

НУБІП України

НУБІП України

НУБІП України

МАГІСТЕРСЬКА РОБОТА

15.04 – МР. 1578 “С” 2020.10.23.015 ПЗ

ГАМАЮНОВА НІКІТИ ДМИТРОВИЧА

2021 р.

НУБІП України

НУБІП України

НУБІП України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

УДК 004.78:004.056.53

НОУБІП УКРАЇНИ

ПОГОДЖЕНО

Декан факультету

Інформаційних технологій

ДОПУСКАЄТЬСЯ ДО ЗАХИСТУ

Завідувач кафедри

Комп'ютерних систем, мереж та кібербезпеки

Глазунова О.Г., д.пед.н, проф.

підпис

ПІБ, вчене звання і ступінь

Лахно В.А., д.т.н., проф.

підпис

ПІБ, вчене звання і ступінь

НОУБІП УКРАЇНИ

« » 2021 р.

« » 2021 р.

НОУБІП УКРАЇНИ

МАГІСТЕРСЬКА РОБОТА

На тему: «Дослідження комплексної системи контролю доступу до приміщення на базі МК ESP32 та STM32. Апаратна частина»

Спеціальність 123 «Комп'ютерна інженерія»

Освітня програма Комп'ютерні системи та мережі

Орієнтація освітньої програми

НОУБІП УКРАЇНИ

НОУБІП УКРАЇНИ

Керівник магістерської роботи:

Лахно В.А.

підпис

ПІБ

Виконав:

Гамаюнов Н.Д.

підпис

ПІБ

НОУБІП УКРАЇНИ

КИЇВ-2021

«ЗАТВЕРДЖУЮ»

завідувач кафедри

комп'ютерних систем, мереж та кібербезпеки

/ Лахно В.А., д.т.н., проф. /

підпис

ПІВ. вчене звання і ступінь

» 20 р.

ЗАВДАННЯ

ДО ВИКОНАННЯ МАГІСТЕРСЬКОЇ РОБОТИ СТУДЕНТУ

Гамаюнову Нікіті Дмитровичу

(прізвище, ім'я, по батькові)

Спеціальність (напрямок підготовки): комп'ютерна інженерія _____

Освітня програма: комп'ютерні системи та мережі _____

Орієнтація освітньої програми: _____

Тема магістерської роботи: «Дослідження комплексної системи контролю доступу до приміщення на базі МК ESP32 та STM32. Апаратна частина»

затверджена наказом ректора НУБіП України від « 23 » жовтня 2020 р. № 1578 «С»

Термін подання завершеної роботи на кафедру: 20 листопада 2021 р.

Вихідні дані до магістерської роботи: програмна частина на мові програмування C++,

можливість в'єднання з серверним додатком, бібліотека Arduino, мікроконтролер ESP32,

кількість карток, які зберігаються на пристрої – до 1000.

Перелік питань, що підлягають дослідженню:

1. Аналітичний огляд

2. Проектування системи

3. Розробка програмного забезпечення для мікроконтролерів

Перелік графічного матеріалу (за потреби): принципи схеми вспоміжного та головного

пристроїв, схеми друкованих плат для вспоміжного та головного пристроїв

Дата видачі завдання « 23 » жовтня 2020 р.

Керівник магістерської роботи

(підпис)

Лахно В.А., д.т.н., проф.

(прізвище та ініціали)

Завдання прийняв до виконання

(підпис)

Гамаюнов Н.Д.

(прізвище та ініціали студента)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Постановка задачі магістерської роботи	23.10.2020	Виконано
2	Аналіз предметної області	12.11.2020	Виконано
3	Проектування системи	23.11.2020	Виконано
4	Реалізація системи	15.02.2021	Виконано
5	Тестування системи	10.05.2021	Виконано
6	Оформлення пояснювальної записки	01.08.2021	Виконано
7	Оформлення графічного матеріалу	10.10.2021	Виконано

Студент

Гамаюнов Н.Д.
(підпис) (ініціали та прізвище)

Керівник роботи

Ляхов В.А.
(підпис) (ініціали та прізвище)

НУБІП України

РЕФЕРАТ

Пояснювальна записка: 68 сторінки, 24 рисунки, 3 таблиці, 4 додатка, 21

НУБІП України

джерело

КОМП'ЮТЕРНА СИСТЕМА, БЕЗПЕКА, КОНТРОЛЬ ДОСТУПУ,
МІКРОКОНТРОЛЕР, ПРОЕКТУВАННЯ, ARDUINO, ESP32, SERBER, C++,
HTTPS, КОНТРОЛЛЕР, СЕРВІС

НУБІП України

Об'єкт розробки – автоматизована система контролю доступу до приміщення.

Мета роботи – побудова системи, яка автоматично веде контроль доступу користувачами до певних приміщень на підприємстві з використанням клієнт-серверних технологій та сучасних фреймворків для реалізації веб-сервісу.

НУБІП України

Предмет – методи та програмні додатки для контролю доступу до приміщення.

Робота складається з трьох розділів.

НУБІП України

Перший розділ присвячено аналізу предметної області. Проведено детальний огляд об'єкта, описано обрану мову програмування та її аналіз.

У другому розділі розкриті питання щодо проектування системи, а саме аналіз та моделювання системи, підбір компонентів та проектування друкованих плат.

НУБІП України

Третій розділ присвячено розробці програмного забезпечення системи. Проведено моделювання поведінки та структури системи та налаштовано роботу серверної частини.

НУБІП України

В результаті виконання дипломної роботи проведено аналіз, моделювання та дослідження розробленої автоматизованої системи та розроблені рекомендації щодо її проектування

НУБІП України

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ 2

НУБІП України

ВСТУП 4

1 АНАЛІТИЧНИЙ ОГЛЯД 6

1.1 Дослідження предметної області 6

1.2 Призначення системи 9

НУБІП України

1.3 Огляд існуючих рішень 10

2 ПРОЕКТУВАННЯ СИСТЕМИ 16

2.1 Аналіз та моделювання системи 16

2.2 Підбір компонентів системи 20

НУБІП України

2.3 Проектування друкованих плат 30

2.4 Монтаж компонентів на друковану плату 38

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ
МІКРОКОНТРОЛЕРІВ 41

НУБІП України

3.1 Моделювання поведінки системи 42

3.2 Моделювання структури системи 45

3.3 Розгортання та налаштування системи 47

НУБІП України

3.4 Тестування працездатності системи 53

ВИСНОВКИ 58

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 59

ДОДАТКИ 61

НУБІП України

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

НУБІП України

СКУД – системи контролю та управління доступом

ПК – персональний комп'ютер

ПЗ – програмне забезпечення

PoE – Power over Ethernet

SDK – software Development Kit

ERP – система планування ресурсів підприємства, Enterprise Resource Planning System

RFID – радіочастотна ідентифікація, Radio frequency identification

ПАК – програмно апаратний комплекс

NRND – не рекомендовано для нових конструкцій, Not Recommended for New Designs

EOL – кінець терміну служби, End-of-life product

Usb – універсальна послідовна шина, Universal Serial Bus

АС – автоматизована система

xml – розширювана мова розмітки, Extensible Markup Language

QR – двовимірний штрих-код, quick response

front-end – інтерфейс для взаємодії між користувачем і back end

PHP – скриптова мова програмування, Hypertext Preprocessor

CSS – каскадні таблиці стилів, Cascading Style Sheets

HTML – мова розмітки гіпертексту, HyperText Markup Language

JS – JavaScript

MBSE – системна інженерія на основі моделей, model based systems engineering

СКБД – система керування базами даних

БД – база даних

IOS – мобільна операційна система, iPhone OS

НУБІП України

НУБІП України
IEEE інститут інженерів з електротехніки та електроніки,
Institute of Electrical and Electronics Engineers

UML – уніфікована мова моделювання, Unified Modeling Language

SysML – мова моделювання загального призначення для

застосувань в системній інженерії, Systems Modeling

Language

НУБІП України
ORM – об'єктно-реляційна проєкція, object-relational mapping

HTTP – протокол передачі гіпер-текстових документів, Hyper Text

Transfer Protocol

НУБІП України
API – програмний інтерфейс застосунку, Application Programing

Interface

URL – уніфікований локатор ресурсів, Uniform Resource Locator

WiFi – бездротова правдивість відтворення, Wireless Fidelity

НУБІП України
IDE – інтегроване середовище розробки, Integrated Development

Environment

НУБІП України

НУБІП України

НУБІП України

НУБІП України

ВСТУП

В даний час автоматизовані системи контролю та управління доступом (СКУД) є складовою інформаційної інфраструктури та невіддільним елементом системи безпеки сучасного підприємства.

