована з бібліотекою AVR Libc, що дозволяє використовувати будь-які її функції. Водночас, вона є простою в освоєнні, і на сьогодні Arduino, мабуть, є найзручнішим способом програмування пристроїв на мікроконтролерах. Мову Arduino можна поділити на чотири розділи: оператори, дані, функції та бібліотеки.

4.8.1 Докладний розбір коду

Перши дії це додавання потрібних нам бібліотек, рис. 4.6 : дисплея, клавіатури, серводвигуна та бібліотек котрі відповідні за запам’ятовування пароля та іконки на дисплеї. Також бібліотеки RFID модуля та SPI . [21] [22] [23] [24] [25]

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

Рисунок 4.6 – Бібліотеки

Далі Ініціалізуємо сервопривід, який буде керувати механізмом замка. Градус повороту в закритому положенні та відкритому. Під’єднаємо його до 6 віходу

#define SERVO_LOCK_POS 20 поворот у закритому положенні

#define SERVO_UNLOCK_POS 90 поворот у відкритому положенні

#define SERVO_PIN 6

Servo lockServo;

Створюємо об'єкт дисплея, для виведення повідомлень користувачу. Та під’єднаємо його до плати.

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

Налаштовуємо RGB LED і Buzzer на вихідний режим

pinMode(RED_PIN, OUTPUT);

pinMode(GREEN_PIN, OUTPUT);

pinMode(BLUE_PIN, OUTPUT);

pinMode(BUZZER_PIN, OUTPUT);

Ініціалізуємо RFID модуль

SPI.begin();

rfid.PCD_Init();

#define RFID_SS_PIN 10 RFID_RST_PIN 9

Код налаштування клавіатури зображено на рис. 4.7. Створюємо об'єкт клавіатурі підключити до виводам плати. Створюємо масив на правильне значення клавіш.

```
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 4.7 – Налаштування клавіатури

Створюємо об’єкт класа котрий схороняє пароль. SafeState зберігає секретний код в EEPROM.

SafeState safeState;

Водна точка програми починається з цього налаштування, код зображен на рис. 4.8. Швидкість передачі даних у системі 115200 байт в секунду. Коли почнеться ця команда замок буде в окритому положенні. Також тут вводим ініціалізацію параметрів екрана з бібліотеки, котрі зображені на рис 4.9. Після операцій йде метод котрий показує стартовий екра н коли вмикаємо код , рисунок 4.10 та 4.11.

```
void setup() {
  lcd.begin(16, 2);
  init_icons(lcd);

  lockServo.attach(SERVO_PIN);
  pinMode(BUZZER_PIN, OUTPUT);
  pinMode(RGB_RED_PIN, OUTPUT);
  pinMode(RGB_GREEN_PIN, OUTPUT);
  pinMode(RGB_BLUE_PIN, OUTPUT);

  SPI.begin();
  rfid.PCD_Init();

  Serial.begin(115200);
  unlock();

  showStartupMessage();
}
```

Рисунок 4.8– Водна точка програми

```
void init_icons(LiquidCrystal &lcd) {
  byte icon[8];
  memcpy_P(icon, iconLocked, sizeof(icon));
  lcd.createChar(ICON_LOCKED_CHAR, icon);
  memcpy_P(icon, iconUnlocked, sizeof(icon));
  lcd.createChar(ICON_UNLOCKED_CHAR, icon);
}
```

Рисунок 4.9– Ініціалізація параметрів екрана

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

Рисунок 4.10 – Код стартового екрана.

Рисунок 4.11 – Вигляд стартового екрана на дисплеї

4.9 Головний цикл

Після операцій стартового екрана код іде в головний цикл – «loop». І там по колу ходить між логікою відкритого замку та закритого. В першому запуску воно іде у логіку з відкритим замком.

