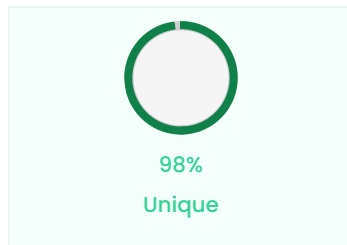
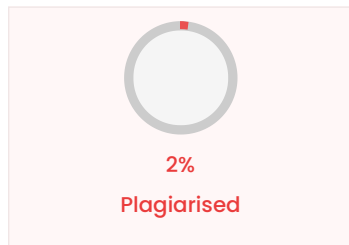


## Plagiarism Scan Report



## Words Statistics

Words	16299
Characters	131303

Exclude URL: None

## Content Checked For Plagiarism

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ПрАТ «ПРИВАТНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД «ЗАПОРІЗЬКИЙ ЕКОНОМІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ» Кафедра комп'ютерної інженерії ДО ЗАХИСТУ ДОПУЩЕНА Зав \_\_\_\_\_ д.т.н., проф. Переверзев А.В. БАКАЛАВРСЬКА ДИПЛОМНА РОБОТА КЛІЄНТ-СЕРВЕРНА КОНТРОЛЮ УПРАВЛІННЯ ДОСТУПОМ СКЛАДСЬКИХ ПРИМІЩЕНЬ Виконав ст. гр. КІ-128 \_\_\_\_\_ А Керівник доцент \_\_\_\_\_ О.А. Жеребцов Запоріжжя 2022 ПрАТ «ПРИВАТНИЙ ВИЩИЙ НАВЧАЛЬНИ «ЗАПОРІЗЬКИЙ ІНСТИТУТ ЕКОНОМІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ» Кафедра комп'ютерної інженерії ЗАІ Зав. кафедрою д.т.н., професор А.В. Переверзев \_\_\_\_\_ 17.01.2022 р. З А В Д А Н Н Я НА БАКА ДИПЛОМНУ РОБОТУ Студенту гр. КІ-128, спеціальності 123 «Комп'ютерна інженерія» Рассохіну Арсенію Костя- Тема: Клієнт-серверна система контролю управління доступом складських приміщень затверджена наказом по інс' 50 від 15 січня 2022 року. 2. Термін здачі студентом закінченої роботи: 18 червня 2022 року. 3. Перелік питань, що розробці: 1. Дослідити предметну область систем контролю управління доступом. 2. Здійснити огляд існуючих систе та проаналізувати принцип їх структурної побудови. 3. Розглянути можливість запровадження клієнт-серверн контролю управління доступом складських приміщень. 4. Здійснити вибір програмно-апаратного комплексу дл проекту. 5. Розробити апаратне рішення електронного вузла виконавчого пристрою, спираючись на обрану базу ком Розробити програмний комплекс керування виконавчим пристроєм. 7. Розробити програмний алгоритм прийому, керування розгорнутої серверної частини. 8. Розробити програмно- апаратне рішення пристрою внесенн ідентифікаційних даних до бази даних. 9. Розробити графічну панель керування системою контролю управління д Провести контрольне налаштування та випробування системи. 11. Оформити результати роботи у вигляді звіту. , завдання: 17 січня 2022 р. Керівник бакалаврської роботи \_\_\_\_\_ О.А. Жеребцов Завдання отримав дс \_\_\_\_\_ А.К. Рассохін РЕФЕРАТ Бакалаврська дипломна робота містить: 123 сторінки, 90 рисунків, 14 та додатків, 43 першоджерела. Мета роботи – створення клієнт-серверної системи контролю управління доступом приміщень, на базі взаємодії периферійних пристроїв з базою даних за допомогою сокет з'єднань. Предмет дос дослідження інформаційних процесів взаємодії у системі з клієнт-серверною архітектурою. Об'єкт дос автоматизована системи контролю та управління доступом. У першому розділі проведено дослідження існую контролю доступу. Розглянуто актуальні методи побудови систем контролю та управління доступом у по промислових випадках застосування. Обґрунтовано рішення розробити і запровадити клієнт-серверну систему к управління доступом, основою якої є, керуючий периферійними пристроями, одно кристальний комп'ютер та розгорнутий за допомогою робочої станції , які об'єднані в локальну мережу. У другому розділі описано та обґрунт програмних та апаратних інструментів розробки для реалізації клієнт-серверного програмно-апаратного рішення. розділі розглянутий процес розробки і проектування апаратної та програмної частини проекту системи контролк також, розглянуто процес тестування та налагодження системи. Результати роботи були оприлюднені на всеукраїнській конференції «Актуальні проблеми математики та інформатики». СКУД, СЕРВЕР, КЛІЄНТ, TCP-3 СОКЕТ, ОБМІН ДАНИМИ, МІКРОКОНТРОЛЕР, STM, MySQL,.NET ЗМІСТ СПИСОК УМОВНИХ ПОЗНАЧЕНЬ 6 РОЗД

СИСТЕМ КОНТРОЛЮ ТА УПРАВЛІННЯМ ДОСТУПОМ 11 1.1 Класифікація СКУД 12 1.1.1 Класифікація СКУД за кількістю пропусків 14 1.1.2 Класифікація СКУД за різновидом пристрою зчитування 14 1.1.3 Класифікація СКУД за різновидом пристрою 15 1.1.4 Класифікація СКУД за рівнем фізичного захисту 18 1.1.5 Класифікація СКУД за методом мережевої ідентифікації 20 1.2 Огляд технології радіочастотної ідентифікації 20 1.3 Огляд популярної СКУД 24 1.4 Створення принципової моделі 27 1.5 Висновки за розділом 29 РОЗДІЛ 2 30 ВИБІР ПРОГРАМНО-АПАРАТНОГО КОМПЛЕКСУ ПРОЕКТУ 30 2.1 Обґрунтування базових вузлів для створення апаратного комплексу проекту 30 2.1.1 Контролер виконавчого пристрою 30 2.1.2 Контролер радіочастотної ідентифікації 36 2.1.3 Чіп зберігання ідентифікаційних даних 39 2.1.4 Контролер з'єднання 39 2.1.5 Контролер пристрою внесення даних 47 2.2 Вибір та обґрунтування базових рішень для створення програмного комплексу проекту 51 2.2.1 Програмне забезпечення для програмування блоку виконавчого пристрою 51 2.2.2 Програмне забезпечення для програмування блоку серверної частини 63 2.3 Висновки за розділом 68 РОЗДІЛ 3 ПРАКТИЧНА РОЗРОБКА ТА ВИПРОБУВАННЯ 69 3.1 Проектування СКУД 69 3.2 Проектування виконавчого пристрою 78 3.3 Проектування пристрою внесення даних 78 3.4 Програмування комплексу виконавчого пристрою 80 3.5 Програмування бази даних 91 3.6 Програмування пристрою внесення даних 101 3.7 Програмування додатку роботи співробітників 103 3.8 Тестування та налагодження СКУД 110 3.9 Висновки за розділом 117 ВИСНОВКИ 118 ПЕРЕЛІК АБРАКАДАБРАС 119 Додаток А. Принципові електричні схеми 124 Специфікація бази електронних компонентів 126 Додаток Б. Моделі плат 127 Додаток В. Вихідний код програми мікроконтролера 129 Додаток Г. Вихідний код серверу обробки даних 135 Додаток Д. Вихідний код клієнтської частини 138 Файл MainForm.cs 138 Файл FormDesigner.cs 151 Файл DataBank.cs 151 Файл FormContact.cs 163 Файл FormContactDesigner.cs 164 ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ Слово/Словосполучення Скорочення А АЛП АЦП Алгоритмічно-логічний пристрій АЦП Аналогово-цифровий перетворювач Д ДП Друкована плата О ОЗП ОС Оперативно-запам'ятовуючий пристрій Операційна система ПІБ Постійний-Запам'ятовуючий Пристрій Прізвище Ім'я По-батькові С СКУД СУБД Система Контролю та Доступом Система Управління Базами Даних Ц ЦАП Цифро-Аналоговий Перетворювач Е ЕОП Електронно-Обчислювальний Пристрій А ARP Address Resolution Protocol С CAN CMSIS CRC Control Area Network Common Microcontroller Interface Standard Cyclic Redundancy Check D DMA Direct Memory Access E EEPROM Electrically Programmable Read-Only Memory F FIFO First In First Out G GPIO GUI General Purpose Input-Output Graphic Interface H HAL Hardware Abstraction Layer I IC ICMP IDE IGMP IOT I/O IP Integrated Circuit Internet Control Protocol Integrated Development Environment Internet Group Management Protocol Internet Of Things Intranet Internet Protocol L LINQ LoRa Language Integrated Query Long Range Protocol M MCU MISO MOSFET Microcontroller Unit Master In Slave Out Metal-Oxid-Semiconductor Field-Effect-Transistor Master Out Microprocessor Unit N NFC NXP Near Field Communication Nexpira O OSI The Open Systems Interconnector PWM Physical Layer Point Of Sale Phase Width Modulation R RAM RFID RNG RX Random Access Memory Random Frequency Identification Random Ranking Generator Receive S SMD SPI SRAM SSL Surface-Mounted Device Peripheral Protocol Static Random Access Memory Secure Sockets Layer T TCP TX Transmission Control Protocol Transmit U UDP UID USART USB User Datagram Protocol Unique Identifier Universal Asynchronous Transmitter Universal Serial Bus W WI-FI WOL Wireless Fidelity Wake On LAN 99 ВСТУП Розвиток науки та проектування та будівництва комп'ютерних систем, знайшов свій шлях у потребі розробки методів взаємодії між різними системами різного рівня інженерної складності. Так з'явилися електронно-обчислювальні прилади, що побудовані на кристалах та, згідно завдання, які займають мало місця й потребують менше ресурсів та складності під час виготовлення мікроконтролерів. Вони зайняли своє почесне місце в сучасних інженерних розробках та в популярній на сьогоднішній день вбудованих системах (англ. Embedded System). Вбудована система – спеціалізована мікропроцесорна система управління, контролю та моніторингу, концепція розробки якої полягає в тому, що така система буде працювати вбудованою безпосередньо у пристрій, яким вона керує. Вбудовані системи використовуються в якості як окремих електронних вузлів або систем, так і в умовах ланцюгового з'єднання, з використанням різних видів та методів Основними напрямками використання структур вбудованого типу є: засоби автоматичного регулювання та контролю технологічними процесами, наприклад авіоніка, контроль доступу, верстати з ЧПУ (Числовим Програмним Контролем) банкомати, платіжні термінали, телекомунікаційне обладнання та величезна кількість інших реалізацій будь-якої належності. Завдяки великій кількості переважуючих властивостей мікроконтролерної невеликогабаритної техніки зайняла суттєву нішу в програмі розробок електронних інженерів в усьому світі. РОЗДІЛ 1 ОГЛЯД СИСТЕМ КОНТРОЛЮ ТА УПРАВЛІННЯМ ДОСТУПОМ За визначенням вікіпедії СКУД це комплекс технічних та програмних засобів безпеки, що регулювання входу / виходу та переміщень людей чи транспортних об'єктів на територіях, які знаходяться під об'єктом адміністративного моніторингу та попереджень несанкціонованого проникнення [1]. На даний час вже існує безліч систем контролю, що давно вийшли за межі їх базової функції – надання доступом до чогось або до когось за наявності

Світовий ринок подібних систем здатен задовільнити будь-якого прискіпливого замовника, та навіть більше, де різноманітні цінові та якісні пропозиції, починаючи від бюджетних пристроїв локального використання, де не потрібний фізичний та програмний рівень захисту до доволі недешевих клієнт-серверних систем, що спроектовані б охороняти фінансові заклади або об'єкти стратегічного значення. Ідентифікатором особи можуть слугувати як біометричні такі як: райдужка ока, відбиток пальців, форма лица, форма долоні та голос, так і більш загальноприйнятні, ті, які зарекомендувалися – дані електронного носія, який комунікує за допомогою радіо-сигналу або його емуляції за спеціального або загально-прийнятого алгоритмів. Інколи, до вище запропонованих методів додаються інтернет можна керувати доступом об'єкту, який треба охороняти. Найважливішим та базовим завданням на початку прототипу системи контролю доступом є вибір програмної та апаратної структури, що заздалегідь повинна бути технічному завданні проекту. Насамперед, треба чітко оглянути існуючі аналоги подібних систем та віднести створюється, до окремої категорії за її характеристиками та технічними можливостями. Для цього має бути визначено від СКУД, а саме: місце її використання, кількість користувачів, фізична та програмна стійкість системи за умов її використання. Створення коректного плану полегшить роботу під час проектування та найголовніше – покращить результат вихідного прототипу, виключаючи велику кількість різноманітних проблем під час проектування за наявності якісного аналізу початку.

### 1.1 Класифікація СКУД

Системи контролю управління доступом поділяються на два базових різновиди за топологією, від яких надалі відбувається розширення властивостей: автономні та мережеві.

- 1) Автономні системи – це системи, що мають здатність контролювати та вести запис безпосередньо за допомогою виконуючого пристрою, що перебуває біля зчитувача (рис. 1.1). До переваг системи відноситься її помітна простота та невелика вартість, те, що потребує апаратних рішень, що можуть вплинути на кінцеву складність, таких як: серверна частина, система управління програмних реалізацій серверної частини. Вона підходить до використання на невеликих об'єктах та там, де відсутній контрольючий охороною об'єктом не треба постійно звертатись до виконавчого пристрою віддалено, задля корекції роботи, слідкування за подіями та видаленням або додаванням нових ідентифікаційних орієнтирів. Негативні характеристики подібних систем є, але вони доволі суб'єктивні, адже відображають окремі класи приладів та не зовсім коректні до порівняння із мережевими клієнт-серверними системами. Тим не менш: простота часто впливає на функціональні особливості автономні системи не завжди можуть задовільнити потребам користувача. Також, подібні пристрої не завжди мають якість виконання як апаратної, так і програмної реалізації, в більшості не мають криптографічного захисту. Рівень безпеки Позначення вузла клієнт
- 2) Мережеві системи – це системи з більш розширеним та наповненим різноманітними можливостями функціоналом. Такі системи створені аби мати можливість вести повний контроль та керувати доступом за допомогою комп'ютеру з різних, нефіксованих місць, таких як: відділ кадрів, офіс та навіть маніпулювати з телефонними системами використовуються на великих об'єктах та підприємствах, де основна вимога – вести керування з єдиного центру управління, та зберігати всі дані саме на ньому. Подібна система за своїми позитивними рисами виграє у систем типу, але користувачу треба заздалегідь мати на увазі більшу вартість такого рішення. Рисунок 1.2 – Позначення вузла сервер

Окрім цього базового поділення також існує декілька класифікацій СКУД, які необхідно розглянути.

### Класифікація СКУД за кількістю пунктів пропуску

За кількістю пунктів пропуску системи контролю керування доступом поділяються на три різновиди. Кожен з них має переваги та недоліки один перед одним, класифікація залежить від потреби пункту, щодо потрібної кількості пунктів керування фізичного доступом на ньому. Поділяються на:

- 1) До 16 пунктів пропуску – автономні і мережеві системи контролю доступом. Характеризується невеликою та, у більшості, нерозвиненою системою мережевих комунікацій. Об'єкти використання: приватні будинки, невеликі навчальні заклади, невеликі підприємства, та інші.
- 2) Від 16 до 64 пунктів пропуску – мережеві системи контролю доступом. Характеризуються більш високим рівнем мережею комунікаційних з'єднань. Об'єкти використання: невеликі або великі навчальні заклади, невеликі підприємства, та інші.
- 3) Від 64 пунктів пропуску – мережеві системи контролю доступом. Нічим не відрізняється від попереднього рівня, окрім більш великої мережі комунікаційних з'єднань. Об'єкти використання: великі навчальні заклади, великі підприємства, та інші об'ємні структурні підрозділи.

### 1.1.2 Класифікація СКУД за різновидом пристрою зчитування

За різновидом пристрою зчитування системи контролю керування доступом поділяються на три основних різновиди:

- 1) Біометричні системи – це системи, що виконують ідентифікацію людини шляхом зчитування його індивідуальних фізичних рис. Існують такі методи ідентифікації людини: 3-х вимірний фотографія обличчя, фотографія та ідентифікація райдужки ока, зчитування відбитків пальців та інші. Переваги: відсутність потреби мати при собі пристрій-носіє ідентифікаційної інформації, що унеможливорює процес підготовки ідентифікатора. Недоліки: менш швидкий процес ідентифікації, більша кінцева вартість системи, через використання складних оптичних або резистивних приладів зчитування.
- 2) Цифрова (RFID) – системи – це системи, що ідентифікують людину шляхом порівняння цифрової інформації, яка знаходиться на електронному носії з тією інформацією, яка знаходиться в базі зберігання даних користувачів електронних носіїв. Передача даних з мітки створюється методом радіосигналів (або емуляції радіосигналів) від мітки до зчитувача. Переваги: невелика вартість носіїв інформації

зчитування загалом, особливо в порівнянні з біометричною, гарні криптографічні властивості зберігання інформ швидкість зчитування ідентифікатора. Недоліки: потреба мати картку-носіє завжди при собі, теоретична несанкціонованого зчитування інформації з носія. 3) Кодова- найбільш проста та застаріла система ідентифі сьогоденні майже не використовується окремо від двох перших. Ідентифікатором є код-пароль, що користувач ма пристрої вводу. Переваги: мала вартість системи, відсутність потреби мати ідентифікаційний носій. Недоліки: п користувача пам'ятати унікальний код доступом, ускладнення процесу ідентифікації окремих осіб та ведення та робочого часу, погана стійкість до зламування.