У загальному випадку під системою контролю та управління доступом розуміють сукупність програмно-технічних та організаційно-методичних засобів, за допомогою яких вирішується завдання контролю та управління приміщеннями підприємства, а також оперативний контроль за пересуванням персоналу та часом його перебування на території підприємства. За допомогою системи контролю та управління доступом, підприємство має можливість підняти рівень загальної безпеки, а також скоротити витрати на її забезпечення.

У фізичній безпеці термін «контроль доступу» відноситься до практик обмеження доступу до власності, до будівель або приміщень, доступ до яких дозволено лише для авторизованих людей. Фізичний контроль доступу, можливо, досягнуто шляхом використання людини (охоронець, вибивала або службовець у приймальні), через механічні способи, такі як замки та ключі на двері, або через технологічні засоби, такі як системи доступу, засновані на картах доступу або біометричній ідентифікації.

Очевидно, ще для того, щоб забезпечити безпеку домашньої інформаційної системи, необхідно принаймні замикати двері на ключ перед відходом. У більшості компаній, які використовують інформаційні системи для роботи, на вході, якщо не в робоче приміщення, то в будівлю знаходиться спеціально навчена людина з написом «охорона» або «служба безпеки» на бейджі або на спині, яка зазвичай вимагає пред'явити пропуск тих, хто працює в цій будівлі і записує в журнал усіх, хто не працює в цій будівлі/приміщенні, але з якихось ділових питань має пройти всередину. За легендою робить він це для того, щоб можна було у разі втрати майна швидко знайти того, хто міг це зробити зі сторонніх. Як правило, сучасні офісні приміщення мають камери

НУБІП УКРАЇНИ
відеоспостереження, тому в разі неправомірних дій відвідувачів їх можна буде легко ідентифікувати. В ідеалі біля кожних дверей у приміщення повинен сидіти спеціально навчений охоронець, звертати перепустки та записувати в журнал хто, коли входив і виходив.

НУБІП УКРАЇНИ
Крім охоронця, двері в приміщення потрібно закривати на замки. Якщо із замками все більш-менш зрозуміло, то з охоронцями, які акуратно фіксують у журнал кожне відчинення дверей, зазвичай проблеми навіть не встигають виникнути, тому що надто не вигідно утримувати таку армію людей не зайнятих у безпосередньому робочому процесі компанії, який приносить основну прибуток.

НУБІП УКРАЇНИ
Для того щоб зменшити витрати на охоронному персоналі та підвищити рівень фізичної безпеки застосовують системи контролю та управління доступом (у будівлі та приміщення).

НУБІП УКРАЇНИ
Для того, щоб система чітко та безперебійно виконувала свої функції, вирішувала всі поставлені завдання та забезпечувала максимальний економічний ефект від своєї експлуатації, вона має бути спроектована з врахуванням усіх можливих сценаріїв взаємодії з користувачем, якісно змонтована, інтегрована з іншими системами безпеки.

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

1 АНАЛІТИЧНИЙ ОГЛЯД

1.1 Дослідження предметної області

Системи контролю та управління доступом для багатьох стали звичним атрибутом повсякденного життя. Люди навіть не замислюються часом, як ця система працює, чого від неї можна очікувати, крім тривіального відчинення дверей, і чим може бути обумовлений той чи інший функціонал конкретної системи.

Залежно від завдань, які має вирішувати СКУД, варіюється її топологія, характеристики окремих компонентів і, звичайно ж, вартість системи. Для офісу маленької фірмочки з кількох людей, що вміщаються в одній кімнаті, досить поставити простий контролер (можливо, конструктивно поєднаний зі зчитувачем карт), і завдання відсікти непроханих гостей вже вирішено. Можна контролер з'єднати з комп'ютером, і тоді з'являється додатковий функціонал: можливість збереження та подальшого аналізу інформації про всі події (звичайно, в першу чергу про проходи). Але і в цьому випадку можна говорити лише про автономну СКУД, незважаючи на наявність зв'язку з ПК.

Описаний вище приклад з'єднання автономного контролера з ПК вже підпадає під поняття централізованої, тобто мережевої системи, а з урахуванням можливості роботи і без ПК – під поняття універсальної, але знову ж таки мережевої системи. Зовсім коротко визначення мережевий СКУД можна було б сформулювати приблизно так: "Мережевий є СКУД, що містить як апаратні, так і програмні компоненти з можливістю їх масштабування, що забезпечує можливість централізованого управління та спостереження".

Звичайно ж, визначення теж не всеосяжне, але дуже важливою є характеристика масштабованості системи, тобто можливість її нарощування як під час проектування, так і протягом життєвого циклу.

Другим у визначенні стоїть використання в комплексі апаратних (контролери доступу, що управляють точками проходу) та програмних (СКУД, що працює, як правило, на ПК) засобів.

Третя характеристика – це можливість централізованого управління та спостереження, тобто для мережного СКУД дана характеристика не залежить від розмірів системи – можливість залишається завжди.

На початку ХХІ століття над ринком технічних засобів забезпечення безпеки з'явилось дуже багато систем контролю та управління доступом (СКУД).

ГОСТ 51241-2008 "Засоби і системи контролю і управління доступом.

Класифікація. Загальні технічні вимоги. Методи випробувань", дає таке визначення: система контролю та управління доступом (СКУД) – сукупність засобів контролю та управління, що мають технічну, інформаційну, програмну

та експлуатаційну сумісність. Під контролем та управлінням доступом у ГОСТі

розуміється таке: комплекс заходів, спрямованих на обмеження та санкціонування доступу людей, транспорту та інших об'єктів до (з) приміщення, будівлі, зони та території. [1]

Вищезгаданий нормативно-технічний документ дає узагальнене визначення, тому, відповідаючи на різні питання та обговорюючи технічні

характеристики СКУД, часто стикаєшся з тим, що наприкінці розмови у людини виникає питання: а навіщо взагалі все це треба? Щоб розібратися, треба розглянути наочний приклад.

Візьмемо невелику компанію, що складається з кількох відділів: відділу продажу, дирекції, секретаріату, відділу комплектації, складу, бухгалтерії та серверної.

У компанію ведуть дві вуличні двері: головний вхід (вхід з вулиці) та додатковий (вхід із двору) та 6 внутрішніх дверей, контрольованих СКУД. Два

вуличні входи, входи на склад та у відділ комплектації, вхід у бухгалтерію та вхід до каси контролюються на прохід в обидві сторони (на вхід та на вихід), входи в

дирекцію та серверну контролюються лише в одну сторону (тільки на вхід).

Організація доступу в робочий час може виглядати так: усі співробітники мають право ходити через головний та додатковий входи; склад мають право доступу

лише люди, безпосередньо пов'язані з комплектацією і відвантаженням (завідувач складом, комірник тощо), до бухгалтерії - лише бухгалтери, до каси - лише касир. Прибиральниця має доступ до всіх приміщень, крім серверної, але тільки з 8.00 до 9.00. Привілейована група користувачів (директора, начальник служби охорони тощо) мають право ходити скрізь та у будь-який час.

Люди, наділені певними однаковими повноваженнями, становлять певну групу доступу. Група доступу характеризується певними точками проходу, через які цій групі доступ дозволено, і тимчасовим профілем, що складається з декількох часових інтервалів, протягом яких дозволена можливість проходу через дозволені точки (двері, турнікет, шлагбаум). Точка проходу може бути двосторонньою - коли прохід в обидві сторони здійснюється по картці, або односторонній - коли вхід здійснюється за пред'явленням картки, а вихід натисканням кнопки.

Абсолютно всі події, що відбуваються в системі, протокуються і (за потреби) з'являються на моніторі у охоронця (можна охоронцеві виводити лише тривожні події). За сучасних обсягів жорстких дисків ці протоколи можуть зберігатися необмежену кількість часу. Стандартна функція будь-якого програмного забезпечення СКУД - перегляд протоколів за різними критеріями, що дозволяє відтворити будь-яку подію, що нещодавно, так і давно минула, переглянути всі тривожні події за минулий місяць, з'ясувати, де вчора ходив завідувач складу. Наприкінці дня можна подивитися, хто залишився в офісі після 18.00, або здійснити швидкий пошук співробітника в робочий час. Можна контролювати дії охорони, якщо, наприклад, охоронець повинен щогодини обходити всі приміщення та відзначатись на всіх зчитувачах. Можна сміливо сказати, що можливості різних СКУД обмежуються лише фантазією проектувальників, а за великим рахунком, можна реалізувати будь-яку ідею.