```
void loop() {
  if (safeState.locked()) {
    checkRFID(); - перевіряє RFID картку
    safeLockedLogic(); - закрита логіка
  } else {
    safeUnlockedLogic(); }} - відкрита логіка
```

4.9.1 Відкрита логіка

У відкритої логіки код задає пароль та перевіряє його. Змінна відповідає за те що б в майбутньому можливо було перевести замок у закрите положення. Починається з очистки дисплея після стартового екрана та задає новий, на котрому зображуєтеся іконки замка та напис рис. 4.12. Треба натиснути «#» що б перейти далі і відкрився доступ для написання, або першого коду, треба ввести 4 цифри від 0 до 9, після цього треба буде підтвердити цей пароль, повторно його ввести, рисунок 4.13. Якщо перевірка коду не пройде, то-б-то, якщо при другому вводі коду допустити помилку і пароль буде відрізнятися

від першого введеного, водно не пройде перевірку і напише про помилку спрацює світлодіод червоного кольору та спрацює spiker котрий повідоміть про помилку, рисунок 4.14, та цикл почнеться з початку, з відкритої логіки. Якщо пароль пройде перевірку (введений код другий раз збігся з першим) то замок зачиниться загориться світлодіод та динамік, рисунок 4.15. Ця змінна завершається та повертається до циклу, воно перевіряє та бачить що замок зачинено і цикл переходить в закриту логіку.

Рисунок 4.12 – Відкрита логіка

Рисунок 4.13 – Ведення нового пароля, та підтвердження

Рисунок 4.14 – Помилка прі вводі пароля

Рисунок 4.15 – Замок пройшов перевірку та зачинився

4.9.2 Закрита логіка

Закрита логіка починається з того що код визначає що замок зачинено. На дисплеї показує що замок зачинено і треба ввести пароль котрий зберігся. Якщо пароль неправильний то на дисплеї появиться повідомлення що пароль не підходе та динамік спрацює та загориться червоний колір, треба буде зачекати 10 с в цей час код іде у цикл та знов виходить у закриту логіку де треба ввести пароль. Рисунок 4.16. При введенні вірного пароля замок відкривається, пролунає динамік іншим звуком та загориться зелений колір , рисунок 4.17, та код переходить у цикл і йде у відкриту логіку де вже пропанує вибір як закрити замок, рисунок 4.18:

– Натиснути «#» для того що б закрити замок використовуючи минулий пароль;

– Натиснути «A» для того що б задати новий пароль і закрити замок.

Натиснувши «#» замок закривається переходи в цикл де йде відкриту логіку, код бачить що замок закритий – йде в закриту логіку де треба ввести

пароль. При виборі нового треба задати пароль та ввести його ще раз, якщо перевірка нового пароля не пройде то воно пройде цикл и повернеться до вибору як закрити замок, рисунок 4.18. При проходження перевірки збереже нову змінну і замок сейфа зачиниться із новим паролям, а старий забуде.

Рисунок 4.16 – Помилка при введені пароля

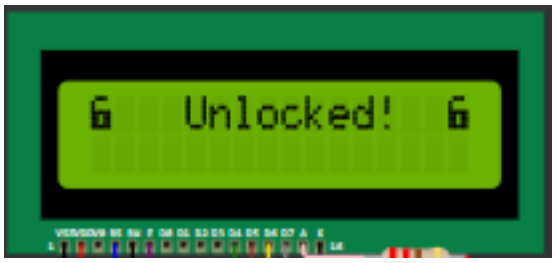

Рисунок 4.17 – Замок відкрито

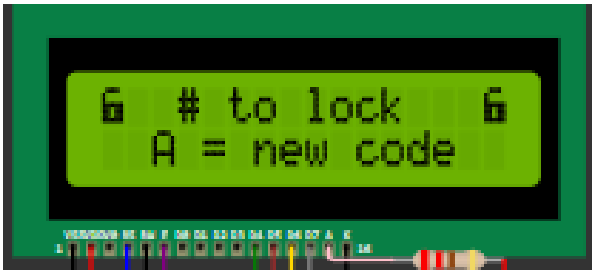

Рисунок 4.18 – Обрати як закрити замок

Початковий етап роботи системи включає додавання необхідних бібліотек для дисплея, клавіатури, серводвигуна та інших компонентів, відповідальних за збереження пароля та відображення іконок. Налаштовується серводвигун для визначення кутів повороту в закритому та відкритому