### 1.1.3 Класифікація СКУД за різновидом виконавчого пристрою

пристрій системи контролю- це фізичний механізм, що призначений обмежувати доступ людини або предмету на що пристрій охороняє. Залежно від об'єкту та від рівня його захищеності, де механізм буде працювати систем керування доступом поділяються на 3 види: 1) Турнікети - найбільш популярний тип захисного механізму пункт підприємств, метрополітену та інших об'єктів, де відбувається великий потік пересування людей, що контролю найбільш продуктивно працюють в тих місцях, де потрібна швидкість перевірки та пересування. Турнікетні систем поділяються на 3 основні типи: \* Полу ростові триподні турнікети- турнікетні системи, де обмежувачем пристроєм ригелі, що спрямовані один відносно іншого під кутом 120 градусів). кінематика приладу сконструйована таким тільки одна людина могла пройти скрізь ного за один такт пропускання. Частіше за все, подібні турнікет зустрічаються на підприємствах, в офісних центрах та навчальних закладах. Рисунок 1.3 – Зображення пол турнікетів \* Повно ростові триподні турнікети- турнікетні системи, що за методом дії схожі із полу ростовим тур мають більш гарні показники безпеки, адже виключають можливість проникнення людини зверху, якщо та намагал зробити. Рисунок 1.4 – Зображення повно ростових турнікетів \* Стулчасті системи- зазвичай зустрічаються пропуску метрополітену, мають найгарніший показник пропускну здатності, адже механізм дуже простий до е Рисунок 1.5 – Зображення стулчастих систем

### 2) Ригельні електромеханічні замки- електромеханічна система, щ відрізняється від звичайного ригельного замка в побутових квартирах, крім того, що обладнані електронно контролю та без ключовим доступом. Мають гарні показники захисту, адже зазвичай вмонтовані в металеві двері

Поділяються на 2 основні типи: \* Механічні - приводом прямої дії є механічна система в невеликим електроде змушує ригель рухатись; \* Соленоїдні- приводом прямої дії є електричний магніт, що змушує ригель рухатись. Рі

### Зображення замку ригельного типу

### 1) Електромагнітні замки- це система, що складається з електричного керується електронною частиною системи контролю керування доступом. Гарно спроектовані системи з використовують електромагнітний метод захисту дуже добре протидіють злому, адже знаходяться з іншої від з. сторони двері або ставнів. Як і ригельні електромеханічні замки, електромагнітні неефективні у випадках, коли великий потік людей на пункті пропуску. Рисунок 1.7 – Зображення замку електромагнітного типу

### 1.1.4 Класифікація рівнем фізичного захисту

За рівнем фізичного захисту системи контролю керування доступом поділяються на 2 ос

### 1) Перший рівень захисту за побудовою конструкції

теоретично дозволяє несанкційно проникнути людині, що не доступом, тому зазвичай на практиці супроводжується контролерами або охоронцями на контрольних пунктах п цього рівня належать полу ростові турнікетні системи та стулчасті системи. 2) Другий рівень механічк несанкційному проникненню людини, адже за побудовою конструкції не залишає незахищеного місця. Охоронні си рівня не потребують обов'язкової присутності охоронця на пункті пропуску. До цього рівня належать: повно ростові системи, ригельні електромеханічні замки та електромагнітні замки.

### 1.1.5 Класифікація СКУД за методом мережев

Спосіб комутації клієнт-серверних систем контролю управління доступом визначає стійкість системи до спроб г злому та до електромагнітної або електричної стійкості. Метод підключення та його реалізація визначає ек параметри системи загалом. На сьогоднішній день існує 2 методи: 1) Безпроводний- метод, що полягає у повній електричних кабелів зв'язку між виконавчим пристроєм на пункті пропуску та клієнтом, що бажає здійснити зв'язок

Загалом, метод використовує технології безпроводного з'єднання: IEEE 802.11 Wi-Fi, новішу LoRa та інші. Основи повна відсутність кабелів дозволяє уникнути складної процедури протягнення кабельної мережі від одного до самим посилюючи безпеку фізичного впливу на процес передачі. Основні недоліки: безпроводні системи, а, точніше підключення безпроводних систем у деяких випадках поступаються мережевому кабелю надійністю каналу зв'язку. створення електромагнітних перешкод може сприяти неповній передачі даних а інколи і повному зриву. 2) Кабель що полягає в підключенні виконавчого пристрою на пункті пропуску до клієнта, або вузла розподілення а по методом протягування кабелю. Загалом, професійні системи контролю використовують стандарти фізичного рівня: F 485 та Ethernet. Основні переваги: надійний зв'язок та забезпечення повної передачі на прийому даних, с електромагнітних утручань. Основні недоліки: потреба прокладання довгих кабелів, що треба ховати задля її фізичного впливу, у деяких випадках - потреба у вузла комутації різних каналів.

### 1.2 Огляд технології рад

ідентифікації

Більш щільно треба оглянути технологію радіочастотної ідентифікації, яким чином вона працює, де

вона використовується, які її переваги та недоліки та чому саме її було обрано використовувати в проекті системи управління доступом складських приміщень. RFID (Radio frequency identification) або радіо-частотна ідентифікація супроводжує історію ще з 1937 року, коли воєнні авіаційні системи навігації вперше використали спосіб розпізнавання чужий». Вона активно використовувалася під час другої світової війни. Перша демонстрація сучасних RFID-модулів активного та пасивного типу датується 1973 роком. Портативна система працювала на частоті 915 МГц та використовувала 12-розрядні мітки. Аж у 1999 році була створена «глобальна система стандартів» для мікросхем RFID. У сьогоденні радіочастотної ідентифікації використовуються в усіх сферах життя, починаючи від автоматизації виробництва до допомоги технології спрощуються та прискорюються процеси всіх видів виробництва, закінчуючи оплатою в магазинах за допомогою NFC на POS-терміналі та, авжеж, системами контролю та управління доступом (СКУД). Технічних настільки багато, що, на жаль, вмістити їх в одну тему неможливо, тож потрібно вказати хоча б основні її види відношення до ідентифікації людини для подальшого опрацювання процесу за допомогою отриманої інформації системи контролю та управління доступом будь-яких об'єктів використовують дві основні ідентифікаційні технології: EM Marine та Mifare (рисунки 1.8). Рисунок 1.8 – Логотипи технологій 1) EM Marine – найбільш популярна технологія зберігання інформації за допомогою «смарт-карт». Формат був розроблений швейцарською компанією EM Microelectronics. Технологія була розроблена перш за Mifare, та стала популярною у використанні з домофонними системами побутових будинків в Україні. Можна зустріти дві основних мікросхеми, на яких базуються прилади зберігання, вони відповідають основним стандартам частот 125 КГц: TK4100 та EM4100. Чіп TK4100 не є оригінальним продуктом, використовується великою кількістю сторонніх компаній за однією технологією, встановлений в більшість карт, що відповідає EM4100 є оригінальним продуктом компанії EM Microelectronics та не є дуже розповсюдженим через значні переваги: невелика ціна, тому велика популярність та розповсюдженість. Через це не є проблемою знайти прилади систем контролю, які будуть сумісні в роботі з мікросхемами TK4100 та EM4100. Недоліки: відсутність функцій перезапису та зчитування. Для цього не знадобиться складних на дорогих приладів для створення операції, номер ідентифікації, котрий інколи написаний на передній частині «тіла» карти або електронного ключа, відсутність анти колізії (механізму запобігання зіткненню двох або більше ідентифікаторів), що при наявності їх поблизу одна з карт при зчитуванні буде викликати помилку. 2) Mifare – торговельна марка розповсюдженої в Україні технології без дротів зчитування за допомогою «смарт-карт» або електронних ключів що належить нідерландській компанії напівпровідникової електроніки Nxp (NXP) Semiconductors. Окрім NXP, ідентифікатори за ліцензією випускає компанія Infineon, тільки ті карти, на яких є позначки цих виробників можна вважати оригінальними, тільки вони є гарантією унікальності номерів ідентифікації. Однак, як і в попередньому варіанті з EM Marine, Mifare виробляється й іншими компаніями. Технологія отримала велику популярність у використанні із різноманітними СКУД, від малих підсистем контролю стратегічно-важливих об'єктів, та завдяки своїм перевагам працює і надалі. Мікросхеми відповідають основним стандартам частот 13.56 МГц та мають декілька серійних різновидів, від яких залежить рівень криптографічного захисту: Mifare Ultralight, Mifare Classic, Mifare Plus, Mifare DESfire, Mifare SmartMX. Переваги: присутність функцій перезапису та зчитування, велика внутрішня пам'ять пристрою, завдяки якій можна вносити записи різної інформації та ключів доступу, механізм анти колізії, що унеможливує «зіткнення» двох або більшої кількості ідентифікаторів створі зчитувача, унікальні номери кожної окремої мікросхеми (UID), більша швидкість обміну інформацією між транспондером та блоком зчитувача. Недоліки: більша вартість «смарт-карт» або електронних ключів в порівнянні з EM Marine, процес запису/перезапису та зчитування, що інколи важливо. Висновок: спираючись на публічний аналіз використання особистий аналіз документації мікросхем радіочастотної ідентифікації компаній EM Marine та Mifare, що використовуються в системах контролю та управління доступом, були порівняні властивості двох технологій. Як сучасну модель без дротів зв'язку для використання в СКУД складських приміщень було обрано технологію Mifare від NXP, через вищу функціональність та якіснішу реалізацію захисту. Нижче приведено таблицю 1.1 порівняння характеристик транспондерів EM Marine та Mifare (Classic) версії. Таблиця 1.1 – Порівняльна характеристика EM Marine та Mifare (Classic) Довжина серійного номеру 3 байта 4/7 байт Робоча частота 125 КГц 13,56 МГц Пам'яті - 1/4 КБ Присутність криптографічного захисту - + Crypto 1 Режим роботи Тільки зчитування Зчитує та записує Дальність зчитування 10 см До 6 см Можливість програмування - + Біля 100 000 циклів Захист від копіювання Прості СКУД Складні СКУД Окрім порівняння систем важливо зазначити їх суміжне різноманіття реалізації. «Смарт-карти» бувають активними та пасивними. Пасивні радіочастотні прилади – це прилади, що виводять зовнішню енергію, що надається від основного радіопередавача для забезпечення живлення для себе для обробки інформації. Загалом, дальність зчитування транспондерів від таких передавачів не більше 10ти сантиметрів, тому що такі карти передавати сигнал, він емулюється тактовими паузами, що створює мікросхема транспондера, коли живиться від гучності. Останній ці паузи зчитує та ініціалізує. Активні радіочастотні прилади – це прилади, у яких за схемотехнічною побудою

елемент живлення, що знаходиться поблизу чипа «смарт-карти». Подібні системи мають змогу виступати перед передавати інформаційний кодовий сигнал більш ніж на 10 метрів.

1.3 Огляд популярної СКУД Однією з найбільш компаній, що спеціалізується на проектуванні та виробництві систем контролю та управління доступом є «Bosch» гігант розробляє велику кількість різновидів СКУД, різного рівня захисту, екстраполяції та розгалуження, поч локальних мережевих систем контролю, закінчуючи потужними контролерами для великих підприємств та уст системи від німецького виробника зроблено саме через скупчення гарних характеристик, показників та інженерні цілому. Проаналізувавши систему можна отримати деякі «підказки» та поради до проектування прототипу сво «Access Easy Controller 2.1 (AEC2.1)» від «Bosch Security System»- це нове покоління контролера доступу, які собі можливості вбудованого веб-сервера та системи відеоспостереження. В цілому, хоч система найбільш схо класом із системою, що проектується, але вона є головним контролером, що під'єднує до себе та комутує викона різних пунктів надання доступу. Це відрізняє її за принципом роботи, адже система, що проектується, має бути відс виконавчим пристроєм та не має бути залежною від центрального контролера, окрім серверу, що розгортаєть подібній машині.

Рисунок 1.9 – Зображення контролера СКУД «Access Easy Controller 2.1 (AEC2.1)» Особливост адміністратора та співробітника:

- \* Кількість облікових записів адміністратора до 25 одиниць;
- \* 128-розрядний SSL шифрування веб-системи керування;
- \* Захист облікових записів співробітників. Усі ідентифікатори користувач зашифровані, забезпечуючи доступ лише для авторизованих користувачів для моніторингу та керування параметр записи транзакцій та діяльності;
- \* Має в наявності вбудований веб-сервер, за допомогою якого можна вести кор налаштування системи, використовуючи будь-який веб-браузер;
- \* Має вихід Ethernet, що дозволяє під'єднувати будь-якої IBM-подібної машини.
- \* Види методів ідентифікації співробітників в системі:
  1. Радіочастотна іден використанням 4-х або 7-ми байтного номеру;
  2. Ідентифікація за допомогою кодового пароллю;
  3. Об'єд ідентифікації, що дозволяє використовувати два вище вказаних методи одночасно.

Основні характеристики СК «Access Easy Controller 2.1 (AEC2.1)»:

- \* Центральний процесор: 32-розрядний мікропроцесор із швидкістю тактування ядра
- Внутрішня пам'ять: 512МБ (RAM) та 512МБ (Flash);
- \* Напруга живлення: 12 В постійного току;
- \* Порти вводу-вихід Ethernet та RS-232;
- \* Кількість зчитувачів можливих до підключення: до 4 (2 двонаправлених виконавчих пр
- Кількість систем, що у випадку масштабування можна підключити у єдину систему контролю: до 32 одиниць кон
- Кількість радіо ідентифікаційних транспондерів можливих до додавання: до 20 480 одиниць;
- \* Кількість можли ієрархій співробітників, що поділяють транспондери за привілеями доступу: до 254 груп;
- \* Можливість логування д присутня;
- \* Кількість пунктів можливого відеоспостереження: до 64 пунктів;
- \* Можливість SMS або e-Mail сп нестандартних випадках;
- \* Можливість перегляду пунктів, на яких є відеоспостереження у реальному часі;
- Підтримка різних мов у веб-додатку: присутня;
- \* Можливість підключення вихідних та вхідних органів: до 6 контролюючих датчиків або сигналізації типу гучномовця.

Рисунок 1.10 – Зображення панелі керування СКУД «Access Easy Controller 2.1 (AEC2.1)»

Окремо можна виділити можливість додавання фотографій осіб співробітників під час с індивідуальних сторінок. Це набагато полегшить роботу для контролюючої особи.

1.4 Створення принципової моделі

Здійснивши класифікацію й огляд СКУД, та найбільш поширений аналог, можна створити принципову схем відокремити блоки, підібрати протоколи зв'язку, за якими вони будуть взаємодіяти.

Рисунок 1.11 – Принципова схема

Структура системи контролю та управління доступом запозичує основу клієнт-серверної топології будован комплексів (рисунок 1.9). Як було вказано вище, принцип побудови такого зразка надає більш широкий функцію роботи та має вищий рівень безпеки зберігання та опрацювання даних. Система має керуватись із спеціальн контролю, який має знаходитись на деякому віддаленні від основного «шлюзу», де реалізується фізичний і допомогою спеціального пункту контролю мають бути реалізовані такі можливості як: зберігання даних корист унікальних ідентифікаційних номерів, безперервний контроль та фіксація дати та часу подій входу та виходу к внесення, зміна та видалення даних користувачів. СКУД має відрізнятись властивостями до боротьби із електро електричними та несанакційними програмними намаганнями втрутитись у її робочий процес. Також, має бути і баланс між бюджетними прототипами локальних систем контролю та професійними реалізаціями клієнт-серверних працюють на підприємствах та у деяких навчальних закладах, тощо. Отже, система контролю та управління належить до систем мережевого клієнт-серверного типу, з одним пунктом пропуску, але з апаратною та і можливістю розширювати комплекс, з'єднуючи в масив. Спосіб ідентифікації- радіочастотний з унікальни ідентифікації «UID». Оскільки прототип є тестовим та проект із програмним та електронно-апаратним погл механічна система чітко визначена не була і залишається із можливістю вибору між замками електромагнітні турнікетними типами.