НУБІП України

1.2 Призначення системи

НУБІП України

Сучасні системи контролю та управління доступом ефективно вирішують завдання безпеки будь-якого рівня, здійснюють попередження про проникнення сторонніх осіб на підконтрольну територію, а також сприяють підвищенню дисципліни праці завдяки обліку робочого часу співробітників компанії. Високий рівень безпеки може досягатися дублюванням ідентифікації.

НУБІП України

Наприклад, додатково для перевірки електронного ключа відвідувача може застосовуватися введення коду доступу з клавіатури або ідентифікація голосу. У сучасних комплексах СКУД є підсистема "тихої тривоги", яка запроваджується у разі будь-якої загрози і повідомляє про неї службу безпеки.

НУБІП України

За допомогою СКУД може здійснюватись управління будь-якими дверима, що є частинами системи. Можливе стеження за дверима максимально дозведеної тривалості її перебування у відкритому стані. Після максимально допустимого часу виникає сигнал тривоги (він може бути різним: звуковий сигнал безпосередньо біля проблемних дверей, повідомлення на пульт чергового та інші варіанти). Під керуванням операційною системою замикаючі механізми можуть

НУБІП України

відкриватися та закриватися у певні періоди часу. Для всіх необхідних шнурів живлення та підключення до локальної мережі може бути використана структурована кабельна система або прокладання одиночних кабелів.

НУБІП України

В основі завдань, що вирішуються системою контролю та управління доступом, лежить можливість розмежування повноважень персоналу за часом та точками доступу.

НУБІП України

Таким чином, типовими завданнями, що покладуються на СКУД є:

- розподіл прав доступу між клієнтами;
- протоколювання фактів проходу через контрольні точки;
- робота з перепустками та їх ідентифікація;
- автоматичний облік робочого часу;
- централізоване управління системою контролю доступу.

НУБІП України

Крім усього іншого, можна виконати інтеграцію СКУД із системами відеоконтролю та охоронно-пожежної сигналізації для підвищення рівня безпеки в цілому, що дозволить понерелжати власника системи про будь-які позаштатні ситуації.

НУБІП України

1.3 Огляд існуючих рішень

У сучасних СКУД зв'язок між контролерами, робочими місцями користувачів та сервером системи здійснюється через мережу Ethernet.

Інтерфейс Ethernet забезпечує високу надійність роботи системи за рахунок застосування типових ІТ-рішень та роботи всіх пристроїв системи в єдиному адресному просторі за єдиним протоколом. Ethernet також дає можливість використання технології PoE (Power over Ethernet) – привабливого альтернативного способу електроживлення мережевих пристроїв, що суттєво полегшує монтаж обладнання СКУД.

Всі контролери та ПК системи працюють у єдиному інформаційному середовищі з єдиною базою даних, встановленою на сервері системи. При цьому наявність постійного зв'язку контролерів із БД не потрібна.

В енергонезалежну пам'ять контролерів передаються всі необхідні права доступу, там же зберігаються події, що реєструються. Для контролерів можна встановити алгоритм роботи, що дозволяє системі тривалий час працювати в автономному режимі без підключення до сервера.

Наприклад, Global antipass підтримується за допомогою таблиць маршрутизації, що передаються всім контролерам системи. Такі установки дозволяють організувати складні рішення контролю доступу з урахуванням зональності. При відключенні електроживлення таблиця маршрутизації залишається у пам'яті контролера без підтримки сервера, і система залишається

повністю функціональної. При відновленні зв'язку з сервером системи події переносяться до БД [2].

Для забезпечення автономної роботи контролера важливою характеристикою обсяг пам'яті. Наприклад, сучасні контролери можуть зберігати в пам'яті дані про 150 000 подій та 50 000 користувачів.

Розширення системи не супроводжується заміною наявних пристроїв – досить просто включити нове обладнання до мережі Ethernet. Висока швидкість передачі даних та паралельна робота всіх контролерів дозволяють будувати системи безпеки без обмеження за кількістю контролерів, у тому числі розташованих у різних будинках, районах міста та у різних містах. Одночасна обробка багатьох подій забезпечує коректну роботу системи в моменти одночасного спрацювання декількох пристроїв.

Можливість підключення до контролера контролерів другого рівня полегшує розширення системи, що особливо актуально для великих підприємств. Для зв'язку контролера з контролерами другого рівня може застосовуватися інтерфейс зв'язку RS-485, що дозволяє значно оптимізувати витрати на розширення системи.

Важливим параметром при виборі контролерів СКУД є кількість керованих виконавчих пристроїв. Універсальні контролери в залежності від налаштувань можуть керувати турнікетами, шлагбаумами чи замками. Застосування контролерів другого рівня дозволяє організувати на базі одного мережевого контролера, наприклад, доступ через турнікет і до 10 внутрішніх приміщень або через два турнікети і до 8 внутрішніх приміщень, істотно знизивши витрати на впровадження СКУД.

При виборі контролерів СКУД варто звернути увагу на наявність у них додаткових входів/виходів для підключення додаткового обладнання: відеокамер, датчиків, пристроїв сигналізації. Вхід Fire Alarm дозволяє підключити пристрій нежезної сигналізації та налаштувати розблокування турнікетів під час отримання тривожного сигналу від пристроїв ОПС. При підключенні камери до виходу контролера можна задати алгоритм, у якому

камера повинна вести запис під час отримання сигналу про тривожному подію. До входу контролера можна підключити датчики заgonлення: при досягненні певного рівня води сигналом від датчиків турнікети розблокуються для евакуації. Додаткові входи та виходи також дозволяють підключати зовнішні верифікуючі пристрої, наприклад, пірметри, алкотестери, ваги.

На рис. 1.1 зображено типову схему СКУД для підприємства.

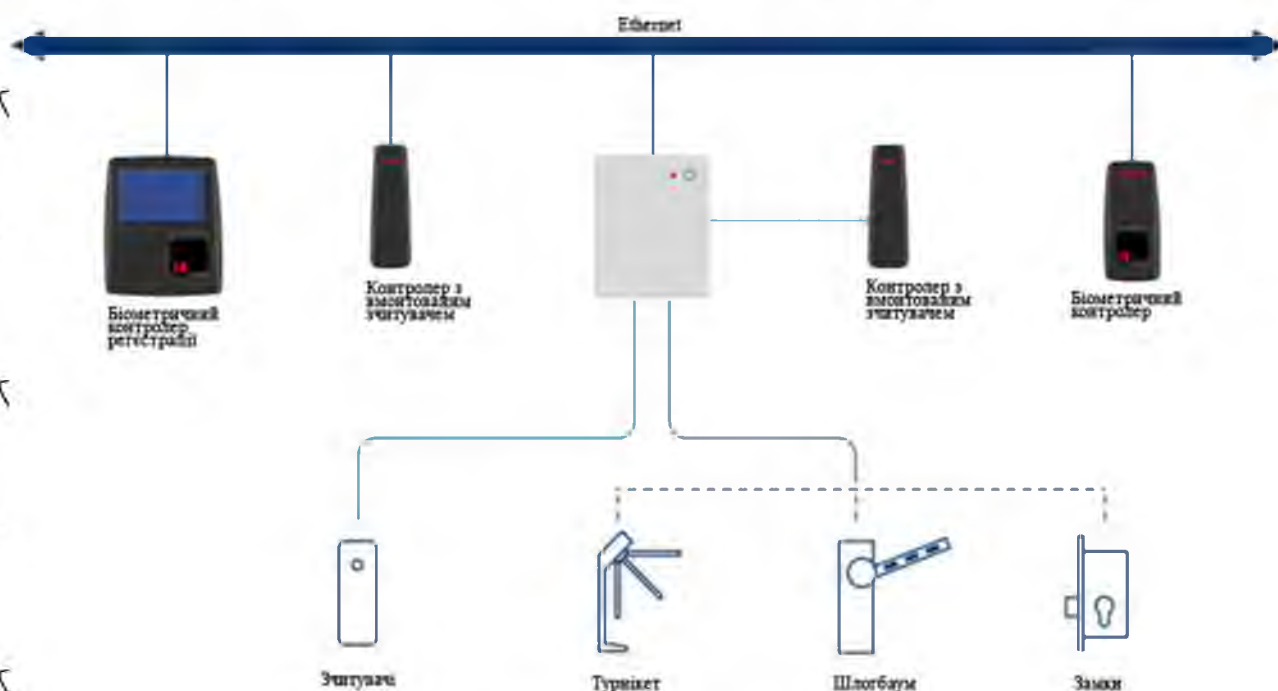


Рис. 1.1. Типова схема системи контролю доступу

1.3.1 Контролер як сервер

Один із головних драйверів ринку СКУД – поширення WEB-технологій. Серед їх переваг – можливість віддаленої роботи на мобільних пристроях із збереженням централізованого адміністрування, функціонування під різними операційними системами.