положеннях, а також клавіатура для введення коду. Створюється об'єкт для збереження пароля в EEPROM.

Основна програма починається з налаштування серводвигуна та дисплея. Після стартових налаштувань система переходить у цикл роботи, в якому постійно чергуються логіки відкритого та закритого замка. При першому запуску логіка відкритого замка дозволяє користувачу задати новий пароль та перевірити його правильність. Якщо перевірка успішна, замок зачиняється, і система переходить до логіки закритого замка.

У логіці закритого замка користувач повинен ввести збережений пароль для відкриття замка. Якщо пароль введено неправильно, система вимагає повторної спроби після затримки. При правильному введенні пароля замок відкривається, і користувач може обрати між закриттям замка зі старим паролем або введенням нового пароля.

Цей алгоритм забезпечує надійний контроль доступу до замка, дозволяючи змінювати паролі та зберігати безпеку системи шляхом циклічної перевірки правильності введених даних.

4.9.3 Відкриття за допомогою RFID

Модуль – як альтернативний спосіб відкрити сейф. У моєму прикладі буде використовуватися лише одна картка-брелок котра йде у набор з модулем. Як що потрібно буде зробити дублікат треба перезаписати код з оригіналу на іншу карту. Також можна зробити від самого початку декілька карт котрі можуть відкрити сейф, та зберегти у EEROM їх код. Для того що б все працювало потрібно знати UID карти. Його можливо узнати піднісши карту до считувача.

Коли замок у закритому стані піднесемо ключ-картку до зони де знаходиться модуль, та воно перевіряє ключ-картку. Як що номер картки співпадає з номером написаним у коді то загориться зелений світлодіод,

пролунає динамік та сейф відчиниться. На екрані з’явиться напис про відкриття, рисунок 4.17. Та цикл піде далі у закриту логіку.

Якщо перевірку не пройдено, номер картки не відповідає, на дисплей екрані з’явиться напис про помилку, загориться червоний колір та спрацює динамік, рисунок 4.16.

ВИСНОВКИ

У ході виконання дипломної роботи було розглянуто й проаналізовано різні аспекти розробки цифрових замків на базі платформи Arduino. Проведено огляд існуючих технологій цифрових замків, що включає використання клавіатур, RFID-технологій та інших засобів ідентифікації.

Розроблено програмне забезпечення для цифрового замка, яке включає функції управління серводвигуном, обробки введення з клавіатури, відображення інформації на LCD-дисплеї та роботу з RFID-картами. Було реалізовано функцію додавання нових карток для можливості розширення списку користувачів, що мають доступ до замка.

У процесі роботи було приділено увагу безпеці зберігання паролів та ідентифікаційних даних. Використання EEPROM для зберігання секретних кодів забезпечує надійність та стійкість системи до втрати живлення. Крім того, розроблена система включає захисні механізми від спроби підбору пароля, такі як затримка у випадку неправильного введення.