1.5 Висновки за розділом

Було здійснено огляд предметної області охоронних та контролюю СКУД, їх класифікації. Також було оглянуто найбільш популярне рішення від надійного виробника, опис використання технології радіочастотної ідентифікації RFID та її вагу під час використання у реальних умовах

принципову схему СКУД до проектування залежно від потреб до системи та обрано напрямок вибору базових вузлів ВИБІР ПРОГРАМНО-АПАРАТНОГО КОМПЛЕКСУ ПРОЕКТУ 2.1 Вибір та обґрунтування базових вузлів для апаратного комплексу проекту Система контролю та управління доступом- це складний механізм, який скл сучасних обчислювальних блоків, де кожен з них чітко виконує своє встановлене завдання. Так зван високотехнологічних пристроїв відкриває великий простір для втілення різноманітних інженерних думок та алг якості реалізації схемотехніки та сумісності обраних електронних компонентів залежить безвідрадність роботи всієї цілому. На момент написання роботи основою або базою будь-якого електронного пристрою, в незалежно програмується він після цього чи ні, є напівпровідникові інтегральні мікросхеми (ІС). В цьому розділі дипломної роботи алгоритм вибору компонентної бази, обґрунтовано вибір кожного вузла об'єднаної системи та створено принциповий виконавчого пристрою (клієнту в клієнт-серверній СКУД) із проектуванням та виробництвом друкованої плати з дос основних вимог інженерного проектування. 2.1.1 Контролер виконавчого пристрою Клієнт-серверна система контролю та управління доступом в частині виконавчого пристрою вимагає використовувати центральний обчислювальний блок кожному сучасному пристрої, починаючи від інфрачервоного пульта управління або холодильнику, закінчуючи управлінням нахилу двигунами космічної ракети встановлено програмовану інтегральну мікросхему. Називає мікросхеми мікроконтролерами або мікропроцесорами. Приставка «мікро» характеризує їх щільність розташування транзисторних елементів на кристалі кремнію. Традиційно, мікропроцесор (MPU-Micro Processor Unit)- це пристрій, який відповідає за виконання арифметичних та логічних операцій та операцій керування, які записані у машині. Реалізований він на основі одного або двох кристалів кремнію. Мікропроцесор не обов'язково потребує додаткових блоків для його роботи, які зазвичай підключаються зовні, а саме: постійно-запам'ятовуючий та оперативно-запам'ятовуючий пристрої. В сучасності завдяки рідкісному використанню процесорів в реальному житті, які не є мікропроцесорами, лексичні терміни «мікропроцесор» та «процесор» рівнозначні. Внутрішня побудова мікропроцесору: арифметичний пристрій (АЛП), блок керування та синхронізації, запам'ятовуючий пристрій, регістри, шини передачі даних та команди. 2.1 – Зображення схематичної побудови мікропроцесора Особливістю мікропроцесорів є велика потужність обробки даних. Вони створюються для використання в потужних системах, таких як: сервери, персональні комп'ютери, телефони, спеціалізовані прилади, де потрібно обробляти великий об'єм інформації. Мікроконтролер (MCU- Micro Controller Unit) інтегральна мікросхема, що побудована на кристалі кремнію, так само, як і мікропроцесор. Типічний мікроконтролер містить кристалі мікропроцесор, периферійні пристрої, що потрібні для роботи мікропроцесора (ПЗП, ОЗП) та додаткові пристрої для роботи із зовнішніми пристроями, такі як: АЦП (Аналого-Цифровий Перетворювач), ЦАП (Цифро-Аналоговий Перетворювач), таймери різного призначення. Мікроконтролери ще називають одно кристальними комп'ютерами, через те, що вони не потребують майже ніяких додаткових блоків та є працездатними самі. Рисунок 2.2 – Зображення схематичної побудови мікроконтролера Мікроконтролер може виконувати великий спектр нескладних завдань, на відміну від мікропроцесора керування периферійними пристроями за допомогою відкритих протоколів, керування силовими елементами, індуктивними механізмами, за допомогою силових напівпровідників, керування дисплеями, блоками керування та управління відповідальності мікроконтролерів- керування вбудованих та переносних пристроїв. Також, мікропроцесор є сумісним з мікроконтролером та зазвичай застосовується в зв'язці. Таблиця 2.1 – Порівняльна характеристика мікропроцесора та мікроконтролера Мікропроцесор Мікроконтролер Реалізація Складається з однієї потужної інтегрованої мікросхеми мікропроцесора Мікропроцесор Мікроконтролер Реалізація Складається з однієї потужної інтегрованої мікросхеми мікропроцесора та зовнішніх пристроїв для функціонування Складається з мікропроцесора невеликої потужності та містить усі необхідні для функціонування блоки Характеристика Залежна одиниця Автономний блок Порти вводу/виводу - + Цільове призначення Ринок найвищого класу Вбудовані системи Енергоспоживання Не має вбудованого контролера розподілення живлення Має вбудований контролер живлення та менш прискіпливий до якості живлення Отже, проацільові напрямки використання двох видів сучасних обчислювальних пристроїв зроблено висновок, що в проекті контролю та управління доступом складських приміщень доцільніше використати основою обчислювального комплексу мікроконтролер. Система не потребує вирішувати тяжкі обчислювальні процеси і, навпаки, потребує деяких функцій вбудованих портів вводу та виводу інформації, роботи з протоколом з'єднання, таким як SPI (Serial Peripheral Interface) Щодо різноманітності мікроконтролерів, то тут вибір ще більший. Існують безліч сучасних одно кристальних електронних обчислювальних машин: Motorola, Zilog, AVR від Atmel, Nordic, ESP від Tensilica, PIC від Microchip, STMicroelectronics та інші. Велика їх кількість здатна опрацьовувати потреби СКУД складських приміщень, вирішити який саме вибрати. На момент написання дипломної роботи найбільшу популярність серед професійних вбудованих систем мають сучасні мікроконтролери від компаній Microchip та STMicroelectronics, серії PIC відповідно. До створення проекту була нагода працювати саме з мікроконтролерами STM32. Вони мають сучасну найвищі показники за характеристиками, чудове співвідношення ціни та якості та якісні компілятори від мікроконтролерів. Компанія STMicroelectronics працює з багатьма відомими компаніями та структурами у а

космічній промисловості, тож за якість можна не хвилюватись. Було обрано будувати систему на базі мікроконтроле виробника. Рисунок 2.3 – Зображення торгівельного знаку компанії ST Мікроконтролери виробника STMicro поділяються на декілька сімейств за розрядністю мікропроцесора, а саме: STM8 (8-розрядний), STM16 (16-розрядний (32-розрядний)). Було обрано 32-розрядну версію. По-перше, через те, що швидкість обробки даних у такого набагато більша, що дозволяє працювати з високошвидкісними протоколами передачі даних. По-друге, вартість 32 версії майже така ж, як і у інших. Основою реалізації було обрано STM32F103. Потужний мікроконтролер із обчислювальним ядром мікропроцесора Cortex-M3 з частотою обчислювання ядра та головної шини 72МГц. Також, присутня велика кількість портів вводу-виводу (I/O ports) та 2 окремих незалежних SPI (Serial Peripheral Interface) протоколи. Рисунок 2.4 – Зображення блоку побудови мікроконтролера STM32F103

Таблиця 2.2 – Характеристика мікроконтролера STM32F103

Мікроконтролер STM32F103 Flash пам'ять	64 КБ
SRAM пам'ять	20 КБ
Базові таймери	23 одиниці
Спеціальні таймери	1 одиниця
Протоколи SPI	2 одиниці
Протоколи I2C	2 одиниці
Протоколи USART	3 одиниці
Протоколи USB	1 одиниця
Шини входу та виходу GPIO	32 одиниці
12-розрядні АЦП	2 таймери по 10 каналів
Частота тактування	72 МГц
Напруга живлення	Від 2 до 3,6 вольт
Температурний діапазон	Від -40 до +80 (за Цельсієм)

2.1.2 Радіочастотна ідентифікація

Пристрій безконтактного обміну цифрових даних виконує функцію передавача та приймача «смарт-карти» або електронного ключа та прийому записаних даних в внутрішню пам'ять від них. Контролер є одним із центральних вузлів, які відповідають за безпеку та якість роботи усієї системи та її охороняє. Річ йде саме про контролер, що керує антеною передавача поблизу основного контролера «клієнта серверній системі. Оскільки раніше було зазначено, що обрано використовувати карти або ключі за технологією Mifare, то треба звернути увагу на якій частоті обміну вони працюють. Це 13,56 МГц, отже й передавач обрано саме за цим частотою. Основною перевагою є те, що прийому-передавач MFRC522 випускається тією ж самою компанією NXP, отже роботи за технологією Mifare. Мікросхема має велику та чітко сформульовану офіційну документацію для створення програмних бібліотек та функцій, а також для апаратного підключення. Рисунок 2.5 – Зображення структурної схеми контролера радіочастотної ідентифікації MFRC522

Рисунок 2.6 – Зображення блоку побудови контролера радіочастотної ідентифікації MFRC522

Мікросхема MFRC522 може керуватися за різними основними протоколами зв'язку, а саме UART. Живиться від тієї ж напруги, що і основний контролер. До мікросхеми під'єднується «антена», а якщо бути електромагнітна індуктивна котушка із чотирьох витків проводу. Вона одночасно виконує функцію живлення для «транспондерів та прийому-передачі інформації. За сутністю – це трансформатор, який не має магнітного ядра. Протоколом передачі обрано SPI (Serial Peripheral Protocol). Він найшвидший із запропонованих та здатен за швидкістю передачі інформації до 10 Мб/сек. В проекті СКУД складських приміщень обмін інформації відбувається з швидкістю 6 Мб/с для запобігання від втручання електромагнітних перешкод на дуже великій швидкості передачі. Швидкість I2C у даному випадку всього 100 або 400 Кб/с у швидкому режимі а UART до 1228.8 Мб/с. Таблиця 2.3 – Характеристика контролера MFRC522

MFRC522 Напруга живлення	Від 2,5 до 3,3 вольт
Протоколи комунікації	SPI, UART
Швидкість SPI	До 10 Мб/секунду
Частота тактування ядра	27,12 МГц
Сопроцесор підрахунку контрольної суми	Так
Дальність зчитування міток	До 50 міліметрів від антени до антени
Підтримка карток A/MIFARE	за стандартом ISO 14443A-3

2.1.3 Чіп зберігання ідентифікаційних даних

Щодо логіки роботи мікросхеми-транспондеру Mifare від NXP, або мікроконтролера зберігання даних було частинно розглянуто раніше. Обраний Mifare версії Classic – це сімейство, що складається з кількох типів: Classic 1K, MIFARE Classic 4K, MIFARE Classic EV1 1K, MIFARE Classic EV1 4K, MIFARE ID і MIFARE Mini. Mifare пропонують надбудову за стандартом ISO 14443A-3 з криптографічним захистом даних. Містить 4-х байтні або 16-байтні незмінювані унікальні ідентифікаційні номери карт та 1КБ або 4КБ даних користувача та конфігураційних даних пам'ять типу EEPROM, що є незалежною від живлення. У картах Mifare Classic використовується пропрієтарний криптоалгоритм Crypto-1. Спочатку стійкість алгоритму була заснована на його таємності. Алгоритм не розкрито, використовувати його можна було лише у складі мікросхем Philips (пізніше NXP Semiconductors). Однією з причин стійкості алгоритму і популярності технології призвела до того, що на сьогоднішній день алгоритм не є секретом і зламується. Але в реальних системах далеко не вся безпека побудована на апаратному шифруванні карти. Як фактор захисту можуть використовуватися, наприклад, мітки часу. Тим не менш, навіть системи, безпека яких не повністю на алгоритм Crypto-1 (або навіть не використовує його зовсім як Mifare Ultralight), можуть бути зламані апаратним особливостям карт. Рисунок 2.8 – Зображення блоку побудови інтегральної мікросхеми ідентифікаційних даних

Для карт стандарту 1K перші 1024 байт організовані у 16 секторів по 64 байти. Кожен сектор складається з 4 блоків. Останній блок кожного сектора називається трейлером сектора, у ньому записані ключі та параметри сектора. Нульовий блок нульового сектора – це спеціальний блок, заблокований на запис, він містить ідентифікаційну інформацію виробника картки. Перед зверненням до сектора для читання або запису необхідно виконати авт



допомогою ключа розміром 6 байт. Авторизація відбувається за три етапним протоколом, при цьому викори пропріетарний алгоритм потокового шифрування Crypto1. Рисунок 2.9 – Зображення структурної схеми кластеріе даних мікросхеми зберігання даних Комунікативний процес або процес з'єднання із зчитувачем відбувається за алгоритмом, що описаний у технічній документації виробника електронних ключів Mifare. Команди ініціюються з контролюються цифровим блоком керування MFIS50yyX/V1. 1) Цикл стандартного запиту. Після того, як н перезавантаження або Reset (POR–Power On Reset) подається живлення карта відповідає на запит команду з код на запит. 2) Цикл анти колізії (боротьби із зіткненням). У циклі анти колізії зчитується ідентифікатор карти. Як карток в операційному полі зчитувача, то їх можна розрізнити за ідентифікатором і може бути вибрана тільки одн здійснення подальших операцій. Невибрані карти повертаються в режим очікування стану і чекають нової команди Цикл вибору карти. За допомогою команди блок зчитування обирає одну окрему картку для автентифікації і опера з пам'яттю та картка повертає ідентифікаційний унікальний номер. 4) Цикл три факторної автентифікації. Після ви блок зчитування вказує розташування доступу до пам'яті та використовує відповідний ключ для три автентифікаційної процедури. Після успішної автентифікації всі команди та відповіді шифруються. Рисунок 2.10 – ( алгоритму обробки даних контролером радіочастотної ідентифікації MFRC522 Після проходження процесу авт відкриваються наступні можливості роботи із пам'яттю EEPROM: 1) Читання одного із 16 блоків 2) Запис в один із Декримент: зменшує вміст блоку та зберігає результат у внутрішньому буфері 4) Інкремент: збільшує вміст блоку результат у внутрішньому буфері 5) Відновлення: переміщує вміст одного з 16 блоків у внутрішній буфер і Передача: записує вміст внутрішнього буфера передачі в блок значення Таблиця 2.4 – Характеристика транспонд Classic EV1 1K Mifare Classic EV1 1K Безконтактна передача даних та живлення + Робоча частота 13,56 МГц ( підрахунку контрольної суми (CRC) 16–розрядний сопроцесор контрольної суми Час виконання однієї операції Ме Дальність зчитування Менше 50 мм Швидкість передавача 106 Кб/с Механізм антиколізії + Довжина ідентифікаційн 4 байти або 7 байтів Тип вбудованої пам'яті EEPROM Об'єм вбудованої пам'яті 1024 байти (16 секторів по 64 байт циклів перезапису Біля 200 000 2.1.4 Контролер з'єднання мережі В будь-яких системах клієнт–серверного типу об відокремлена частина «клієнт», де відбуваються одні обчислювальні процеси та відокремлена частина «сере аналогічно проходять обчислення спираючись на завдання. Ці дві системи хоч і є відокремленими, але все одно тандемі, виконуючи глобальне завдання всієї системи. Клієнт–серверна система контролю та управління доступом приміщень не є виключенням. Судячи з самої назви, система побудована на двох відокремлених вузлах обміну інф персональним комп'ютером, на основі якого побудований сервер, все зрозуміло. В ньому вбудована мережев допомогою якої можна під'єднуватись до будь-якого вузла мережевого зв'язку, якщо є його IP (Internet Protoc Апаратну частину мережевої карти зазвичай не довго й вибирають. Сучасні мережеві карти мають гарні характер більш для проекту такого рівня велика швидкість та складність обміну не потрібна, а у половині випадків, н ноутбуках, вона вмонтована на головну друковану плату, що ще більш спрощує процес вибору та подальшого пр системи. З «клієнтом», що у даному випадку є виконавчий блок клієнт–серверної системи трохи складніше. Че проектування вбудованих систем займає більш специфічну та відокремлену нішу, аніж збірка персонального к готових фабричних рішень, воно потребує більш щільного аналізу для подальшої інтеграції в систему. Як було вка: (див. 1.2.1 Принципова схема СКУД складського приміщення), СКУД тримає зв'язок за протоколом пакетної пер Ethernet (TCP/IP), то й контролер потрібен із його підтримкою. Проаналізувавши технологічний «ринок», яки контролери Ethernet–зв'язку для проектування та використання у вбудованих системах (IoT– Internet Of Th з'ясовано, що однією з найпопулярніших виробників мікросхем класу TCP/IP є компанія WIZnet. WIZnet – це к спеціалізується за платформою інтернету речей. Виробник обіцяє технологію жорсткого «TCP/IP», яка забезп стабільну та продуктивну роботу «вбудованого приладу», в порівнянні з використанням технологій програмнс локального або глобального зв'язку на базі основного обчислювального контролеру системи. Якщо поясню спрощено– контролер WIZnet безпосередньо бере участь у роботі та виконує усі складні алгоритми обчислен своїми силами. Це значно покращує становище для мікроконтролера STM32F103. Рисунок 2.11 – Зображення тс знаку компанії WIZnet Компанія WIZnet пропонує кілька апаратних ітерацій контролерів зв'язку, а саме: W3150 W5100S, W5100, W5200, W5300, W5500. Через те, що так званім «флагманом» є 5500 версія, було обрано саме 2.12 – Зображення схеми блоку мережевої комунікації Ethernet WIZnet W5500– це вбудований Ethernet–к підключенням TCP/IP, який забезпечує спрощене підключення до інтернету або локальної мережі для вбудовани допомогою SPI (Serial Peripheral Interface). W5500 найкраще підходить у випадках, коли потребується підключення до інтернету або локальної мережі, використовуючи єдиний чіп для реалізації стека TCP/IP, 10/100 Etb та PHY. Стек TCP/IP підтримує TCP, UDP, IPv4, ICMP, ARP, IGMP і PPPoE та інші. Це було доведено в різних програм: багатьох років. W5500 використовує внутрішній буфер 32 Кбайт як свою пам'ять для передачі даних. Використовун

можна реалізувати програму Ethernet, використовуючи технологію простих програмних сокетів замість роботи : контролером Ethernet, з потребою написання складних алгоритмів. Можливе використання 8 незалежних апарат одночасно. SPI (Serial Peripheral Interface) забезпечує легку інтеграцію із зовнішнім мікроконтролером. W5500 SF швидкість передачі даних до 80 МГц. У випадку потреби зменшити енергоспоживання системи, W5500 забез (Wake on LAN) і режим вимкнення живлення. Таблиця 2.5 – Характеристика контролера WIZnet W5500 WIZ1 Протоколи взаємодії TCP, UDP, ICMP, IPv4, ARP, IGMP, PPPoE Кількість сокетів 8 незалежних сокетів Режим живлення + Режим запуску за зовнішнім сигналом + Інтерфейс зовнішнього зв'язку Високошвидкісний SPI (д Внутрішня пам'ять 32 КБ для буферів TX/RX Режими роботи Повний і напів-дуплекс Фрагментація IP – Напруга жив Толерантність до напруги 5 вольт логічних сигналів + Зовнішня індикація + За основну швидкість роботи : інтерфейсу комунікації SPI обрано 18 МГц. Мікросхема контролера локальної мережі будується біля мікроконтролеру, що дозволяє працювати на високій швидкості. За конструктивними характеристиками тактування мікроконтролеру STM32F103 швидкість роботи шини оптимальна. Окрему увагу можна приділити якійсь програмні від виробника WIZnet. Вона має конструкції усіх базових програмних функцій для роботи з контролером, ц написання програми.