Контролери нового покоління дозволили вбудувати програмне забезпечення, що дозволило використовувати контролер як сервер. Така архітектура спрощує використання системи та знижує її вартість. Наприклад,

система PERCo-Web, побудована без такого контролера, може обробити дані 500 співробітників і 500 відвідувачів і мати у складі до 10 контролерів. Контролер підключається до мережі за інтерфейсом Ethernet. Для контролю доступу в компанії чисельністю до 100 співробітників буде достатньо безкоштовна версія

ПЗ [3].

Розвиток інтернет-технологій та пропускної спроможності каналів дозволяє говорити про те, що Web-технології незабаром замінять традиційний підхід до розробки не лише програмного забезпечення СКУД, а й взагалі будь-яких систем. У міру появи більш потужних контролерів всі можливості ПЗ систем контролю доступу можна буде реалізувати в них самих, без встановлення сервера системи на комп'ютер.

1.3.2 Web-інтерфейс контролерів

Web-інтерфейс дозволяє підключатися до контролерів безпосередньо з комп'ютера та здійснювати необхідні налаштування. Web-інтерфейс контролерів нового покоління

дозволяє призначати права доступу співробітникам та відвідувачам, використовувати режими «Охорона» та «Комісіювання», додавати ідентифікатори в систему, створювати вбудовані реакції в контролері, проводити діагностику контролера та оновлення вбудованого програмного забезпечення. Для реалізації цих завдань встановлення додаткового програмного забезпечення не потрібне, тому на невеликому підприємстві можна побудувати міні-СКУД без використання програмного забезпечення, що мінімізує витрати на впровадження системи.

1.3.3 Можливості інтеграції

Отримання контролера SDK дозволяє здійснювати інтеграцію з різними системами: наприклад, системами платного доступу або ERP-системами. Відкритий протокол контролера дозволяє організовувати на його базі контроль

доступу у фітнес-центрах, музеях, театрах, парках розваг, парковках та багатьох інших об'єктах.

1.3.4 Вибір способів ідентифікації

При виборі обладнання для системи контролю доступу необхідно визначити, які способи ідентифікації будуть використовуватися на об'єкті: карти доступу форматів EMM/HID або MIFARE із захистом від копіювання, мобільний доступ, штрих-код, відбитки пальців, розпізнавання обличчя. Характеристики контролерів та зчитувачів повинні дозволити реалізувати обраний спосіб ідентифікації.

Для зв'язку контролера та зчитувальних пристроїв застосовуються інтерфейси Wiegand, RS-485 та USB. Wiegand застосовується в СКУД для читання магнітних карт та RFID-ідентифікаторів. Серед переваг інтерфейсу: простота, поширеність, дальність дії до 150 метрів, сумісність обладнання різних виробників. Серед недоліків – вразливість для злому за рахунок відсутності двосторонньої аутентифікації та шифрування даних, відсутність контролю цілісності переданих даних та лінії між контролером та зчитувачем [4].

Для підключення до контролера сканерів відбитків пальців або штрих-кодів, а також для підключення до ПК контрольних зчитувачів, призначених для занесення ідентифікаторів до системи, використовується USB-інтерфейс – універсальний інтерфейс для зв'язку між собою різних цифрових електронних пристроїв.

Для використання одразу кількох способів ідентифікації, наприклад, доступу по картах EMM/HID та MIFARE із захистом від копіювання, а також мобільного доступу можна вибрати мультиформатні зчитувачі. При виборі зчитувача важливо звертати увагу на такі характеристики, як робочий діапазон температур (при використанні на відкритому повітрі), ступінь захисту IP та вандалозахищеність.

Зручним рішенням може стати контролер із уже вбудованим зчитувачем карток доступу, підтримкою мобільної ідентифікації та сканером відбитків

пальців. Такі рішення дозволяють здійснити поступовий перехід від традиційних до захищених способів ідентифікації. Варто звернути увагу на спосіб монтажу контролера. Контролери, що працюють за протоколом Ethernet, досить просто включити в мережу, що значно полегшує монтаж.

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

2 ПРОЕКТУВАННЯ СИСТЕМИ

2.1 Аналіз та моделювання системи

Розглянемо поведінку автоматизованої системи контролю доступу до приміщення як набір функцій та акторів, що з ними взаємодіють. Модель поведінки системи подано в нотатції UML, діаграмою прецедентів, що описує доступні дії і випадки (рис. 2.1).

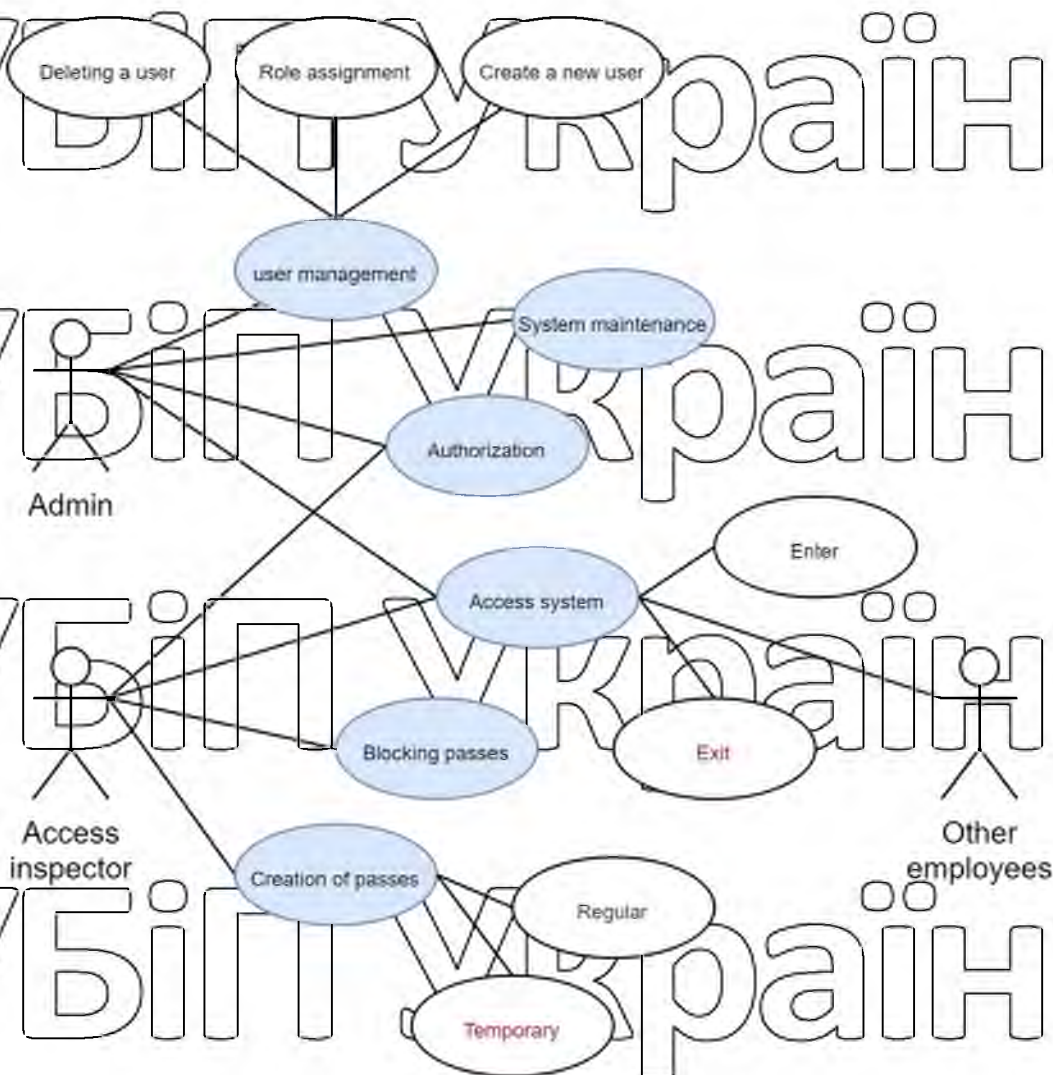


Рис. 2.1. Діаграма прецедентів автоматизованої системи контролю відвідувань

Для предметної області виділено наступних акторів, що зображено в табл.