Практична частина роботи продемонструвала ефективність створеної системи. Цифровий замок на базі Arduino успішно виконує поставлені завдання, забезпечуючи надійний захист приміщення та зручність використання. Отримані результати підтверджують доцільність використання платформи Arduino для розробки подібних систем, завдяки її гнучкості, доступності та широким можливостям для інтеграції з іншими компонентами.

Ця дипломна робота демонструє можливості сучасних цифрових замків, їх переваги та області застосування. Розроблене рішення може бути використано як основа для подальших досліджень та вдосконалень у сфері безпеки та автоматизації, що відкриває нові перспективи для створення ще більш надійних та зручних систем захисту.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЯ

1. Arduino Uno R3. [Електронний ресурс]. https://www.robostore.com.ua/ua/otladochnaia-plata-arduino-uno-rev3/
2. Arduini Nano [Електронний ресурс] https://store.arduino.cc/products/arduino-nano?_gl=1*emplfe*_gcl_au*NjE0MTk3NjM1LjE3MTYzNjY4NTc.*FPAU*NjE0MTk3NjM1LjE3MTYzNjY4NTc.*_ga*MTc3NDIyNTM3NC4xNzE2MzY2ODU1*_ga_NEXN8H46L5*MTcxNjM3Mzc2OC4zLjEuMTcxNjM3NDkzMi4wLjAuMzA0MjY2MTEz*_fplc*VzQ2Y3Z1ZEQ1Z1pXN0xxaWZ3NnVPRExZbnBmcUZWY3lOczhFTEF2SnIyckZneHUxRiUyQnJ4bTRZSW53UjFnVTE4QzhuQUR5dndHdUIzZ01WN3FRd1VRVUg2SXVBbE5VbEZxRXZRYWExJTJGcVR4VktnWlU3VmhnWDY3SzJpSkhaQSUzRCUzRA
3. Arduino Mega [Електронний ресурс] https://store.arduino.cc/products/arduino-mega-2560-rev3?_gl=1*151oewy*_gcl_au*NjE0MTk3NjM1LjE3MTYzNjY4NTc.*FPAU*NjE0MTk3NjM1LjE3MTYzNjY4NTc.*_ga*MTc3NDIyNTM3NC4xNzE2MzY2ODU1*_ga_NEXN8H46L5*MTcxNjM3Mzc2OC4zLjEuMTcxNjM3NTAxMS4wLjAuMzA0MjY2MTEz*_fplc*VzQ2Y3Z1ZEQ1Z1pXN0xxaWZ3NnVPRExZbnBmcUZWY3lOczhFTEF2SnIyckZneHUxRiUyQnJ4bTRZSW53UjFnVTE4QzhuQUR5dndHdUIzZ01WN3FRd1VRVUg2SXVBbE5VbEZxRXZRYWExJTJGcVR4VktnWlU3VmhnWDY3SzJpSkhaQSUzRCUzRA..
4. Шарфельд Т. Системы RFID низкой стоимости / Пер. с англ. / Под ред. Корнеева С.В. М.: 2006.- С.9-11. [Scharfeld T.A. An Analysis of the Fundamental Constrains on Low Cost Passive Radio Frequency Identification System Design, 2001].

5. Arduini Due [Електронний ресурс] https://miniboard.com.ua/mcu/744-robotdyn-due.html
6. Arduino Sketches [Електронний ресурс] https://docs.arduino.cc/learn/programming/sketches/
7. https://doc.arduino.ua/ru/guide/Environment
8. https://doc.arduino.ua/ru/about/
9. RFID RC522 [Електронний ресурс] https://arduino.ua/ru/prod649-rfid-modyl-rc522-s-kartochkoi-dostypa-dlya-arduino
10. Аппаратная платформа Arduino [Электронный ресурс]. https://arduino.ua/
11. Офіційний сайт розробника цифрових замків August [Электронный ресурс] https://august.com/
12. Офіційний сайт розробника цифрових замків Shelage [Электронный ресурс]. https://www.schlage.com/en/home.html
13. Офіційний сайт розробника цифрових замків Yale [Электронный ресурс] https://www.yalehome.com/global/en/trusted-innovation/our-product-portfolio/smart-locks
14. Офіційний сайт розробника цифрових замків Kwikset [Электронный ресурс]. https://www.kwikset.com/
15. Офіційний сайт розробника цифрових замків Samsung [Электронный ресурс] https://www.samsungdigitallife.com/