2.1.5 Контролер пристрою внесення даних Пристрій внесення даних – це електронний прил якому створюється внесення та перезапис ідентифікаційної інформації, що знаходиться у таблицях бази даних, чергу зберігає дані користувачів, що потрібні для коректної роботи із системою контролю та управління доступом приміщень. Було вирішено створення приладу на базі мікроконтролеру, через те, що від процесу обробки такої інс потребується потужність системи та велика кількість портів для взаємодії. В цьому випадку використовується 8 одно кристальний комп'ютер від компанії Atmel сімейства Mega моделі 328. На момент написання дипломної р найбільш популярним серед інших моделей цієї компанії та є повністю відповідаючим вимогам проекту СКУД. вартість є найменшою серед інших мікроконтролерів від Atmel, навіть, не дивлячись на його кращі характеристи моделі та серії. Рисунок 2.13 – Зображення торговельного знаку компанії Atmel Апаратно, мікроконтролер пристрою даних зв'язує другий контролер радіочастотної ідентифікації із персональним комп'ютером, на якому базується частина проекту. Як і у випадку з вузлом «клієнта», MFRC522 комунікує з мікроконтролером задіявши протокол

Рисунок 2.14 – Зображення структурної побудови мікроконтролера ATMega 328 Основою ж з'єднання мікрокс персональним комп'ютером є мікросхема-перетворювач серійного зв'язку SERIAL до універсальної синхронної ши допомогою цього рішення пристрій є переносним, через використання відомого стандарту зв'язку, апаратна можли надається більшості персональних комп'ютерних систем. Таблиця 2.6 – Характеристика мікроконтролера A1 Мікроконтролер ATMega 328 Flash пам'ять 32 КБ EEPROM пам'ять 512 Б SRAM пам'ять 2 КБ Базові таймери 2 розрядні) Спеціальні таймери 1 одиниць (16 розрядний) Протоколи SPI 1 одиниця Протоколи I2C 1 одиниця Протокок одиниця Протоколи USB Протоколи CAN 1 одиниця – Шини входу та виходу GPIO 23 одиниці 10-розрядні АЦП Частота тактування ядра 16 МГц Напруга живлення Від 1,8 до 5,5 вольт Температурний діапазон Від -40 до +85 (з; Рисунок 2.15 – Зображення методу підключення мікросхеми-перетворювача із комп'ютерної системою Табі Характеристика мікросхеми CN 340 CN 340 Підтримка USB +, версії 2.0 Підтримка операційних систем зовніш Windows та інші Швидкість передавача Від 50 б/с до 2 Мб/с Режими роботи Повний та напів-дуплекс Толерантніс логічних рівнів +, 3,3 та 5 В Напруга живлення Від 3,3 до 5 В 2.2 Вибір та обґрунтування базових рішень для програмного комплексу проекту Створення програмного комплексу системи контролю та управління доступом приміщень – друга проектна частина. Від коректного підбору сучасного та якісного програмного забезпечення тако безвідрадність та якість роботи всієї системи. Структура проектної частини було вирішено складати з чотирьох кодів, за допомогою яких можна користуватися функціоналом СКУД. Перша частина – програмне забезпеч програмування блоку виконавчого пристрою або «клієнта». Друга частина – програмне забезпечення для програмує серверної частини або «сервера». Третя частина – програмне забезпечення для програмування блоку внесення та даних користувачів. Четверта частина – програмне забезпечення для задля програмування графічного додатк ідентифікаційних номерів, що встановлюється на персональний комп'ютер. У цьому розділі дипломної роботи а різноманіття програмного забезпечення. Спираючись на це, робиться висновок щодо програмного забезпеченн використовуватись в роботі. Також, обґрунтовується вибір окремих рішень.

2.2.1 Програмне забезпечення для про блоку виконавчого пристрою Як було описано раніше, апаратна частина блоку виконавчого пристрою базуєть кристальному ЕОП (Електронно-обчислювальному пристрої) STM32 моделі F103 від компанії виробника STMicroe За роки професійної роботи виробник зміг створити багато різноманітних програмних продуктів, спрям програмування усієї лінійки мікроконтролерів STM, яких у неї більше 16 повноцінних серій. Однак, існує інший проду велику популярність серед програмних інженерів та не є плодом розробки від ST. Отже, порівнюється 2 основних с розробки для ведення розробки програмної частини «клієнта»: CubeIDE та KEIL uVision. Також, до інтегрованого с

розробки CubeIDE належить програмний додаток, що візуалізує та спрощує процес конфігурації налаштувань, через вважати, що додається ще й третій пункт порівняння Cube MX. KEIL uVision- програмний продукт, що дозволяє п проектами будь-якого ступеня складності, починаючи з введення та виправлення вихідних текстів і закінчуючи схемним налагодженням коду та програмуванням ПЗП мікроконтролера. Від розробника прихована більшість д функцій, що сильно розвантажує інтерфейс і робить управління інтуїтивно зрозумілим. Рисунок 2.16 – Зображені вікна програми «Keil» KEIL uVision користується багато інженерів програмного забезпечення, середовище не ноє було розроблено аж у кінці минулого століття. Програмне забезпечення вважається гарно виваженим та допр інтерфейс якого інтуїтивно зрозумілий. Також, за середовищем написано багато статей та є велика кількість д Таблиця 2.8 – Порівняння програми KEIL uVision KEIL uVision Переваги Недоліки Велика база до мікроконтрс докладна інформація до всіх пристроїв Інколи невірні вказівники помилок синтаксису коду Менеджер проектів К продукт з великою вартістю авторизації Вбудований редактор коду, що полегшує роботу з вихідним текстом М: програми, що компілюється, у випадку використання безплатної версії (до 32 КБ) Відладчик-симулятор, що налагоджувати роботу програми, що була скомпільована Майже відсутня довгострокова підтримка проекту для і керування, корегування та вдосконалення. Через деякий період проект можна не відновити Проста побудова інтегрованого середовища розробки Відсутня підтримка інших операційних систем окрім Windows Велика кількість та документації щодо роботи з середовищем розробки Доволі не новий продукт, що не мав великих вдосконаленн: часи експлуатації Висновок: проаналізувавши характеристики, властивості, переваги та недоліки інтегрованого с розробки KEIL uVision було з'ясовано, що більш за все, воно підходить для використання на великих підприє великих компаній, де йдеться постійна розробка профільних продуктів у сфері вбудованої електроніки та тих к програмне забезпечення комп'ютерних систем не змінюється довгостроково. Інтегроване середовище розробки С візуальний конфігуратор Cube MX буде розглянуто сумісно, адже вони позиціонуються як єдиний продукт. Тим не різних програмних продукти. Cube IDE- це відносно нове середовище, що було анонсоване у 2019 році STMicroelectronics як спеціалізований продукт програмної розробки під мікроконтролери тієї в компанії. Програми є безплатним як для комерційного, так і для приватного використання. Він започаткував усі відносно старі та ба інших, більш старих середовищ розробки для мікроконтролерів. Програмування ведеться мовою C, з використанн бібліотеки HAL та бібліотек низького рівня CMSIS. Бібліотека HAL (Hardware abstraction layer) має в своєму роз функції налаштовані на високий рівень програмування (інтуїтивно зрозумілі). Використовується у більшій частині г коду через легкість використання. Рисунок 2.17 – Зображення частини списку команд бібліотеки HAL Бібліотека CM Microcontroller Software Interface Standard) більш продуктивна, точна та швидка, використовується для процесів, де важливі інтервали часу: генерація ШИМ-сигналів, трактувань та апаратних пауз. Працює з реєстра 2.18 – Зображення частини списку команд бібліотеки CMSIS Основним нововведенням було створення цілого п комплексу у купі із візуальним додатком конфігурації налаштування Cube MX. Завдяки ньому вже не пот конфігурувати основні параметри мікроконтролеру самостійно, що зробити на архітектурі від ARM було до Ускладнювали це завдання велика кількість налаштувань апаратних таймерів, протоколів, портів вводу та вивод було зібрати у логічний конструкт для забезпечення комфортного програмування. Таблиця 2.9 – Порівняння прог IDE Переваги Недоліки Вбудований редактор коду, що полегшує роботу з вихідним текстом Інколи недопрацьовані порівнянні з KEIL uVision через те, що є відносно новим Відладчик-симулятор, що допомагає налагоджувати робот що була скомпільована Менша кількість інформації щодо продукту у порівнянні із KEIL uVision Вдосконалений проектів – Підтримка великої кількості операційних систем із якою працює середовище – Безкоштовний програмний для комерційного так і для приватного використання – Один і той же виробник програмного продукту що й мікро STM – Новіший продукт у порівнянні з KEIL uVision – Єдине та об'єднане середовище розробки з Cube MX – Огляд та базових функцій інтегрованого середовища розробки Cube IDE: 1) Поле написання основної програми. Тут ств записується основний її код мовою програмування C. Рисунок 2.19 – Зображення робочого поля написання програм програмі «Cube IDE» 2) Швидка панель доступу до основних функцій. Рисунок 2.20 – Зображення поля швидкого програмі «Cube IDE» 3) Панель доступу до файлів проекту, драйверів бібліотек, конфігураційних файлів та інших. Р – Зображення поля доступу файлів в програмі «Cube IDE» 4) Вікно консолі налаштування програмного коду. Рис Зображення консольного поля для налаштування програмного коду в програмі «Cube IDE» Cube MX є візуальним редактором для конфігурування мікроконтролерів сімейства STM32, що дозволяє генерувати код на осно використовуючи для цього графічних помічник. Графічний генератор коду здатен конфігурувати: 1) Конфігурація головного обчислювального ядра, тактування дільників та перед дільників, тактування шин, аналоговс перетворювачів. 2) Конфігурація протоколів: SPI, UART, USB, I2C, CAN. 3) Конфігурація прямого доступу до пам': Конфігурація таймеру підрахунку контрольної суми CRC. 5) Конфігурація входів та виходів мікроконтролера. Огляд

та базових функцій графічного конфігуратора налаштувань Cube MX : 1) Розділ вибору існуючих одно кристалън STM, отримання повної документації на мікроконтролери, огляд вартості в офіційному магазині. Рисунок 2.23 – 2) Поля вибору мікроконтролера в програмі «Cube MX» 2) Головна візуальна панель задання конфігурації пінів входів мікроконтролеру, що дозволяє бачити фізичну модель корпусів мікроконтролерів та порівнювати із проектом, що знаходиться в розробці. Рисунок 2.24 – Зображення поля конфігурації мікроконтролера в програмі «Cube MX» 3) Панель конфігурації: прямого доступу до пам'яті (DMA), пінів входів-виходів (GPIO), таймерів, аналогово-цифрових перетворювачів, протоколів зовнішнього з'єднання та інше. Рисунок 2.25 – Зображення поля конфігурації мікроконтролера в програмі «Cube MX» 4) Панель конфігурації тактувань ядра та периферійних пристроїв, таймерів, шин, дільників, лічильників. Завдяки цій панелі можна обрати джерело тактування ядра, джерело тактування годинника реального часу, налаштувати частоту тактування усіх можливих інформаційних шин мікроконтролера. Рисунок 2.26 – Зображення поля налаштування мікроконтролера в програмі «Cube MX» 5) Панель налаштувань «менеджер проекту». Завдяки панелі можна вибрати назву проекту, його розташування, інтегроване середовище розробки, до якого буде генеруватись конфігураційний код проекту, кількість доповнюючих налаштувань. Рисунок 2.27 – Зображення поля конфігурації проекту в програмі «Cube MX» 6) Таблиця математичного розрахунку та налаштувань живлення мікроконтролера та окремих його елементів, враховуючи тактичні параметри усіх елементів та температурні режими роботи процесора. Будування діаграм. Рисунок 2.28 – Зображення поля налаштування мікроконтролера в програмі «Cube MX» Таблиця 2.10 – Порівняння програми Cube MX Cube MX Переваги Недоліки

Бази даних усіх мікроконтролерів від виробника STM Сумісність тільки з інтегрованим середовищем розробки  
Наявність всієї технічної документації до кожного мікроконтролера Вбудований інтерфейс конфігурації тактування таймерів та ядра для подальшої генерації програмного коду Вбудований інтерфейс конфігурації портів вводу та виводу для подальшої генерації програмного коду Вбудований інтерфейс конфігурації протоколів зовнішнього зв'язку для генерації програмного коду Зручний візуальний спосіб налаштувань усієї периферії контролера Проаналізувати популярних середовищ розробки для створення програмної частини для програмування виконавчого пристрою Контроль та управління доступом було вирішено використовувати інтегроване середовище розробки від мікроконтролеру STM Cube IDE, та задля спрощення і запобігання помилок під час процесу налаштування генератор коду Cube MX.

### 2.2.2 Програмне забезпечення для програмування блоку серверної частини

Будова серверної частини – це найбільш накопичена щодо програмного коду частина прототипу автоматизованої системи контролю та управління доступом. Саме у цьому структурному блоці зберігається та обробляється уся корисна інформація, яка потрібна для автоматизації процесу контролю доступу, а саме: ідентифікаційні номери користувачів та відвідувачів, статус, у яких знаходяться, дати подій та час. Саме програма серверної частини дозволяє виконавчому пристрою відкрити і підтримувати коректної роботи серверної частини було обрано використовувати саме такі сучасні програмні рішення, а саме: середовище розробки «Microsoft Visual Studio 2022», веб-інтерфейс адміністрування систем управління базами даних «PHP збірку веб-серверу «XAMPP», мову високо-рівнявого програмування «C#», мову програмування «SQL», реляційну систему управління базами даних «MySQL». Microsoft Visual Studio 2022 – це сучасна лінійка продуктів компанії «Microsoft» у собі інтегровану середу розробки та інші продукти. За допомогою інтегрованого середовища розробки можна писати та змінювати програмний код на мові програмування «C#». Під час проходження практики використовується версія програми «Microsoft Visual Studio 2022» розширення «Community», що є безоплатною версією, завантаженою з офіційного сайту проекту. Рисунок 2.29 – Зображення робочого поля програмування в програмі «Microsoft Visual Studio 2022»

Веб-інтерфейс адміністрування систем управління базами даних «phpMyAdmin» – це веб-додаток з відкритим вихідним кодом написаний мовою PHP – є веб-інтерфейсом для адміністрування систем управління базами даних (СУБД) «phpMyAdmin» дозволяє за допомогою браузера здійснювати адміністрування сервера «MySQL», запускати команди «SQL», переглядати вміст таблиць та баз даних. Під час проходження переддипломної практики, веб-додаток «phpMyAdmin» одним із найбільш популярних інструментів для керування таблицями баз даних. Рисунок 2.30 – Зображення робочого поля програми «phpMyAdmin»

Збірка веб-серверу «XAMPP» – це між-платформна збірка веб-серверу, що поєднує в собі веб-сервер «Apache», «MySQL», інтерпретатор скриптів «PHP», що дозволяють створювати повноцінний локальний сервер. Збірка веб-серверу «XAMPP» є найбільш популярних щодо використання збірок. «XAMPP» – працює з усіма 32-розрядними ОС (98[3]/2000/XP/2003/Vista/7), а також з Linux, Mac OS X і Solaris. Програма вільно розповсюджується згідно GNU General Public License і є безкоштовним, зручним у роботі веб-сервером, здатним обслуговувати динамічні сторінки. З сьогоднішнім днем «XAMPP» є однією з найкращих збірок веб-сервера, за допомогою цієї збірки можна швидко розпочати роботу на своєму комп'ютері повноцінний та швидкий веб-сервер. Повний пакет збірки «XAMPP» включає в себе: 1) Web-сервер з підтримкою SSL 2) СУБД MySQL 3) PHP 4) Perl 5) FTP-сервер FileZilla 6) POP3/SMTP сервер 7) Утиліту phpMyAdmin