§.1.

Таблиця 2.1

Визначені актори

Актор	Короткий опис
Співробітник	Звичайний співробітник, який користується системою і має доступ до входу і виходу з приміщення ;
Адміністратор	Співробітник, який має можливість переглядати статистику користувачів та корегувати в разі необхідності;
Інспектор бюро перепусток	Користувач, який може переглядати статистику відвідування закладу окремих користувачів, коригувати доступ до системи, реєструвати нових користувачів.

Розглянемо тепер, які можливості має надавати наша система:

- актор Співробітник використовує систему для доступу у приміщення;
- актор Адміністратор використовує систему для перегляду звітності роботи системи, реєстрації та адміністрування системи (редагування користувачів та надання прав іншим користувачам системи);
- актор Надавач доступу використовує систему для реєстрації та адміністрування системи (надання прав іншим користувачам системи);

Узагальнюючи роботу системи, можна зробити висновок, що інспектор бюро перепусток формує картки співробітників, прив'язує до них ідентифікатор мітки і передає дані в систему контролю доступом. Адміністратор системи стежить за працездатністю системи, створює та керує обліковими записами користувачів. Статистика формується автоматично виходячи з інформації переданої мікроконтролером в програмну частину програмно апаратного комплексу (ПАК). Система взаємодіє з іншими співробітниками, зчитуючи їх пропуски (мітки) та забороняючи/дозволяючи прохід через певну зону контролю, а також фіксує їх прохід у журналі подій.

На підставі вищевикладеного можна виділити наступні факти, що зображені в таблиці 2.2:

НУБІП України

Опис прецедентів

Прецедент	Короткий опис
Керування користувачами	Запускається адміністратором. Дозволяє редагувати доступ до приміщень, видаляти користувачів, та створювати нових.
Обслуговування системи	Запускається адміністратором. Дозволяє переглянути статистику та перевірити стан пристроїв системи.
Авторизація	Запускається адміністратором та інспектором бюро перепусток. Дозволяє увійти у персональний кабінет.
Доступ до системи	Запускається співробітником, адміністратором та інспектором бюро перепусток. Дозволяє користуватися системою зареєстрованим користувачам.
Блокування доступу	Запускається інспектором бюро перепусток. Дозволяє блокувати доступ до різних частин підприємства.
Надання доступу	Запускається інспектором бюро перепусток. Дозволяє надавати постійний або тимчасовий доступ до різних частин підприємства.

Розглянути детальніше алгоритм роботи апаратної частини СКУД

дозволяє діаграма станів, див. рис. 2.2.

НУБІП України

НУБІП України

НУБІП України



Рис. 2.2. Діаграма станів, де продемонстровано алгоритм роботи апаратної частини

При прикладанні пропуску співробітником та зчитуванні даних відбувається автентифікація. В результаті автентифікації система реагує на подію: відкриває механізм, що замикає, або ігнорує дії співробітника, переходячи в початковий стан. При виконанні вищезгаданих дій система записує всі події в журнал.

В якості платформи для розробки контролера ЕКУД використовується платформа ESP32. За способом керування контролер відноситься до мережевих.

що дозволяє створити СКУД будь-якого ступеня складності. Контролер дозволяє використовувати передачу даних по мережі WiFi на сервер, що використовується для зберігання інформації.

Для зчитування карт до контролера підключений зчитувач RC522, який дозволяє зчитувати дані проксіміті карт.

Для запобігання помилок зчитування карт при відсутності підключення до інтернету, потрібно використовувати локальну пам'ять, для цього може підійти мікросхема AT24C256N.

В якості виконавчих пристроїв для здійснення пропускнуго режиму в системі контролю обмеженого доступу використовуються електромагнітні замки.

2.2 Підбір компонентів системи

Підбір компонентів для друкованої плати є одним з важливих аспектів під час проектування системи. Тому під час проектування схеми друкованої плати для її правильної роботи в остаточному проекті необхідно підібрати правильні компоненти. Схема повинна працювати відповідно до своїх специфікацій і для цього буде перевірена за допомогою засобів моделювання та аналізу схем, щоб перевірити це під час розробки плати.

Але продуктивність роботи системи під час аналізу – це не єдиний аспект, який повинен бути для успішного проектування друкованої плати.

Фізичні компоненти, які будуть міститися на платі повинні бути сумісними з характеристиками інших компонентів, які будуть використовуватися для створення плати. Однак перед вибором компонента є ще кілька інших міркувань, які потрібно переглянути.

При виборі компонентів для використання на друкованій платі також необхідно враховувати наступні три фактори:

Ціна: вартість компонента може відрізнятись у різних виробників і дистриб'юторів і повинна бути підтверджена перед тим, як взяти на себе зобов'язання щодо великих тиражів друкованої плати.

Доступність: деталі можуть бути в межах цінового діапазону для бюджету проекту, але затримки доставки все ще можуть підірвати збірку друкованої плати. Затримки можуть виникати з ряду причин, включаючи національні свята, політичні проблеми або навіть стихійні лиха, як-от пандемія.

Життєвий цикл: Компоненти час від часу вилучатимуться з продажу їх постачальника, що може спричинити проблеми для довгострокових збірок друкованої плати. Багато разів деталі вибирають для створення прототипу плати, але не розглядають їх довговічність. Важливо з'ясувати статус компонента та уникати тих, які мають позначку NRND (не рекомендоване для нових конструкцій), OBS (застарілий) або EOL (кінець терміну служби), перш ніж почати його використання [5].

Після вибору компонентів для дизайну наступне завдання – зберегти їх, щоб можна було повторно використовувати їх у майбутніх конструкціях. Ось тут вам потрібна добре розвинена бібліотека друкованих плат, щоб усі ці частини були готові до наступного дизайну.

2.2.1. Обґрунтування вибору мікроконтролеру ESP32

Мікроконтролер ESP32 – це одна з найдоступніших і найпотужніших платформ для створення розумних ардуїно-проектів з підтримкою WiFi.

Прийшовши на зміну ESP8266, цей чіп дав нові можливості для розробників, хоча, як і раніше, залишилися старі проблеми з підтримкою та документацією [6].

Пристрій являє собою систему на кристалі, побудовану за технологією TSMC 40 нм, з Wi-Fi та Bluetooth контролерами. Воно оснащено двоядерним 32-бітним процесором, який працює на частотах 80, 160 або 240 МГц. Також у систему інтегровані антенні комутатори, радіочастотні компоненти, фільтри,

підсилювачі, модулі керування живленням. Підключається ESP32 до комп'ютера через звичайний провід USB.

Технічні характеристики ESP32:

- двоядерний 32-бітовий процесор Tensilica Xtensa LX6;
- тактова частота – 160 чи 240 МГц;
- 520 Кб SRAM;
- максимальний струм споживання 260 мА, в режимі сну – 10 мА;
- стандарти бездротового зв'язку – Wi-Fi: 802.11 b/g/N, Bluetooth: v4.2 BR/EDR and BLE;

- наявність датчиків температури, Холла, тач-сенсорів;
- інфрачервоне дистанційне керування;
- можна підключати двигуни та світлодіоди через ШІМ роз'єм;

- стандарт IEEE 802.11 з підтримкою WPA, WPA/WPA2 та WAPI;
- можливість безпечного завантаження;
- шифрування флеш-диска.

Також у модулі традиційно є вбудоване управління енергоживленням. Для цього використовуються лінійний регулятор, індивідуальне живлення RTC (ядро низького енергоспоживання), пробудження по таймеру або сенсорному датчику.

Мікроконтролер оснащений 48 контактами та 1 великим тепловим контактом, які мають різні функції [7].

На рис. 2.3 зображено схему вихідних портів мікроконтролера.