Рисунок 2.31 – Зображення робочого поля конфігурації вбудованих додатків в програмі «XAMPP» «C#» (або «C Sharp») – це об'єктно-орієнтована мова програмування високого рівня, що була створена інженерами компанії «Microsoft». За с

мова подібна до мов програмування «С» та «С++», а саме така мова («С») використовується під час про мікроконтролеру STM32, котрий є частиною апаратної реалізації прототипу проекту автоматизованої системи к управління доступом. Через велику кількість синтаксичних конструкцій, ця мова є потужним інструментом та заощ під час конструювання алгоритмів, які на інших мовах займатимуть набагато більше часу. Мова програмування «С найбільш надійних методів програмування через те, що задіє віртуальний процесор під час запуску написаних п при виникненні помилок вони не поширюються на інші програми, що працюють паралельно. Також, мова програму має потужний набір бібліотек, які використовуються під час роботи із серверними завданнями. Під час проходжен було звернено до українського інтернет видання «dou.ua», проведення незалежного опитування продемонструвал році програмування на «С#» залишається популярним та займає друге місце у загальному рейтингу. Рису Зображення статистики використання мов програмування серед розробників «SQL»- декларативна мова програм створена для роботи із реляційними базами даних. На відміну від «С#», є інформаційно-логічною мовою програм створена для опису, вилучення та внесення корективів у дані, що знаходяться у реляційній базі даних. За допомог даних можна зручно створювати таблиці, змінювати та вносити в них дані, змінювати їх структуру. Згодом SQL г збагатився новими конструкціями, забезпечив можливість опису та управління новими об'єктами, що зберігаються і індекси, уявлення, тригери і процедури, що зберігаються) – і став набувати рис, властивих мовам програмування. Пр змінах SQL залишається найпоширенішим лінгвістичним засобом взаємодії прикладного програмного забезпечені даних. У той самий час сучасні СУБД, і навіть інформаційні системи, використовують СУБД, надають користуваче засоби візуального побудови запитів. Рисунок 2.33 – Зображення робочого поля програмування в програмі «php «MySQL»- це свобідна реляційна система управління базами даних. Інструмент йде у пакеті програмного комплексу «ХАМРР». Є рішенням для малих та середніх додатків, що якнайкраще підходить по концепції «взаєм частини» автоматизованої системи контролю та управління доступом із «сервером». Розробку та підтримку MySQL корпорація Oracle, яка отримала права на торгіву марку разом із поглиненою Sun Microsystems, яка раніше шведську компанію MySQL AB. Продукт розповсюджується як під GNU General Public License, так і під власною к ліцензією. Крім цього, розробники створюють функціональність на замовлення ліцензійних користувачів. Саме зав замовленню майже в ранніх версіях з'явився механізм реплікації. Рисунок 2.34 – Зображення емблеми додатку « компанії «Oracle» 2.3 Висновки за розділом Під час виконання другого розділу було визначено та обґрунтов: електронних приладів та компонентів, що мають бути використані в проекті СКУД складських приміщень. розглянутий список програмного забезпечення та додатків, які сприятимуть процесу розробки програмних кодів с виконавчої частин та їх налаштувань. РОЗДІЛ 3 ПРОГРАМНО-АПАРATНА РОЗРОБКА ТА ВИПРОБУВАННЯ 3.1 Пр СКУД Проектування системи контролю та управління доступом складських приміщень- це предметна область, ц найбільшої уваги та терпіння. Після щільного аналізу програмних, апаратних комплексів, вибору об'єктів для досі розробки- можна починати складання системи. Завданням до розробки СКУД є: 1) Створення програмно/апара виконавчого пристрою або «клієнта» 2) Створення програмно/ апаратного блоку пристрою внесення та і інформації ідентифікаційних номерів 3) Створення програмного блоку серверної частини або «сервера» 4) комп'ютерного додатку до пристрою внесення та корегування інформації ідентифікаційних номерів 3.2 Пр виконавчого пристрою Проектування та побудова апаратної частини СКУД ґрунтується на потребах системи в ціло де і як вона буде працювати, які режими роботи зустрічатимуться, які впливи навколишнього світу загрожуватим процесу треба приділити увагу, бо електроніка має багато «підводного каміння», яке обов'язково треба врахувати під час розробки та проектування. Основу електронної компонентної бази виконавчого пристрою, з якої складаєть було розкрито у минулому розділі, а саме: мікроконтролер STM32F103, мікросхему підтримки мережі WIZN Мікросхему безконтактного зчитування електронних даних MFRC522 та чіп «смарт-карт» від Mifare Classic. розглянуто технологію збереження даних електронного ключа. Мета: розглянути важливі електроні комп сприятимуть правильній роботі електронної схеми виконавчого приладу СКУД, створити електронну схему виконавч СКУД, створити проект друкованої плати з розгляданням процесу конфігурації налаштувань для цього. Ри Зображення моделі друкованої плати контролеру СКУД Перш за все, треба проаналізувати умови використання си системою захисту та контролю. Перед тим, як вона почне процес контролю, її треба захистити саму. Насамперед який електронний прилад, що має велику кількість електронних напівпровідників та комунікує із іншим приладс фізичного поєднання їх струмопровідних провідників, прилад «боїться» впливу електричного струму на електр перешкод. Через це, обов'язковими є вузли захисту від зовнішніх впливів, оптичні, що мають діелектричний проміж все, вузли з'єднання електронних приладів безконтактного зчитування MFRC 522 із головним мікроконтролером для чого були обрані мікросхеми MAX 14850, які здатні «розв'язати», за допомогою оптичного бар'єру, інформ: мікроконтролера із зовнішнім світом. Рисунок 3.2 – Зображення частини діелектричного захисту принципової «

схеми По-друге, виконавчий пристрій має органи керування: кнопки, індикацію та кінцевий вимикач. Вони також без контактують із периферійними контактами мікроконтролера, де будь-який несанкційний вплив на них може вивести із робочого стану. Щоб уникнути цього- використовуються оптичні пари PC 817. Вони також мають бар'єр від струм частин на зовні. Рисунок 3.3 – Зображення частини діелектричного захисту принципової електронної схем мікросхема 74HC125D. Запобігає несвоєчасному спрацюванню електронних приладів безконтактного зчитування на MFRC 522. Виключає можливість виникнення помилок на інформаційній шині SPI. За допомогою неї відбувається «с зчитувачів із основним мікроконтролером. Рисунок 3.4 – Зображення частини буферної мікросхеми принципової « схеми Живиться схема від напруги 12 вольт, через те, що більшість пристроїв фізичного запобігання проникненню від цієї напруга, вона у більшості випадків є універсальною. Сама ж логічна частина схеми, починаючи від мікро закінчуючи контролером локальної мережі живиться від напруги логічного рівня 3,3 вольт. Для цього потрібна стабілізація напруги. Його параметри мають відповідати нормам низької пульсації постійної напруги для потрапляння перешкод на лінію живлення, що допоможе уникнути можливих проблем із працездатністю обчислювальних пристроїв та передавачів. Вузол перетворення та стабілізації напруги побудований на мікросхемі AMS 1117. Після стабілізації, згідно стандартам, присутня схема запобігання так званої «переполюсовки» живлення, коли при г плутаються провідники за полюсами. Також, розташовано силовий запобіжник для захисту сильно-струмової частини логічний запобіжник, що має захистити схему обчислювальних блоків у випадку виникнення нестандартних ситуацій. Рисунок 3.5 – Зображення частини блоку живлення контролера СКУД принципової електронної схеми Звісно ж, захист присутній і на приладах безконтактного зчитування, що розташовані не на друкованій платі виконавчого пристрою якості ізолятора живлення- ізоляційний перетворювач імпульсно-трансформаторного типу з гальванічною «розв'язкою» B1205, стабілізатор AMS 1117. Рисунок 3.6 – Зображення частини блоку живлення периферії принципової електронної схеми Вузол комутації зовнішнього навантаження, а саме електричного магніту, під який проектувався пристрій засилу силовому транзисторі MOSFET-типу K3918, що здатен до комутації великого струму. Перспективним етапом у розвитку є додавання контрольно-вимірювальної функції наявності навантаження. Це допомогло б системі коувесь час присутність навантаження на виході приладу СКУД. Рисунок 3.7 – Зображення частини комутації принципової електронної схеми WIZnet W5500- це вбудований Ethernet-контролер з підключенням TCP/IP, який спрощене підключення до інтернету або локальної мережі для вбудованих систем за допомогою SPI (Serial Interface). Особлива увага приділена до якості вузла живлення контролера. Мається роз'єднувальний трансформатор, який ізолює з'єднання Ethernet фізичного протоколу. Рисунок 3.8 – Зображення частини блоку Ethernet-контролеру г електронної схеми STM32F103 потужний мікроконтролер із обчислювальним ядром мікропроцесора Cortex-M3 обчислювання ядра та головної шини 72МГц, який є центральним вузлом обчислення для всіх периферійних блоків. Тактування відбувається зовнішнім резонатором кварцового типу. Рисунок 3.9 – Зображення частини обчислювального блоку принципової електронної схеми Створено повноцінну електричну схему за допомогою автоматизованого проектування «EasyEDA». Рисунок 3.10 – Зображення принципової електронної схеми СКУД Easy платформне веб-орієнтоване середовище автоматизації проектування електроніки, що включає редактор принципів, редактор топології друкованих плат, SPICE-симулятор, хмарне сховище даних, систему управління проектами, а також підготовки та замовлення виготовлення друкованих плат. Після створення електричної схеми СКУД розробляється плата, що фізично відображає спроектовану електричну схему. Використовуються всі загальні засоби та правила при друкованих платах. Наприкінці ДП конвертується в формат Gerber для числового програмного виготовлення спеціалізованому виробництві. Рисунок 3.11 – Зображення моделі двовимірної моделі друкованої плати автоматизованого проектування Після виробництва отримуємо готову друковану плату для подальшого монтажу є деталей за схемою. Монтаж відбувається спеціалізованим обладнанням за технологією SMD (поверхової) прив'язки. Рисунок 3.12 – Зображення друкованої плати СКУД 3.3 Проектування пристрою внесення даних Проектування та виготовлення апаратної частини пристрою внесення даних ґрунтується на потребах системи вносити та корегувати ідентифікатор користувачів в систему таблиць бази даних. Рисунок 3.13 – Зображення моделі пристрою внесення даних Основу «схеми» становить компонентної бази пристрою внесення даних, з якої складається система, було розкрито у минулому розділі мікроконтролер ATmega 328, мікросхема безконтактного зчитування даних MFRC522 та мікросхема-периферія інтерфейсів UART-USB CH340. Перший електронний вузол приладу зчитування та корегування даних- вузол Великих потреб до нього не представляється, через те, що прилад працює в умовах офісного простору біля комп'ютера бази даних, де зберігається уся інформація. Окрім цього- персональний комп'ютер і без того має гарну якість струму на виходах інформаційних портів USB, тому можна обійтися елементарною базою. Рисунок 3.14 – Зображення частини блоку живлення принципової електронної схеми Другий вузол, це вузол обчислення, зчитування даних. Мікроконтролер тактується зовнішнім кварцовим резонатором на 16 МГц. Інформаційна шина приладу без

зчитування під'єднана до апаратного протоколу зовнішньої комунікації SPI. Працює на низькій швидкості, тому ш даному випадку не потребує. Мікросхема-перетворювач протоколів USB-UART під'єднана до двох виходів прийому даних (RX/TX). Також, вона виконує функцію знижувального стабілізатора напруги до 3,3 вольт для живлення безконтактного зчитувача MFRC522 та модуля програмування для мікроконтролера. Рисунок 3.15 – Зображення г електронної схеми Як і у випадку виконавчого пристрою «клієнта», проектується друкована плата, спираючись на схему пристрою. Таким чином компонується Gerber-файл проекту та віддається на виробництво. Рисунок 3.16 – 3 двовимірної моделі друкованої плати в системі автоматизованого проектування 3.4 Програмування комплексу і пристрою Як було описано раніше, для проектування програмної частини виконавчого пристрою СКУД бу використовувати інтегроване середовище розробки Cube IDE та візуальний генератор конфігураційного програ Cube MX. Спочатку, створюється проект у генераторі конфігураційного програмного коду Cube MX. Обирається мікроконтролер STM32F103. Надається значення усім потрібним контактам логічного входу та виходу, таким світлодіодів індикатора, вихід звукового індикатора на основі п'єзо-випромінювача, вихід до силового транзистору К кнопки виходу та кінцевого вимикача. Рисунок 3.17 – Зображення конфігураційної моделі мікроконтролера «CubeMX» Визначаються джерело, параметри тактування головного ядра та інформаційних шин. Обирається так кварцового резонатору на 8 МГц. Частота роботи процесора мікроконтролера 72 МГц, частота тактування шин 36 Рисунок 3.18 – Зображення конфігураційної панелі тактування мікроконтролера Визначаються основні хар: протоколу SPI до пристроїв зчитування на мікросхемах MFRC 522: Рисунок 3.19 – Зображення налаштування протоколу із MFRC522 в програмі «CubeMX» Таблиця 3.1 – Опис характеристик протоколу SPI1 SPI 1 Режим ро дуплексний (Full Duplex Master) Розмір пакету даних 8 біт Перший біт MSB First Дільник тактування шини 8 тактування шини 4,5 МГц (4,5 Мб/с) Фаза такту 1 Edge Визначаються основні характеристики протоколу SPI д з'єднання мережі Ethernet: Рисунок 3.20 – Зображення налаштування фізичного протоколу із W5500 в програмі Таблиця 3.2 – Опис характеристик протоколу SPI2 SPI 2 Режим роботи Повно дуплексний (Full Duplex Master) Рс даних 8 біт Перший біт MSB First Дільник тактування шини 2 Швидкість тактування шини 18 МГц (18 Мб/с) Фаза т Визначаються характеристики конфігурації внутрішнього базового таймера для роботи з часовими інтервалами т імпульсна модуляція для роботи з п'єзо-випромінювачем. Рисунок 3.21 – Зображення налаштування базового програмі «CubeMX» Таблиця 3.3 – Опис характеристик базового таймера Base Timer Режим роботи Лічильни сторону Дільник шини тактування 3 Період рахунку 6000 Режим ШІМ (модуляції) PWM Mode 1 Канал ШІ Налаштування проекту, а саме надання йому ім'я, визначення директорії знаходження, вибір інтегрованого с розробки для генерації програмного коду та інші. Рисунок 3.22 – Зображення налаштування проекту в програмі Далі натисканням кнопки «Generate Code» генерується стартові конфігураційні налаштування апаратного середовища «Cube IDE» мовою C, потім відкриває його. Для коректного програмування пристрою треба визначити пристроїв, які мають протоколи з'єднання та вбудований контролер, що використовує ці протоколи. Бібліотеки: RC 5 W 5500, Wizchip\_conf. Файли для будування бібліотек надаються виробниками мікросхем MFRC522 та W5500. Р – Зображення налаштування проекту в програмі «CubeMX» Програмування, як було вказано раніше, відбувається Мова C- компільована статично типізована мова програмування загального призначення, розроблена в 1969- співробітником Bell Labs Денісом Рітчі як розвиток мови Бі. Систематично використовується програмними вбудованих систем для програмування одно кристальних ЕОМ завдяки гарним характеристикам компіляторів та потужному синтаксису написання програмного коду. По-перше, визначаються бібліотеки, що додалися у файл проє 3.1 – Додавання бібліотек на початку проекту #include "main.h" #include "stdio.h" #include #include "socket.h" "w5500.h" #include "rc522.h" #include "rc5221.h" «main.h» додає заголовний файл основного коду програми. І головні ідентифікатори портів вводу та виводу мікроконтролера. Це полегшує роботу із портами під час програмуван інтуїтивно зрозумілому формуванню. Бібліотека "stdio.h"- це заголовний файл стандартної бібліотеки мови C, за якої можна працювати із вводом та виводом даних в консоль та буфери. Має вбудовані макроси, структури та типи формування правильного запиту для стандартних операцій. Бібліотека – також заголовний файл стандартної бібл C, за допомогою якої є можливість працювати із вводом та виводом рядків, що закінчуються та 0. Бібліотек надається виробником контролеру Ethernet від WizNET. Враховує всі базові функції для можливості роботи із п сокетамі в системи, такі як: підключення, прийом та передача даних. Бібліотека "w5500"- також надається і контролеру Ethernet та знаходиться ц відкритому до завантаження доступі. Має функції для роботи із : налаштуваннями контролеру зв'язку, є основною бібліотекою для з'єднання частини клієнта та сервера. "rc522" · бібліотеки для роботи із мікросхемою радіочастотної ідентифікації MFRC522. Створено 2 різних класи для спрощен кожним ідентифікатором окремо. Класи мають одні й ті ж функції для роботи з чіпом. Створені за допомогою документації мікросхем із описом базових функцій роботи з ними. Додаються ідентифікатори для спрощення і

розуміння коду та можливості зручного коректування інформації в процесі налагодження приладу: Лістинг 3.2 – назв констант #define DATA\_BUF\_SIZE 2048 #define TCP\_SOCKET 0 #define PORT\_TCPS 5000 #define DATA\_BUF\_SIZE 2048 uint8\_t gDATABUF[DATA\_BUF\_SIZE]; Створюється масив даних налаштувань із мережевими конфігураціями (адреса Ethernet- контролеру, IP- адреса, маска підмережі, адреса маршрутизатору мережі) та визначаються базові функції для роботи із мережевим пристроєм WizNET за протоколом SPI. Фізична адреса Ethernet-контролеру власноруч, варіативність за бажанням, а не як зазвичай, прописується виробником мікросхеми під час виробництва – Конфігурування змінних wiz\_NetInfo gWIZNETINFO = { .mac = {0x80, 0xfa, 0x5b, 0x6a, 0xe2, 0xd1}, .ip = {192, 168, 31, 203}, .sn = {255, 255, 255, 0}, .gw = {192, 168, 31, 1}, .dns = {0, 0, 0, 0}, .dhcp = NETINFO\_DHCP}; void W5500\_SetGPIO\_WritePin(GPIOA, SPI2\_CS1\_Pin, GPIO\_PIN\_RESET);//correct} void W5500\_UnselectGPIO\_WritePin(GPIOA, SPI2\_CS1\_Pin, GPIO\_PIN\_SET);//correct} void W5500\_ReadBuff(uint8\_t\* buff, uint8\_t len, HAL\_SPI\_Receive(&hspl2, buff, len, HAL\_MAX\_DELAY);} void W5500\_WriteBuff(uint8\_t\* buff, uint8\_t len, HAL\_SPI\_Transmit(&hspl2, buff, len, HAL\_MAX\_DELAY);} uint8\_t W5500\_ReadByte(void){ uint8\_t byte; W5500\_ReadBuff(&byte, sizeof(byte)); return byte;} void W5500\_WriteByte(uint8\_t byte){ W5500\_WriteBuff(&byte, sizeof(byte));} uint8\_t stat; Визначаються приватні функції, масиви, цілочисельні на символічні типи даних до ідентифікаційних номерів, конфігурації мережевого пристрою та інші. Також, функції ініціалізації портів вводу протоколів зв'язку та передачі даних, що визначаються автоматично конфігуратором CubeMX. Лістинг 3.4 – функції ініціалізації структурних блоків, визначення змінних void SystemClock\_Config(void); static void MX\_GPIO\_Init(void); static void MX\_SPI1\_Init(void); static void MX\_SPI2\_Init(void); static void MX\_TIM2\_Init(void); uint8\_t blockAddr; uint8\_t status; uint8\_t str[MFRC522\_MAX\_LEN]; uint8\_t sn[5]; uint8\_t snl[5]; uint8\_t bo = 0; char snbuff[64]; char statbuff[64]; char statbuffl[64]; uint8\_t i; Визначаються основні функції обробники з програмою сокетом, отримання та відправки через сокет для першого «клієнт» (вхідного) та другого «клієнт» пристроїв, функції обробки вхідної інформації. Сокет (Socket)– це назва програмного інтерфейсу забезпечення обміну між процесами. Процеси при такому обміні можуть виконуватися як на одному пристрої обміну, так і різних, пов'язаних між собою мережею. Сокет – абстрактний об'єкт, що представляє кінцеву точку з'єднання. Для взаємодії між машинами за допомогою стека протоколів TCP/IP використовуються адреси та порти. Адреса є 32-бітною структурою для протоколу TCP. Адреса порту – ціле число в діапазоні від 0 до 65535 (для протоколу TCP). Лістинг 3.5 – Створення функції передатку даних за допомогою TCP-сокета void SendSocket(void){ stat = socket(TCP\_SOCKET, Sn\_MR\_TCP, 8080, 0); uint8\_t addr[4] = {192,168,31,203}; stat = connect(TCP\_SOCKET, (uint8\_t\*)addr, 8080); HAL\_Delay(5); send(0, (uint8\_t\*)statbuff, strlen(statbuff)); send(0, (uint8\_t\*)snbuff, strlen(snbuff)); HAL\_Delay(5); uint8\_t rc[32]; recv(0, ((uint8\_t\*)rcbuff), 32, 0); sprintf(rcbuff, "%c", rc[0]); if (rc[0] == 'R') { HAL\_GPIO\_WritePin(GPIOA, MOSFET\_Pin, GPIO\_PIN\_SET); for(i=0; i<=2; i++){ HAL\_TIM\_PWM\_Start (&htim2, TIM\_CHANNEL\_4); HAL\_Delay(50); HAL\_TIM\_PWM\_Stop (&htim2, TIM\_CHANNEL\_4); HAL\_Delay(100);} HAL\_Delay(1550); HAL\_GPIO\_WritePin(GPIOA, MOSFET\_Pin, GPIO\_PIN\_RESET); HAL\_Delay(2000);} disconnect(TCP\_SOCKET); close(TCP\_SOCKET); HAL\_Delay(5);} У циклі коду «main» виконуються функції, що виконуються один раз при старті системи. По-перше, відбувається ініціалізація портів входу та виходу, ініціалізація протоколів передачі даних SPI, ініціалізація всіх периферійних пристроїв та базових системних таймерів, лічильників таймерів. По-друге, після запуску протоколів, передаються дані до мікросхеми мережевого зв'язку, визначаються буферні для зберігання інформації, що передається від «клієнта» до «сервера». Запускається стартовий звуковий сигнал звукового індикатора. Лістинг 3.6 – Стартовий блок програмного коду MX\_GPIO\_Init(); MX\_SPI1\_Init(); MX\_SPI2\_Init(); MX\_TIM2\_Init(); HAL\_Delay(200); for (i=0; i<=2; i++){ HAL\_TIM\_PWM\_Start (&htim2, TIM\_CHANNEL\_4); HAL\_Delay(50); HAL\_TIM\_PWM\_Stop (&htim2, TIM\_CHANNEL\_4); HAL\_Delay(100);} HAL\_Delay(500); HAL\_TIM\_PWM\_Start (&htim2, TIM\_CHANNEL\_4); HAL\_Delay(500); HAL\_TIM\_PWM\_Stop (&htim2, TIM\_CHANNEL\_4); HAL\_GPIO\_WritePin(GPIOA, RF\_RST\_Pin, GPIO\_PIN\_SET); HAL\_GPIO\_WritePin(GPIOA, LAN\_RST\_Pin, GPIO\_PIN\_RESET); HAL\_GPIO\_WritePin(GPIOA, LAN\_RST\_Pin, GPIO\_PIN\_SET); HAL\_Delay(1000); reg\_wizchip\_cs\_cbfunc(W5500\_CS1\_Pin, W5500\_Unselect); reg\_wizchip\_spi\_cbfunc(W5500\_ReadByte, W5500\_WriteBuff); reg\_wizchip\_spiburst\_cbfunc(W5500\_ReadBuff, W5500\_WriteBuff); uint8\_t rx\_tx\_buff\_sizes[] = {2, 2}; wizchip\_init(rx\_tx\_buff\_sizes, rx\_tx\_buff\_sizes); wizchip\_setnetinfo(&gWIZNETINFO); ctlnetwork(CN\_SENETINFO, (void\*) &gWIZNETINFO); HAL\_Delay(1000); Наприкінці- основний цикл програмного коду «While». У ньому відбуваються за безкінцевим циклом. Сюди завантажуються функції перевірки програмних сокетів та їх аналізується інформація, що надходить з приладів безконтактного зчитування на мікросхемах MFRC522. Також, основні апаратні затримки без кінцевого циклу. Процеси утворюються та обробляються постійно, поки ввімкнено основний мікроконтролер STM32F103. Лістинг 3.7 – Основний блок циклічного програмного коду while (1){ MFRC522\_Init(); status = MFRC522\_Request(PICC\_REQIDL, str); status = MFRC522\_Ant



```
(status == MI_OK) { sprintf(snbuff, "%x%x%x%x\r\n", sn[0],sn[1],sn[2],sn[3]); sprintf(statbuff, "ENTER\r\n"); SendSocket1(); HAL_Delay(5);} status = MI_ERR; status = MFRC522_Request1(PICC_REQIDL, str); status = MFRC522_Antic
(status == MI_OK) { sprintf(snbuff1, "%x%x%x%x\r\n", sn1[0],sn1[1],sn1[2],sn1[3]); sprintf(statbuff1, '
SendSocket1(); HAL_Delay(5);} 3.5 Програмування обробника бази даних З'єднання серверу та клієнту, а саме в
блоку на мікроконтролері, відбувається за протоколом з'єднання TCP (Transmission Control Protocol). Він є одним
протоколів передачі даних у мережі інтернет. До його переваги належить велика надійність до передачі та прийом
даних. Призначений для керування передачею даних інтернету. Пакети TCP називаються сегментами. У стеку
TCP/IP виконує функції транспортного рівня моделі OSI. Основний алгоритм роботи за допомогою протоколу TCP
з'єднанні полягає у тому, що «клієнт» (одиниця зв'язку) відкриває запит до «серверу» (інша одиниця зв'язку), яка в
прослуховує запити до неї. Після з'єднання між двома учасниками зв'язку проходить передача потрібних дан
відправлення на серверну частину зчитаного ідентифікаційного коду виконавчим пристроєм. Після цього сервер
встановлений зв'язок на руйнує Socket з'єднання. TCP здійснює надійну передачу потоку байтів від одного процес
Перший етап, це встановлення інтегрованого середовища розробки Visual Studio 2022, додатків та їх наг
Програмування відбуватиметься за допомогою високорівневої мови програмування C#. Створюється консольний
назвою TCP_Server, що виконуватиме обов'язки програми-конектора із базою даних MySQL. Після цього додає
директорій, які забезпечують можливість використання готових структур та функцій для подальшого написання г
коду за допомогою їх можливостей. Лістинг 3.8 – Додавання бібліотек на початку проекту using System; using Sy
using System.Collections.Generic; using System.Linq; using System.Text; using System.Threading.Tasks;
using System.Net; using System.Net.Sockets; using MySql.Data.MySqlClient; Клас System означає, що викори
бібліотека System у проекті. Це дає деякі корисні класи та функції, такі як клас Console або функція/метод Writ
System.Data дозволяє об'єкту реалізувати DataAdapter і представляє набір методів і властивостей, пов'язаних
використовуються для заповнення та оновлення набору даних та оновлення джерела да
System.Collections.Generic забезпечує загальну реалізацію стандартної структури даних, як-от пов'язані списки, і
та словники. Клас System.Linq надає різні типи класів і методів, які підтримують запити LINQ. Клас System.Text мі
які представляють кодування символів ASCII та Unicode. Абстрактні базові класи для перетворення блоків симвл
байтів та з них і допоміжний клас, який маніпулює та форматує об'єкти String без створення проміжних екземпл
Клас System.Threading.Tasks надає типи, які спрощують роботу з написанням паралельного та асинхронного
System.Net означає, що використовується бібліотека для розробки програм Windows Store або настільних пр
System.Net.Sockets- це підключення бібліотеки для роботи із програмними сокететами. Програмування сокетів
з'єднання двох вузлів мережі для зв'язку один з одним. Це одностороннє налаштування клієнта та сервере
підключається, надсилає повідомлення на сервер, а сервер показує їх за допомогою підключення через сокет. Де
встановлення бібліотеки MySql.Data.MySqlClient, за допомогою якої програма Dot Net матиме змогу зв'язку і
MySQL. Після встановлення бібліотеки з офіційного сайту MySQL, вона так само додається до проекту за допомо
using. Розглянемо основні функції алгоритму взаємодії процесів в програмному коді, що відповідають за процес п
відправки та отримання даних за допомогою сокет-з'єднання. По-перше, перед переходом до основного циклу, в
отримаємо ім'я хосту (ім'я машини), якою розгорнуто серверний блок та виведемо його у строку за допомогою н
додатку. Лістинг 3.9 – Блок отримання ім'я хосту String strHostName = string.Empty; strHostName = Dns.GetHo
Console.WriteLine("Local Machine's Host Name: " + strHostName); По-друге, отримаємо адресу IP машини дв
видів: версій ipv4 та ipv6 та виведемо їх у строку за допомогою консольного додатку. Якщо кількість адресів бі
зпускається функція вибору потрібної адреси. У даному випадку система працює з використанням адреси ве
технічних можливостей для роботи в локальній мережі достатньо. Лістинг 3.10 – Блок отримання IP- адреси for (i
addr.Length; i++) { Console.WriteLine("IP Address {0}: {1} ", i, addr[i].ToString()); } String ip; if (addr.Le
Console.Write("There is more than one ipv4 address. Please select number ="); int i = int.Parse(Console.Read
= addr[i].ToString()); }else{ string st1 = addr[0].ToString(); string st2 = addr[1].ToString(); if (st1.Length < st2.Le
addr[0].ToString(); else ip = addr[1].ToString();} Далі вмикається процес авторизації, де необхідно ввести логін
щоб перейти до блоку підключення з базою даних MySQL. Також, формується рядок SQL-запросу, що буде відпра
містить: порт підключення, ім'я бази даних, до якої є потреба підключення та авторизаційні дані. Після введенн
проходження авторизації очищуємо консольний вивід. Відправляємо запит із вказаними даними за вказаними р
Якщо з'єднання трапилось- до консолі надсилається відповідна рядок узгодження, якщо відбулась помилка- на
рядок, що сповіщає про помилку. Лістинг 3.11 – Блок авторизації Console.Write("Input login:"); string
Console.ReadLine(); Console.Write("Input password:"); string password = Console.ReadLine(); string Conn
(password == "")){ ConnectString = " datasource = localhost; port=3306; username = " + login + ";database =
```

```

}else{ ConnectString = " datasource = localhost; port=3306; username = " + login +
="+password+";database = db_acs;"} Console.Clear(); MySqlConnection DBConnect = new MySQLConn

```

основному програмному циклі while за допомогою локальних змінних створюємо буферний простір для зберігання даних, які надходять до пристрою зчитування. Вводимо програму у режим сокет прослуховування, коли машина надходить запит за з'єднання від клієнта. Лістинг 3.12 – Блок визначення буферів

```

var bufferSTAT = new byte[256];
var bufferUID = new byte[256];
var sizeSTAT = 0;
var sizeUID = 0;

```

Далі відбувається процес прийому надходження даних до пристрою зчитування. Для зберігання даних застосовується вище створені буферні простори bufferSTAT та bufferUID. Лістинг 3.13 – Блок прийому даних

```

do{ sizeSTAT = listener.Receive(bufferSTAT);
dataSTAT.Append(Encoding.UTF8.GetString(bufferSTAT, 0, sizeSTAT));
sizeUID = listener.Receive(bufferUID);
dataUID.Append(Encoding.UTF8.GetString(bufferUID, 0, sizeUID));
}while (listener.Available > 0);

```

Після прийому даних відбувається алгоритм порівняння ідентифікаційних даних, що надійшли, із даними, що зберігаються у базі даних. Надається відповідь до частини клієнту у випадку співпадання даних із наявними у базі даних. Лістинг 3.14 – Блок порівняння ідентифікаційних даних

```

MySQLCommand command = new MySQLCommand("SELECT * FROM t_Status=1 AND UID = @a;", DBConnect);
command.Parameters.Add("@a", MySQLDbType.VarChar).Value = a;
adapter.SelectCommand = command;
adapter.Fill(table);
if (table.Rows.Count > 0){
int empID = table.Rows[0][2].ToString();
int termID = table.Rows[0][1].ToString();
listener.Send(Encoding.ASCII.GetBytes("R"));
listener.Shutdown(SocketShutdown.Both);
listener.Close();
}

```

Цикл обробки закінчується відправкою даних до бази даних, що надаються при вході або виході, та запис їх у згідні таблиці бази даних. Лістинг 3.15 – Блок відправки даних до бази даних

```

command.Parameters.Add("@d", MySQLDbType.VarChar).Value = empID;
command.Parameters.Add("@t", MySQLDbType.VarChar).Value = termID;
command.Parameters.Add("@u", MySQLDbType.VarChar).Value = empID;
adapter.SelectCommand = command;
adapter.Fill(temp);

```

Також, створено наявне відображення алгоритму роботи частини за допомогою блок-схеми, що описує послідовність роботи програмного коду. Рисунок 3.24 – Зображення алгоритму роботи методу підключення

Наступним кроком є формування структури реляційної бази даних та її таблиць з сортування. За допомогою цієї системи зберігатиметься список обов'язкових та додаткових даних, потім перевірка системи в цілому. Як було розглянуто раніше, основою маніпуляції інформацією між таблицями та зчитування в проекті системи контролю та управління доступом є реляційна система управління БД MySQL. Для роботи з блоком зберігання встановлюється збірка веб-серверу XAMPP. З нею набагато зручніше буде працювати адже пакет містить потрібну MySQL та веб-інтерфейс PHPMyAdmin, що дозволяє адмініструвати вказану СУБД контролю збірки обираються порожні порти, за якими працюватиме СУБД, після вмикається режим адміністратора переносить до панелі керування PHPMyAdmin. Рисунок 3.25 – Зображення панелі налаштування програми