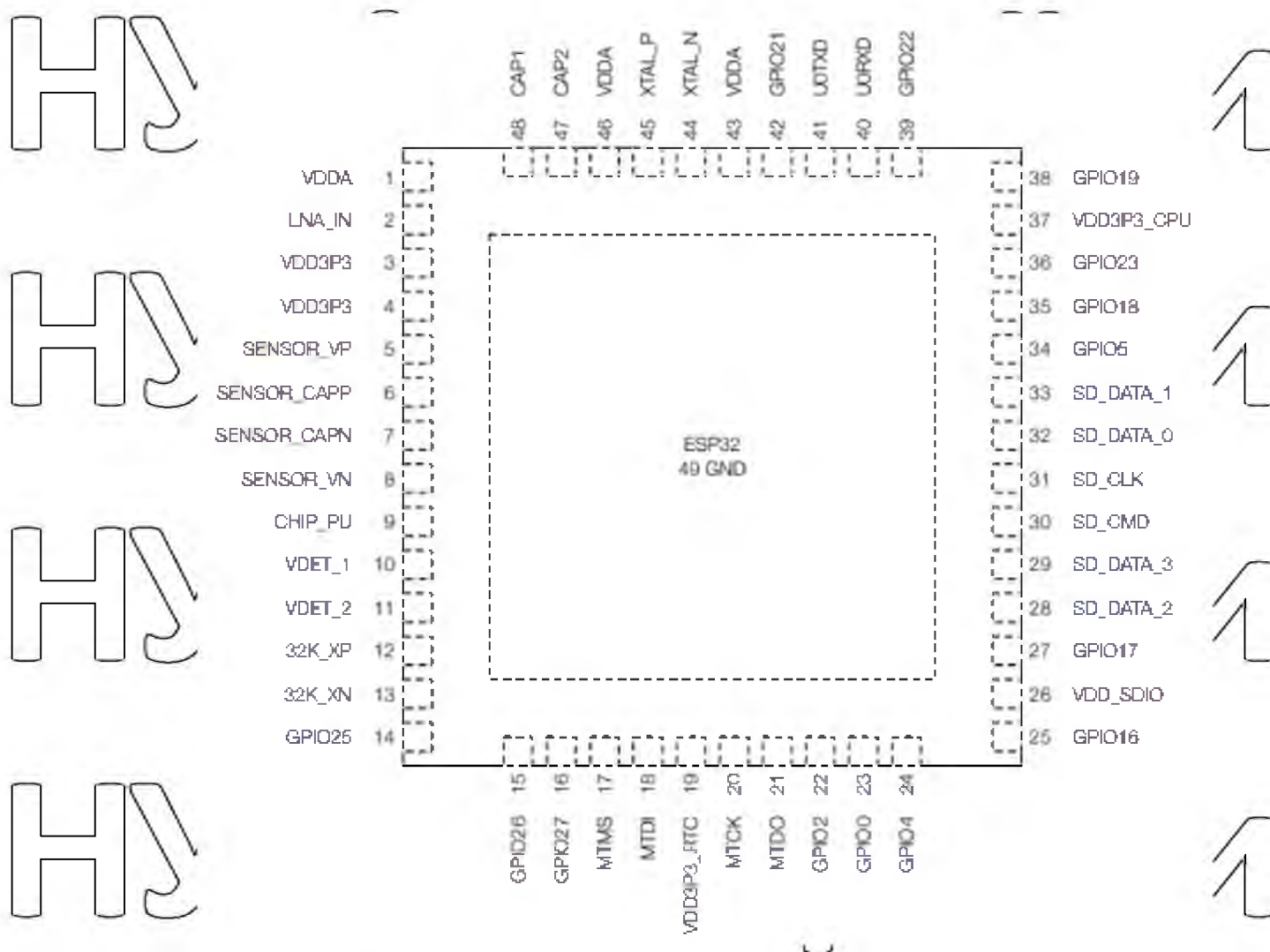


Рис. 2.3. Розпінування плати ESP32

2.2.2 Обґрунтування вибору мікросхеми AT24C256N

Прилади, що входять до сімейства AT24C, є послідовними електрично стираються та програмується ПЗУ ємністю від 1 до 1024 Кбіт з 8-ми розрядною організацією. У режимі читання забезпечується читання одного байта, послідовності байтів до повного обсягу пам'яті, а режимах програмування забезпечується запис як окремого байта, і цілої сторінки чи його частини. Об'єм сторінки становить від 4 до 256 байт (залежно від типу приладу). Вхід та вихід даних організується за допомогою простого двопровідного інтерфейсу. Усі прилади сімейства, крім AT24C01, оснащені засобами апаратного захисту даних. Крім того, є можливість каскадування приладів сімейства збільшення доступного об'єму пам'яті (за винятком приладів AT24C01, AT24C21 і AT24C16).

Всі члени сімейства AT24C розроблялися для тих областей застосування промислового та побутового діапазону температур, де важливе мале споживання та низька напруга живлення, та розміщені у 8-висновних, 14-висновних корпусах PDIP, JEDEC SOIC та EIAI SOIC, що займають незначну поверхню друкованої плати [8].

Характеристики мікроконтролеру:

- 2-шинний послідовний інтерфейс;
- тригер Шмідта, входи з фільтрами для придушення шуму;
- двонаправлений протокол передачі;

- робота з низькою та стандартною напругою;

- внутрішня організація 32.768 x 8;

- сумісність 100 кГц (1.8В), 400 кГц (2.7В) та 1МГц (5В);

- 64-байтовий режим запису сторінки (5 мс типове);

- доступний захист від запису;

- автосинхронізований цикл запису (10 мс максимум);

- висока надійність;

- строк служби: 1 мільйон циклів запису;

- термін зберігання: 40 років;

- захист ESD: > 4000В;

- 8-контактні корпуси JEDEC PDIP та JEDEC SOIC, 14-контактні корпуси TSSOP та корпус з 8-стовпцевими контактами.

AT24C128, AT24C256 забезпечує 128К/256 Кбіт програмованого постійно запам'ятовуючого пристрою (EEPROM), організованого як 16.768/32.768 слів по 8 біт / кожне. Пристрій оптимізовано для використання в програмах, які потребують зберігання даних та послідовної передачі інформації керування та конфігурації. AT24C128, AT24C256 доступний в економних місцях 8-контактних корпусах JEDEC PDIP, 8-контактних корпусах JEDEC SOIC, 8-контактних корпусах EIAI, 14-контактних корпусах TSSOP і в корпусах LAP з 8-стовпцевими контактами.

2.2.3 Обґрунтування вибору резисторів

Недротяні постійні резистори (С2-33И), що застосовуються в пристрої гальванічної розв'язки, призначені для роботи в ланцюгах постійного, змінного струмів і в імпульсних режимах, містять резистивний елемент у вигляді дуже тонкої (десятки частки мікрометра) металевої плівки, обложеної на підставі з шаруватого пластику, ситалу або іншого ізоляційного матеріалу. Недротяні постійні резистори менш стабільні порівняно з дротяними, але поряд з цим мають менші габарити, їх опір менш залежить від частоти та напруги, вони значно дешевші. Тому в апаратурі недротяні резистори застосовуються значно частіше, ніж дротяні [9].

Характеристики, постійних резисторів:

- мінімальний термін збереження – 25 років;
- діапазон робочих температур - від -60°C до $+70^{\circ}\text{C}$;
- відхилення опору: 5%;

Отже, постійні дротяні резистори типу С2-33И підходять для використання в проєктованому пристрої.

2.2.4 Обґрунтування вибору конденсаторів

Конденсатори К50-35, К50-35Б, К50-35І - окисно-електролітичні конденсатори (електроліти) постійної ємності в ущільненому алюмінієвому циліндричному корпусі, накопичують заряд від 2,2 мкФ до 4700 мкФ при напрузі від 6,3В до 350В. Допустиме відхилення ємності становить $\pm 20\%$. Застосування – ланцюг постійного чи пульсуючого струму, імпульсні режими [10].

Виготовляють конденсатори у всекліматичному виконанні (В) та для помірного або холодного клімату (УХЛ). Конденсатори серії К50-35Б оснащені запобіжним клапаном, що забезпечує вибухобезпечність. Конденсатори серії К50-35І виконані у ізольованому корпусі.

Дротеві виходи конденсаторів К50-35, К50-35Б, К50-35І радіальні, односпрямовані, полярні, що передбачає дотримання полярності при включенні конденсатора в схему. На бічній поверхні корпусу електролітичного

накопичувача наведено технічні характеристики із зазначенням полярності (знак "+"). Довжина негативного виведення не менше ніж 15 мм, довжина позитивного виводу перевищує довжину негативного не менше ніж на 5 мм.

Кріплення конденсаторів - за виходи за допомогою групового паяння або корпус. При згинанні висновків мінімальна відстань від корпусу становить 1,5 мм.

Підвищена робоча температура середовища становить трохи більше $+85^{\circ}\text{C}$, робоча знижена температура - не нижче -40°C , гранична знижена температура сягає -60°C . Тангенс кута втрат $\text{tg}\delta$ не вище 0,24, максимальний струм витoku - 3мкА. Напрацювання при цьому становить не менше 2000 год.

До зовнішніх факторів, що впливають електроліти K50-35, K50-35Б, K50-35I, відносять вібрації з частотою 10-500Гц при прискоренні 10g і механічні удари одиночного (прискорення 150g) прискорення 40g) дії.