Методом декларативної мови програмування SQL створюється база даних «db\_acs». В базі даних створюються таблиці: «t\_contact», «t\_employ», «t\_log», «t\_position», «t\_terminal», «t\_uid», що умовно поділяються на 2 осні рівні: 1) Рівень внесення логів даних співробітника (Employee Layer); 2) Рівень базової ідентифікаційної інформації (Entrance Layer). Рисунок 3.26 – Зображення структурної схеми бази даних

Спочатку розглянемо рівень Entrance Layer, що має відповідати за можливість контролю подій входу та виходу, не дозволяючи тим чи іншим діям на пункті пропуску виконавчого пристрою. Із цієї шарою працює програма-обробник TCP/IP, описувався вище в цьому розділі. Рівень працює із трьома таблицями: «t\_log», «t\_terminal» та «t\_uid». \* Таблиця «t\_uid» містить 3 осередки: «UID», «EmployID», «Status» 1) «UID»- зберігає ідентифікаційні дані радіочастотних міток на які працює алгоритм порівняння під час дозволу пропуску; 2) «EmployID»- зберігає номер користувача картою в таблиці; 3) «Status»- зберігає статус активності мітки. Лістинг 3.16 – Створення таблиці «t\_uid»

```

CREATE TABLE `t_uid` ( `ID` int(11) NOT NULL, `EmployID` int(11) NOT NULL, `Status` int(11) NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

```

\* Таблиця «t\_log» містить 5 основних осередків: «Dir», «UID», «TerminalID», «EmployID», «\_datetime». 1) «Dir»- зберігає інформацію логування щодо напрямку руху за пунктом пропуску виконавчого пристрою; 2) «UID»- зберігає ідентифікаційний номер до власника ідентифікаційним номером; 3) «TerminalID»- зберігає логування номеру виконавчого пристрою; 4) «EmployID»- зберігає логування номеру користувача картою в таблиці; 5) «\_datetime»- зберігає дату та час подій, що відбулися на терміналі. Лістинг 3.17 – Створення таблиці «t\_log»

```

CREATE TABLE `t_log` ( `ID` int(11) NOT NULL, `Dir` varchar(50) NOT NULL, `UID` varchar(50) NOT NULL, `TerminalID` int(11) NOT NULL, `EmployID` int(11) NOT NULL, `_datetime` datetime NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

```

\* Таблиця «t\_terminal» містить 2 осередки: «Name», «Location». 1) «Name»- зберігає ім'я терміналу в системі терміналів; 2) «Location»- зберігає місцезнаходження або адресу фізичного терміналу в системі терміналів. Лістинг 3.18 – Створення таблиці «t\_terminal»

CREATE TABLE `t\_terminal` ( `ID` int(11) NOT NULL, `Name` varchar(50) NOT NULL, `Location` varchar(200) NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4; Рівень бази даних Employee Layer здебільшого задіяний у фоновому зберіганні даних до пристрою внесення даних нових користувачів і працівників, тим не менш, цей рівень належить до бази даних «db\_acs», отже буде описаний в цьому ж підрозділі. Рівень працює теж із трьома таблицями: «t\_employ», «t\_contact».

\* Таблиця «t\_employ» містить 6 осередків: «FirstName», «LastName», «Patronymic», «DateOfBirth», «Passport», «PositionID». 1) «FirstName»- зберігає ім'я співробітників; 2) «LastName»- зберігає прізвища співробітників; 3) «Patronymic»- зберігає ім'я по-батькові співробітників; 4) «DateOfBirth»- зберігає дату народження співробітників; 5) «Passport»- зберігає серію та дані паспортів співробітників; 6) «PositionID»- зберігає номери посади співробітників.

3.19 – Створення таблиці «t\_employ»  

```
CREATE TABLE `t_employ` ( `ID` int(11) NOT NULL, `FirstName` varchar(50) NOT NULL, `LastName` varchar(50) NOT NULL, `Patronymic` varchar(50) NOT NULL, `DateOfBirth` date NOT NULL, `Passport` varchar(50) NOT NULL, `PositionID` int(11) NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

\* Таблиця «t\_contact» містить 4 основних осередки: «WorkPhone», «HomePhone», «eMail», «Address». 1) «WorkPhone»- зберігає робочі телефони співробітників; 2) «HomePhone»- зберігає домашні телефони співробітників; 3) «eMail»- зберігає електронні пошти співробітників; 4) «Address»- зберігає адреси співробітників.

Лістинг 3.20 – Створення таблиці «t\_contact»  

```
CREATE TABLE `t_contact` ( `ID` int(11) NOT NULL, `WorkPhone` varchar(20) DEFAULT NULL, `HomePhone` varchar(20) DEFAULT NULL, `eMail` varchar(50) DEFAULT NULL, `Address` varchar(200) DEFAULT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

Таблиця «t\_position» містить 1 основний осередок: «PositionName». Він зберігає назви посад співробітників.

Лістинг 3.21 – Створення таблиці «t\_position»  

```
CREATE TABLE `t_position` ( `ID` int(11) NOT NULL, `PositionName` varchar(50) NOT NULL ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

3.6 Програмування пристрою внесення даних Для програмування пристрою внесення даних було обрано використовувати інтегроване середовище розробки «Arduino IDE» через нескладність ним. В користі робочого програмного коду було обрано готовий варіант, що сконструйований за допомогою інструментів бібліотеки радіо ідентифікаційної мікросхеми MFRC522 мовою програмування C. Програмна бібліотека завантажена за допомогою відкритого посилання. В користі апаратної платформи було обрано платформу «Arduino», що побудована на мікроконтролері ATmega328. Назва проекту «ReadUidMultiReader». Його функція забезпечує зчитування даних ідентифікаційних даних транспондерів, що потрібно внести в таблицю бази даних. Після відправки зчитаних даних за допомогою послідовного порту на робочу машину, де розгорнуто сервер. Рисунок 3.22 – Зображення заголовку програми відправки даних транспондерів По-перше, підключаються заголовочні файли всіх функціональних бібліотек, що забезпечують роботу вмонтованих функцій. Бібліотека «MFRC522» та «SPI». Перш ніж користуватися командами спілкування з мікросхемою, друга ініціалізує налаштування роботи з SPI- протоколом.

Лістинг 3.22 – Додавання бібліотек на початку проекту  

```
#include <SPI.h>
#include <MFRC522.h>
```

У циклі Main ініціалізується мікросхема та процес передачі даних за UART-протоколом послідовного порту. Лістинг 3.23 – Ініціалізація послідовного порту  

```
Serial.begin(9600); while (!Serial); SPI.begin();
```

В безкінцевому циклі While перевіряється наявність піднесеного картки. Проходить алгоритм запобігання зіткненню карт, зчитується сектор, що зберігає ідентифікаційний номер транспондера. Після цього вимикається та чекає наступного піднесення. Лістинг 3.24 – Блок безкінцевого циклу  

```
if(mfrc522[reader].PICC_IsNewCardPresent() && mfrc522[reader].PICC_ReadCardSerial()) { Serial.print(F("In hex:")); dump_byte_array(mfrc522[reader].uid.uidByte, mfrc522[reader].uid.size); Serial.print(mfrc522[reader].PICC_HaltA()); mfrc522[reader].PCD_StopCrypto1();}
```

Передача унікального номеру ідентифікації відбувається у шістнадцятковому форматі чисел по чотири байти в масиві. Кожен байт може приймати значення від 0 до 15. Для цього створена окрема функція. Лістинг 3.25 – Функція прийому байтових даних і створення масиву  

```
dump_byte_array(byte *buffer, byte bufferSize) { for (byte i = 0; i < bufferSize; i++) { Serial.print(buffer[i] < 10 ? '0' + String(buffer[i]) : HEX);}
```

3.7 Програмування додатку роботи з даними співробітників Невід'ємною складовою структурно-закінченої системи контролю та управління доступом є гарно сконструйована панель керування внесення, видалення, зміна, корегування та експорт повинні створюватися без потреби втручання до змін програмного коду низькому рівні системи, адже метою встановлення подібних приладів є полегшення контролю та керування досить складними процесами.

Рисунок 3.28 – Зображення вікна керування даними співробітників Основою написання програми розгалуженій СКУД обрано те ж програмне забезпечення від компанії «Microsoft» що й у випадку програми частини- «Visual Studio». Програмування відбувається мовою C#. Особливості мови й середовища розробки в інших мовах мають перевагу. Велика кількість вбудованих інструментів для роботи із вводом та виводом даних інструменти, що дозволяють працювати із конструюванням GUI (Graphic User Interface)- форм графічного інтерфейсу, достатніх за складністю для задоволення усіх поставлених до проекту завдань. Основними потребами від додатку керування даними співробітників є зручність внесення та зміни ідентифікаційних даних та моніторингу даних логуються відбуваються на виконавчому пристрої системи. Додаток має містити список співробітників (ПІБ), номери «прив'язки»

транспондерів, дати народження, номери документів підтвердження особи, посади, контактні дані. Також, інте дозволяти отримувати контролюючому дані статистики за кожним співробітником, особу якого внесено до бази даних. Для реалізації проекту використовуються такі функціональні бібліотеки як: Лістинг 3.26 – Додавання бібліотек на початку проекту System; using System.Collections.Generic; using System.ComponentModel; using System.Data; using System.Linq; using System.Text; using System.Threading.Tasks; using System.Windows.Forms; using MySql.Data.MySqlClient; using System.IO; using System.IO.Ports; using iTextSharp.text; using iTextSharp.text.pdf. Клас System означає, що використовується бібліотека System у проекті. Це дає деякі корисні класи та функції, такі як Console або функція/метод WriteLine. Клас System.Collections.Generic забезпечує загальну реалізацію колекції даних, як-от пов'язані списки, стеки, черги та словники. Клас System.ComponentModel надає інструменти, які використовуються для реалізації поведінки компонентів під час виконання та проектування. Клас System.Data.SqlClient надає можливість реалізувати DataAdapter і представляє набір методів і властивостей, пов'язаних із дією, які використовуються для заповнення та оновлення набору даних та оновлення джерела даних. Клас System.Drawing надає функціональні можливості роботи з графічним інтерфейсом. Клас System.Linq надає різні типи класів і методи, які підтримують запити LINQ. Клас System.Text містить класи, які представляють кодування символів ASCII та Unicode. Абстрактні базові класи для перетворення блоків символів у блоки байтів та з них і допоміжний клас, який надає методи форматує об'єкти String без створення проміжних екземплярів String. Клас System.Threading.Tasks надає методи, які спрощують роботу з написанням паралельного та асинхронного коду. Клас MySql.Data.MySqlClient надає серію класів, які надають доступ до сервера MySQL. Весь зв'язок між програмою C# і сервером MySQL маршрутизується через об'єкт MySqlConnection, який надає можливість встановлення бібліотеки MySql.Data.MySqlClient, за допомогою якої програма Dot Net матиме змогу з'єднатися з сервером MySQL. Після встановлення бібліотеки з офіційного сайту MySQL, вона так само додається до проекту за допомогою функції using. Класи System.IO та System.IO.Ports – ці класи можна використовувати для читання та запису даних або потоки даних. Класи iTextSharp.text та TextSharp.text.pdf – це розширені бібліотеки інструментів, які використовуються для створення складних звітів у форматі PDF. Процес створення додатку полягає у будівництві конструктора додатку з окремих структурних елементів, таких як: поля, кнопки та вікна. Кожен елемент має свою окрему обробку, алгоритм роботи якої будується індивідуально за потребами, та панель налаштувань і властивостей. Рисунок 3.29 – Зображення панелі налаштування кнопки в програмі «Visual Studio 2022» Далі зображено у виді блок-схеми алгоритму роботи кнопки найважливіших функціональних елементів – кнопок (button). При натисканні кнопки викликається функція обробки, що виконує створений для неї алгоритм роботи. Алгоритм роботи кнопки підключення «bConnect» до бази даних «db\_acs» та відключення. Передача даних з послідовного порту відбувається в асинхронному режимі за допомогою метода «Handler». Рисунок 3.30 – Зображення алгоритму роботи методу підключення Алгоритм роботи кнопки : ідентифікаційного номеру з транспондеру за окремим співробітником. За допомогою цього алгоритму здійснюється виведення нового ідентифікаційного номеру співробітнику, зміна вже існуючого та видалення вже безпосередньо існуючого. Рисунок 3.31 – Зображення алгоритму роботи кнопки закріплення ідентифікаційного номеру з транспондеру Рисунок 3.32 – Зображення алгоритму роботи кнопки збереження даних співробітників Алгоритм роботи кнопки збереження даних співробітників за допомогою алгоритму є можливість додавання прізвищ імен та по-батькові, табельних номерів, дат народження, номерів документів підтвердження осіб. Додавання контактних даних є функцією, за допомогою якої є можливість додавання номерів телефонів та електронні адреси й адреси місця проживання співробітників. Для цього створено окреме вікно «контактні дані співробітника». Рисунок 3.33 – Зображення вікна додавання комунікаційних даних співробітників Рисунок 3.34 – Зображення алгоритму роботи кнопки переносу до панелі додавання комунікаційних даних співробітників За допомогою спеціального елемента «button» потрапляємо в окрему форму редагування даних. Обмін між двома формами відбувається за допомогою методів створеного класу «DataBank», що надає функції буферної зони для можливості передачі даних. 3.8 Технічне налагодження СКУД Щоб привести систему контролю управління доступом до робочого стану треба зробити налаштування вузлів та їх фізичне підключення у єдину структуру, що функціонуватиме одночасно та в той же час незалежно. Для цього треба прагнути зменшення кількості утручань у роботу системи, первинний запуск системи потребує налаштування, як програмного, так і апаратно- функціонального. Рисунок 3.35 – Зображення виконавчого пристрою СКУД із приладами зчитування до програматора ST-link v.2.0 треба підключити контролер (виконавчий пристрій) СКУД за допомогою порту периферії SWD. В програмному коді треба встановити актуальну IP-адресу машини, за допомогою якої розгорнуто сервіс маршрутизатора мережі та адресу контролера виконавчого пристрою. За допомогою цих даних розгалужені вузли зможуть отримати інформацію щодо правил підключення один до одного в суспільній мережі. Рисунок 3.36 – Зображення фрагменту програмного коду Знайшовши кнопку компіляції проекту та його завантаження імпортуємо проект, після чого мікроконтролер СКУД. Рисунок 3.37 – Зображення вікна завантаження програмного коду до мікроконтролера СКУД. Відкривши панель керування програмою «XAMPP», треба запуснути роботу бази даних «MySQL» та допоміжні

«Arache». За допомогою функціональної кнопки «Admin» розпочинається робота серверу «MySQL». Рисунок 3.39 – Зображення панелі програми «XAMPP» Наступна дія – розгортання консольного додатку, що оброблятиме потоки інформації від виконавчого пристрою та, навпаки, відправлятиме дані, оброблені додатком керування сервером та б до початкового вузла. Для цього запускається «.exe» файл проекту. Пишемо ідентифікаційні дані для з'єднання із б. Рисунок 3.39 – Зображення панелі консольного додатка-обробника серверної частини Щоб мати змогу вносити корегувати та отримувати дані з таблиць бази даних було створено графічний додаток «WinHuman». Розгортається за допомогою розширення додатку «.exe». Рисунок 3.40 – Зображення вікна роботи із співробітниками Із налаштуванням проекту закінчено. Робиться це одноразово, саме під час першого запуску системи. Апаратне потребує підключення контролеру СКУД до електричного живлення постійного току, напругою 12 В. Тестування пр електронним навантаженням на виході виконавчого пристрою замість електро-магнітної системи. Струм на виставляється біля 1000 Ма, що повністю емулює підключення реального навантаження. Для роботи при забезпечити достатній рівень струму, щоб можна було жити як електронне навантаження на виході контролера, частину контролера, без зменшення напруги, що її живить. Для того, щоб дізнатись потрібні технічні дані блоку скористаємось лабораторним джерелом, виставивши напругу на виході біля 12 В. Підключмо контролер лабораторного блоку живлення без задіяння електронного навантаження. Струм споживання контролера для своїх потреб 108 мА. За законом Ома, потужність рівняється добутку струму з напругою:  $W=U*I$ ,  $12*0.108=1.296$ , що дорівнює 1,3 Вт потужності. Спираючись на це- блок живлення треба обирати таких, який зможе забезпечити жив та навантаження, а це у купі близько 1.5 А при 12 В. Рисунок 3.41 – Зображення лабораторного джерела ж допомогою спеціалізованого кабелю Ethernet з'єднання із фізичними роз'ємами типу RJ45, підключаємо контрол маршрутизатора, вхід локальної (LAN- local area network) мережі маршрутизатора. Рисунок 3.42 – Зображ виходів RJ45 на мережевому маршрутизаторі Машину, за допомогою якої розгорнуто серверну частину теж треба п систему локальної мережі для забезпечення суспільного зв'язку на одному рівні. Фізичне підключення виконано готова до роботи. Щоб протестувати програмний алгоритм системи та коректність виконання команд, треба створеним інтерфейсом керування. Відкривається файл додатку «WinHuman.exe», за допомогою функції «Під'єдна під'єднується до бази даних та вносяться гіпотетичні дані співробітника, а саме: його ПІБ, паспортних даних, дати н посади та табельного номера. Перед цим вносяться список посад підприємства. Рисунок 3.43 – Зображення панел співробітників Після внесення даних особи, за допомогою функціональної кнопки «Підключити» система під'єд пристрою внесення даних. Пристрій встановлює з'єднання із машиною, за допомогою якої розгорнуто сервер, вико послідовний порт, отже треба підключити його за допомогою USB порту. Далі скануємо ідентифікаційну мітку-тз закріплюємо за особою співробітника, що додана. Паралельно, усі необхідні дані повинні бути автоматично внесені бази даних «db\_acs». Рисунок 3.44 – Зображення панелі ідентифікаційних номерів в програмі «phpMyAdmin» П внесений транспондер до зчитувача входу або виходу- отримуємо результат. Вихід контролера СКУД переходить д стану та видає напругу на підключене до цього моменту електронне навантаження. Дисплей навантаження, електричний магніт, демонструє наявність потрібних характеристик, що надає контролер, потрібних для роботи зам Рисунок 3.45 – Зображення показників електронного навантаження 3.9 Висновки за розділом Результатом виконани проектування апаратної та програмної частин системи контролю та управління доступом. Також, було створе керування внесенням, видаленням та корегуванням даних співробітників в систему бази даних. Наприкінці, було н та протестовано систему в робочому вигляді. ВИСНОВКИ У результаті виконання роботи дипломного проекту було предметну область систем контролю та керування доступом, види, методи та способи їх використання на практи актуальне положення систем контролю, рівень глибини, популярності та доцільності їх використання в сучас Установлено доцільність створення клієнт-серверної системи, спираючись та існуючі аналоги та потреби в охорони та слідкування складських приміщень. Розроблено прилад виконавчого пристрою на базі одно кристали додаванням фізичного контролеру безпечного мережевого зв'язку. Інтегровано систему контролю фізичного допомогою керування замком електромагнітного типу. Розроблено програмний алгоритм керування та з'єднанн частини, що була розгорнута за допомогою IBM-подібної машини, із вузлом виконавчого пристрою з метою можливості комунікації та швидкого обміну даними на великій дистанції. Розроблено графічний додаток панельно внесення й керування ідентифікаційними даними співробітників складського приміщення з використання розташованих в базі зберігання даних. ПЕРЕЛІК ПОСИЛАНЬ 1. Система контролю і управління доступом [Електрон / Режим доступу [www. URL: https://uk.wikipedia.org/wiki/Система\\_контролю\\_та\\_управління\\_доступом](https://uk.wikipedia.org/wiki/Система_контролю_та_управління_доступом) - Системи контролю та управління доступом [Електронний ресурс] / Режим доступу [www. URL: https://www.shop.com/catalog/kontrol\\_dostupa\\_20436/](https://www.shop.com/catalog/kontrol_dostupa_20436/) 3. СКУД. Системи контролю доступу [Електронний ресурс] / Реж [www. URL: https://www.aktivsb.ru/kontrol\\_dostupa/](https://www.aktivsb.ru/kontrol_dostupa/) 4. Access Control [Електронний ресурс] / Режим доступу

<https://www.techtarget.com/searchsecurity/definition/access-control> 5. What are the benefits of Access Control Systems? [Електронний ресурс] / Режим доступу [www. URL: https://cie-group.com/how-to-av/vi/blogs/access-control-systems](http://www.cie-group.com/how-to-av/vi/blogs/access-control-systems) 6. What is access control? [Електронний ресурс] / Режим доступу <https://www.fortinet.com/resources/cyberglossary/access-control> 7. Мікроконтролери початок. [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=\\_YfayPCMLMQ&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=_YfayPCMLMQ&ab_channel=VladimirMedintsev) 8. Вивчаємо STM 32 [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=vZVwvEeHkI&t=2s&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=vZVwvEeHkI&t=2s&ab_channel=VladimirMedintsev) 9. STM 32 Доступні операційні системи. [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=Da3-xi9GQcw&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=Da3-xi9GQcw&ab_channel=VladimirMedintsev) 10. Структура коду в STM32CubeIDE [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=I9c2XNeuqJU&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=I9c2XNeuqJU&ab_channel=VladimirMedintsev) 11. STM32 + RC522 Як зберігати та копіювати конфігурацію карти працюють картки NXP MIFARE [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=kwCaREpBE&t=1160s&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=kwCaREpBE&t=1160s&ab_channel=VladimirMedintsev) 12. Інтерфейс SPI з прикладу STM32. Підключення дисплея та пам'яті [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=6ruS8eR\\_Rdc&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=6ruS8eR_Rdc&ab_channel=VladimirMedintsev) 13. STM32 + Ethernet / Як вибрати Ethernet контролер / Набір бібліотеки LWIP / LAN8720 / Частина 1. [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=ww-2mkR8xlG&t=3s&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=ww-2mkR8xlG&t=3s&ab_channel=VladimirMedintsev) 14. STM32 + Ethernet / Чіп W5500 від WIZnet / Сокет розповідь про побудову WEB сервера. Частина 2. [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=PTDoDJadyi0&t=1172s&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=PTDoDJadyi0&t=1172s&ab_channel=VladimirMedintsev) 15. Програмування STM32. Урок 91. LAN. W5500. HTTP Server. Частина 1 [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=W85Zaysv2PY&ab\\_channel=narodstream](https://www.youtube.com/watch?v=W85Zaysv2PY&ab_channel=narodstream) 16. Ethernet для МК. WIZnet ENC28J60. MQTT [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=LwDDEIh63cA&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=LwDDEIh63cA&ab_channel=VladimirMedintsev) 17. Від ідеї до готового пристрою. З чого починати опрацювання ідеї майбутнього електронного пристрою? [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=Ib5s&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=Ib5s&ab_channel=VladimirMedintsev) 18. Datasheet STM32F103 [Електронний ресурс] / Режим доступу <https://www.st.com/resource/en/datasheet/stm32f103c8.pdf> 19. IPRAW. [Електронний ресурс] / Режим доступу <https://docs.wiznet.io/Product/iEthernet/W5500/Application/ipraw/> 20. Datasheet W5500 [Електронний ресурс] / Режим доступу [www. URL: https://docs.wiznet.io/Product/iEthernet/W5500/datasheet/](https://docs.wiznet.io/Product/iEthernet/W5500/datasheet/) 21. Limitation Note [Електронний ресурс] / Режим доступу [www. URL: https://docs.wiznet.io/assets/files/limitation\\_note\\_\\_arp\\_problem\\_in\\_the\\_nlb\\_environment\\_-\\_english\\_03b8d367a148e870c05217389964f76f15.pdf](https://docs.wiznet.io/assets/files/limitation_note__arp_problem_in_the_nlb_environment_-_english_03b8d367a148e870c05217389964f76f15.pdf) 22. Захист схем від переполюсування живлення за допомогою N-MOSFET [Електронний ресурс] / Режим доступу [www. URL: https://myelectronics.ru/mosfet-reversed-prot/](https://myelectronics.ru/mosfet-reversed-prot/) 23. Вибір мікроконтролера разом [Електронний ресурс] / Режим доступу [www. URL: https://habr.com/ru/post/481478/](https://habr.com/ru/post/481478/) 24. Ізольовані прийомопередавачі SPI [Електронний ресурс] / Режим доступу [www. URL: https://www.analog.com/russian/category/spisolator.html](https://www.analog.com/russian/category/spisolator.html) 25. MAX14850AEE+ Datasheet [Електронний ресурс] / Режим доступу [www. URL: https://www.alldatasheet.com/datasheetpdf/pdf/1366756/MAXIM/MAX14850AEE+.html](https://www.alldatasheet.com/datasheetpdf/pdf/1366756/MAXIM/MAX14850AEE+.html) 26. K3918 Datasheet [Електронний ресурс] / Режим доступу [www. URL: https://datasheetpdf.com/datasheet/K3918HR911105A](https://datasheetpdf.com/datasheet/K3918HR911105A) 27. Datasheet [Електронний ресурс] / Режим доступу [www. URL: https://datasheet.lcsc.com/szlcsc/Z1HanRun-Elec-HR911105A\\_C12074.pdf](https://datasheet.lcsc.com/szlcsc/Z1HanRun-Elec-HR911105A_C12074.pdf) 28. STM32 + CMSIS + STM32CubeIDE [Електронний ресурс] / Режим доступу [www. URL: https://habr.com/ru/post/481478/](https://habr.com/ru/post/481478/) 29. CH340 Datasheet [Електронний ресурс] / Режим доступу [www. URL: https://html.alldatasheet.com/htmlpdf/1132602/WCH/CH340/60/1/CH340.html](https://html.alldatasheet.com/htmlpdf/1132602/WCH/CH340/60/1/CH340.html) 30. EM4100 Protocol [Електронний ресурс] / Режим доступу [www. URL: https://www.priority1design.com.au/em4100\\_protocol.html](https://www.priority1design.com.au/em4100_protocol.html) 31. Marine - огляд стандарту, як правильно вибрати карти, брелоки, браслети та зчитувачі? [Електронний ресурс] / Режим доступу [www. URL: https://securityrussia.com/blog/em\\_marine.html](https://securityrussia.com/blog/em_marine.html) 32. Біометрія: що це і як вона змінює світ [Електронний ресурс] / Режим доступу [www. URL: https://fincult.info/article/biometriya-cto-eto-i-kak-ona-mir-finansov/](https://fincult.info/article/biometriya-cto-eto-i-kak-ona-mir-finansov/) 33. AMS1117-3.3 Datasheet [Електронний ресурс] / Режим доступу [www. URL: https://www.alldatasheet.com/datasheet-pdf/pdf/205691/ADMOS/AMS1117-3.3.html](https://www.alldatasheet.com/datasheet-pdf/pdf/205691/ADMOS/AMS1117-3.3.html) 34. Уроки C# .NET Windows Forms / #3 - Підключення MySQL та створення бази даних [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=pNTQy48Pafc&list=PL0IO\\_mIqDDFWOMqSKFaLypANf1W7-o87q&index=10&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=pNTQy48Pafc&list=PL0IO_mIqDDFWOMqSKFaLypANf1W7-o87q&index=10&ab_channel=VladimirMedintsev) 35. Уроки C# .NET Windows Forms / #3 - Підключення MySQL та створення бази даних [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=pNTQy48Pafc&list=PL0IO\\_mIqDDFWOMqSKFaLypANf1W7-o87q&index=10&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=pNTQy48Pafc&list=PL0IO_mIqDDFWOMqSKFaLypANf1W7-o87q&index=10&ab_channel=VladimirMedintsev) 36. Сокети (socket), серверна взаємодія за протоколами TCP та UDP у C# - Вчимо Шарп #18 [Електронний ресурс] / Режим доступу [www. URL: https://www.youtube.com/watch?v=pNTQy48Pafc&list=PL0IO\\_mIqDDFWOMqSKFaLypANf1W7-o87q&index=10&ab\\_channel=VladimirMedintsev](https://www.youtube.com/watch?v=pNTQy48Pafc&list=PL0IO_mIqDDFWOMqSKFaLypANf1W7-o87q&index=10&ab_channel=VladimirMedintsev)

[https://www.youtube.com/watch?v=ZRGgBtUgJKE&ab\\_channel=CODEBLOG](https://www.youtube.com/watch?v=ZRGgBtUgJKE&ab_channel=CODEBLOG) 37. Уроки C# .NET Windows For Авторизація користувача через базу даних [Електронний ресурс] / Режим доступу w [https://www.youtube.com/watch?v=eLQAgHexThM&ab\\_channel](https://www.youtube.com/watch?v=eLQAgHexThM&ab_channel) 38. B1205S-2W Datasheet [Електронний Режим доступу www. URL: [https://www.alldatasheet.com/datasheetpdf/pdf/274840/MORNSUN/B1205gclid=CjwKCAjwjtOTBhAvEiwASG4bCPIWgfavKWT2Uvn8aA0JMAQyEWST56V5yg\\_gvYaheTL7BikvKPLBNhoCTfw](https://www.alldatasheet.com/datasheetpdf/pdf/274840/MORNSUN/B1205gclid=CjwKCAjwjtOTBhAvEiwASG4bCPIWgfavKWT2Uvn8aA0JMAQyEWST56V5yg_gvYaheTL7BikvKPLBNhoCTfw) 39. Авторизація користувачів у базі даних. [Електронний ресурс]. Режим доступу: w <https://www.youtube.com/watch?v=eLQAgHexThM/> 40. Переваги мови C#. [Електронний ресурс]. – Режим дос URL: <https://forum.itvdn.com/t/urok-1-preimushhestva-yazyka-c/2941/> 41. Протоколи TCP та UDP. [Електрон – Режим доступу: www. URL: [https://professorweb.ru/my/csharp/web/level1/1\\_4.php/](https://professorweb.ru/my/csharp/web/level1/1_4.php/) 42. Класи та об'єкти. [Е ресурс]. – Режим доступу: www. URL: <https://metanit.com/sharp/tutorial/3.1.php> 43. Сокети та клієнт-серверне [Електронний ресурс]. – Режим доступу: www. URL: [https://www.youtube.com/watch?v=ZRGgBtUgJKE&ab\\_channel=CODEBLOG/](https://www.youtube.com/watch?v=ZRGgBtUgJKE&ab_channel=CODEBLOG/) 44. Пам'ятка по SQL. [Електронний ресурс]. – Режим доступу: <https://habr.com/ru/post/564390/>

## Source

**Запорізький інститут економіки та інформаційних ...Підрозділи - Запорізь...**

0.33%

Запорізький інститут економіки та інформаційних ...Підрозділи - Запорізький інститут економіки та ...

<https://www.zieit.edu.ua/>

**З а в д а н н я на курсову роботу**

0.17%

З а в д а н н я на курсову роботу

<https://studfile.net/preview/3021665/page:5/>

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕ...**

0.17%

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ ТС  
– технологічна система; ТП – технологічний процес;

<https://radio.bobrodobro.ru/32025>

**бу АЮ Слободчук · 2019 — комплекс технічних та програмних засобів без...**

0.17%

бу АЮ Слободчук · 2019 — комплекс технічних та програмних засобів безпеки, що здійснює регулювання входу / виходу та переміщень людей чи транспортних об'єктів на територіях, які ...

[http://repository.kpi.kharkov.ua/bitstream/KhPI-Press/49375/1/vestnik\\_KhPI\\_2019\\_01\\_ETPT\\_Slobodchuk\\_Zastosuvannia.pdf](http://repository.kpi.kharkov.ua/bitstream/KhPI-Press/49375/1/vestnik_KhPI_2019_01_ETPT_Slobodchuk_Zastosuvannia.pdf)

Системи контролю та управління доступом (СКУД) ► В ...

0.17%

Системи контролю та управління доступом (СКУД) ► В ...

<https://sheriff.com.ua/uk/uslugi/sistemi-kontrolia-dostupa-2/>

by EE Молнар · 2019 — мікроконтролера. Від розробника прихована велик...

0.17%

by EE Молнар · 2019 — мікроконтролера. Від розробника прихована велика частина другорядних функцій, що сильно розвантажує інтерфейс і робить управління інтуїтивно зрозумілим.

[https://ela.kpi.ua/bitstream/123456789/30310/3/Molnar\\_bakalavr.pdf](https://ela.kpi.ua/bitstream/123456789/30310/3/Molnar_bakalavr.pdf)

by IO Шевцов · 2019 — мікроконтролерів сімейства STM32, що дозволяє ге...

0.17%

by IO Шевцов · 2019 — мікроконтролерів сімейства STM32, що дозволяє генерувати код на основі мови. Сі, використовуючи для цього графічний помічник [36].

[https://openarchive.nure.ua/bitstream/document/11913/1/2019\\_M\\_RTIKS\\_Shevtsov\\_IO.pdf](https://openarchive.nure.ua/bitstream/document/11913/1/2019_M_RTIKS_Shevtsov_IO.pdf)

by KE Веласко · 2021 — На сьогоднішній день хатрр є однією з найкращи...

0.17%

by KE Веласко · 2021 — На сьогоднішній день хатрр є однією з найкращих збірок веб-сервера, за допомогою цієї збірки ви зможете швидко розгорнути на своєму комп'ютері повноцінний і ...

[http://eir.zntu.edu.ua/bitstream/123456789/7579/1/MR\\_Velasco.pdf](http://eir.zntu.edu.ua/bitstream/123456789/7579/1/MR_Velasco.pdf)

Довідник хостингових термінів - TutHost

0.17%

<https://tuthost.ua/uk/faq/spisok-khostingovykh-terminov/>

Сокет (програмний інтерфейс) - Вікіпедія

0.17%

[https://uk.wikipedia.org/wiki/%D0%A1%D0%BE%D0%BA%D0%B5%D1%82\\_\(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B8%D0%B9\\_%D1%96%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81\)](https://uk.wikipedia.org/wiki/%D0%A1%D0%BE%D0%BA%D0%B5%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B8%D0%B9_%D1%96%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81))