Основною перевагою електролітичних конденсаторів K50-35, K50-35Б, K50-35I є високі показники ємності та номінальної напруги при невеликих габаритах.

Застосовуються електролітичні конденсатори K50-35, K50-35Б, K50-35I для неавтоматизованого та автоматизованого складання апаратури, при згладжуванні вимрямлячами пульсуючого струму в ланцюгах зі змінним струмом, а також широко використовуються в звуковій техніці.

Виходячи з параметрів, дані конденсатори підходять для застосування їх у пристрої.

2.2.5. Обґрунтування вибору транзисторів

Характеристики біполярного кремнієвого транзистора BC337 дозволяють віднести його до високочастотних напівпровідникових пристроїв середньої потужності. Матеріал кремній, структура n-p-n. Струм колектора в перших партіях від Philips не перевищував 500 мА. Разом з тим, сучасні екземпляри інших виробників здатні витримувати до 800 мА, з максимальною напругою до 45 В. Це дозволяє безболісно застосовувати їх в управлінні різними

навантаженнями (діодами, лампочками, реле), що вимагають більше 24 В. Високий коефіцієнт НFE робить можливим використовувати їх у різних схемах передусім та у вихідних каскадах УНЧ малої потужності [11].

У максимальних рейтингах відображено граничні експлуатаційні властивості BC337. Перевищення цих значень може призвести до виведення пристрою з ладу. Їх повний список:

- напруга між колектором та емітером (при розімкненому ланцюгу бази) V_{CEO} не більше 45 В;

- напруга між колектором та емітером (база з'єднана з емітером) V_{CES} не більше 50 В;

- напруга між емітером та базою V_{EB0} не повинна перевищувати 5 В;

- струм колектора I_C не повинен бути більше 800 мА;

- максимальна потужність P_C , яку транзистор може розвіяти 625 мВт;

- гранично допустима температура 150°C;

- діапазон робочих температур A_{STG} від -55°C до 150°C.

Біполярний n-p-n транзистор S8050 – напівпровідниковий прилад, який відноситься до високочастотних біполярних транзисторів загального

призначення. Набув широкого поширення в електронному світі в основному в

схемах комутації та підсилювачах. Має наступні основні параметри: колекторний струм до 700 мА, максимальна напруга до 26 В, низький рівень власних шумів і коефіцієнт посилення по струму до 300 [12].

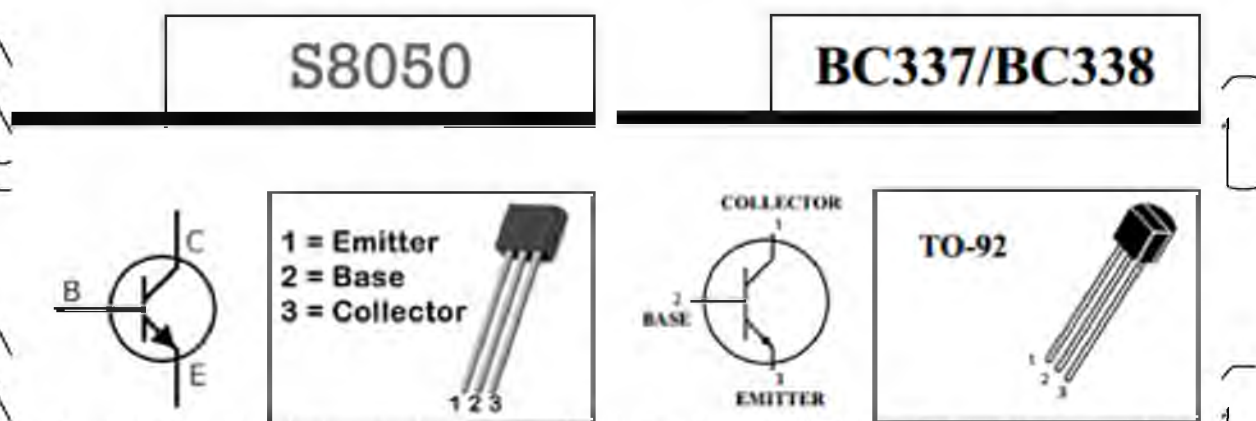


Рис. 2.4. Транзистори S8050 та BC337

Отже, транзистори типу BC337 та S8050DAF підходять для використання в проєктованому пристрої.

2.2.6 Обґрунтування вибору стабілізатору напруги

Для системи потрібно 3 варіанти підключення: 3.3V, 5V та 12V. Для цього потрібно використовувати стабілізатори з 12V до 5V і з 5V до 3.3V.

Пристрої, що входять до схеми блоку живлення, і підтримують стабільну вихідну напругу, називаються стабілізаторами напруги. Ці пристрої розраховані на фіксовані значення виходу напруги: 1.3, 3.3, 5, 9 або 12 вольт. Але є пристрої з наявністю регулювання. У них можна встановити бажану напругу у певних доступних межах.

Більшість стабілізаторів призначені на певний максимальний струм, який вони витримують. Якщо перевищити цю величину, то стабілізатор вийде з ладу. Інноваційні стабілізатори оснащені блокуванням струму, що забезпечує вимкнення пристрою при досягненні найбільшого струму в навантаженні та захищені від перегріву. Разом зі стабілізаторами, які підтримують позитивне значення напруги, є пристрої, що діють з негативною напругою. Вони застосовуються у двополярних блоках живлення.

Стабілізатор 7805 виготовлений у корпусі, подібному до транзистора. Він розрахований на напругу 5 вольт і 1 струм ампер. У корпусі є отвір фіксації стабілізатора до радіатора. Модель 7805 є пристроєм позитивної напруги.

Дзеркальне відображення цього стабілізатора – це його аналог 7905, призначений для негативної напруги. На корпусі буде позитивна напруга, на вхід надійде негативне значення. З виходу знімається -5 В. Щоб стабілізатори працювали у нормальному режимі, потрібно подавати на вхід 10 вольт.

Стабілізатор 7805 має розпинання, яке показано на рис. 2.5. Загальний висновок з'єднаний із корпусом. Під час встановлення пристрою це відіграє важливу роль. Дві останні цифри позначають напругу, що видається мікросхемою.

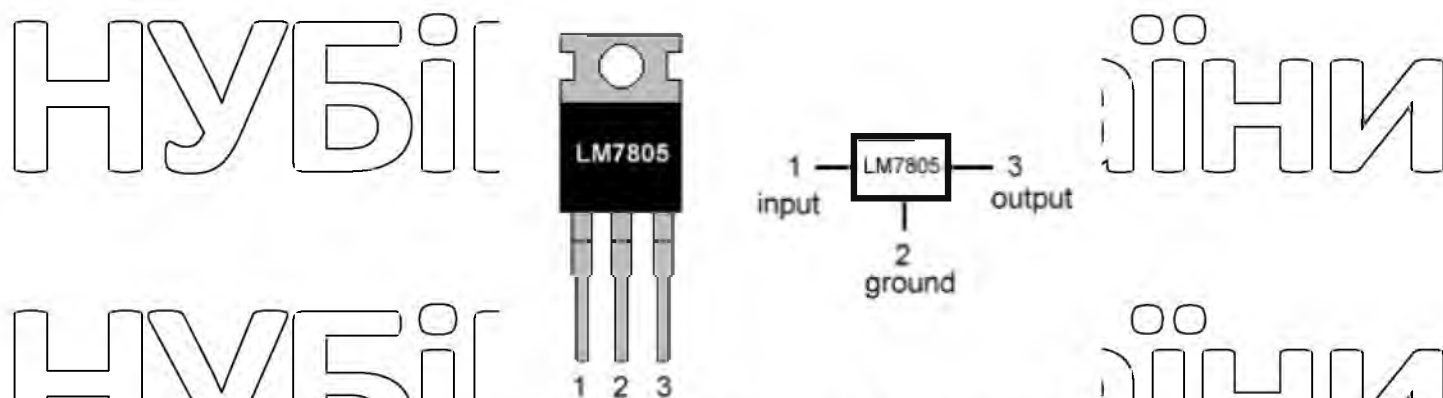


Рис. 2.5. Стабілізатор LM7805

AMS1117-3.3 є понижуючим стабілізатором напруги постійного струму популярної серії AMS1117. Вихідна напруга мікросхеми AMS1117-3.3 становить 3,3В. Стабілізатор напруги часто використовується в аналогових пристроях, портативних приладах, зарядках акумуляторів тощо [13].

Мікросхема AMS1117-3.3 виконана у корпусі «SOT-223». Максимальне значення вихідного струму стабілізатора становить 800 мА. Цей стабілізатор має широкий діапазон робочої температури, який знаходиться в межах від -40 до $+125$ °C.

Стабілізатор AMS1117-3.3 має 3 виходи. Призначення виходів:

- GND: "земля";
- V_{out} : вихідна напруга 3,3 В;
- V_{in} : вхідна напруга 4,8 – 12 В.

Незважаючи на невеликі габарити стабілізатора, він має широкий діапазон вхідної напруги, який знаходиться в межах від 4,8 до 12 В (див. рис. 2.6).

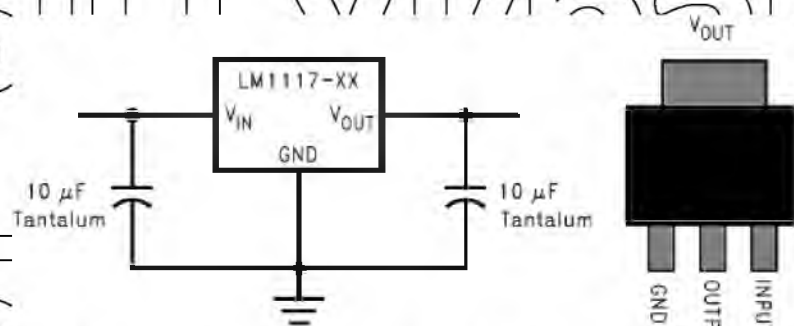


Рис. 2.6. Стабілізатор LM1117

2.3 Проектування друкованих плат

Для проектування друкованої плати було вирішено використовувати онлайн сервіс для проектування та емуляції плат.

EasyEDA – це кросплатформовий комплекс, призначений для розробки електричних принципових схем, автоматизованого розведення друкованих плат та надає можливість здійснити замовлення на виготовлення ваших плат.

Можливо, що вас зацікавить і симулятор електронних схем, який також входить до списку безкоштовних послуг EasyEDA. До складу EasyEDA входить:

- редактор електричних схем, компонентів і готових модулів з великою бібліотекою, що автоматично оновлюється, що містить сотні тисяч комплектуючих;

- трасувальник, редактор топології друкованих плат;

- симулятор схем, двигун симулятора від ngspice;

- переглядач файлів формату Gerber;

- виготовлення друкованих плат.

Проект у EasyEDA – це сукупність схем та даних про розмітку плати.

Новий проект можна створити «з нуля» або скопіювати в когось уже існуючий.

Також хтось може надати вам доступ до свого проекту для спільної роботи. Для створення першого проекту «з нуля» потрібно перейти в редактор і там створити

новий проект кнопкою зліва вгору. Надалі ви можете створювати нові проекти

безпосередньо з головної сторінки EasyEDA.

Для початку потрібно створити принципові електричні схеми, які визначають повний склад приладів, апаратів та пристроїв (а також зв'язків між ними), дія яких забезпечує вирішення завдань управління, регулювання, захисту, вимірювання та сигналізації. Принципові схеми є основою розробки інших

документів проекту: монтажних таблиць щитів і пультів, схем зовнішніх з'єднань та інших.

Ці схеми дають детальне уявлення про роботу системи і служать також вивчення принципу дії системи, вони необхідні під час виробництва налагоджувальних робіт і експлуатації.

При розробці систем автоматизації технологічних процесів принципові електричні схеми зазвичай виконують стосовно окремих самостійних елементів, установок або ділянок автоматизованої системи, наприклад виконують схему управління засувкою, схему автоматичного та дистанційного управління насосом, схему сигналізації рівня в резервуарі і т.п. Використовуючи ці схеми, становлять у разі потреби принципові електричні схеми, що охоплюють цілий комплекс окремих елементів, установок або агрегатів, які дають повне уявлення у зв'язках між усіма елементами управління, блокування, захисту та сигналізаційних установок або агрегатів. Прикладом таких схем може бути принципова електрична схема управління насосною установкою, що складається з насоса, вакуум-насоса та кількох електрифікованих засувок.

При всьому різноманітті принципових електричних схем у різних системах автоматизації будь-яка схема, незалежно від ступеня її складності, є певним чином складене поєднання окремих, досить елементарних електричних ланцюгів і типових функціональних вузлів, в заданій послідовності виконують ряд стандартних операцій: передачу командних сигналів від органів управління або вимірювання до виконавчих органів, посилення або розмноження командних сигналів, їх порівняння, перетворення короткочасних сигналів на тривалі та, навпаки, блокування сигналів тощо. До елементарних ланцюгів можна віднести типові схеми включення вимірювальних приладів різного призначення.

Розробка принципових електричних схем завжди містить певні елементи творчості і вимагає вмілого застосування елементарних електричних ланцюгів і типових функціональних вузлів, оптимального компоновання їх у єдину схему з урахуванням задоволення вимог, що пред'являються до схем, а також можливого спрощення та мінімізації схем. У практиці проектування важливих електричних схем з урахуванням досвіду проектування монтажу, налагодження та експлуатації різноманітних систем автоматизації склалися деякі загальні

принципи побудови електричних схем. Питання методах розробки важливих електричних схем у процесі проектування систем автоматизації технологічних процесів слід розглядати у загальному комплексі питань, що з контролем, управлінням і регулюванням даного об'єкта. У всіх випадках крім повного задоволення вимог, що висуваються до системи управління, кожна схема повинна забезпечувати високу надійність, простоту та економічність, чіткість дій при аварійних режимах, зручність оперативної роботи, експлуатації, чіткість оформлення.

Основним обчислювальним центром виступає мікроконтролер ESP32, до якого підключені периферійні пристрої, на рис. 2.7. зображено частину принципової схеми з мікроконтролером.

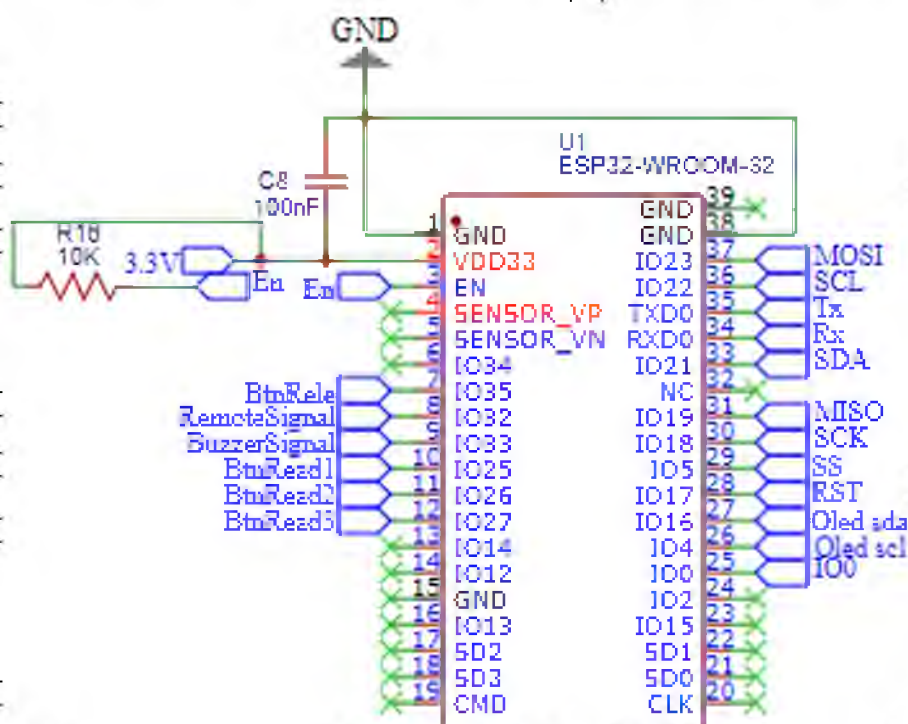


Рис. 2.7. Частина принципової схеми з контролером

У системі присутня панель керування, яка складається з дванадцяти кнопок. Першим варіантом було підключення усіх кнопок напряму до контролеру, але ця схема буде витрачати ресурси мікроконтролеру та займати багато портів. Тому для вирішення цієї проблеми було прийнято рішення

використовувати дільник напруги, а саме для отримання з вихідної напруги лише його частину, тому схема будується з урахуванням багатьох резисторів

Існує безліч компонентів, які змінюють опір в залежності від зовнішніх умов. Так термістори змінюють опір від нуля до певного значення в залежності від температури, фоторезистори змінюють опір в залежності від інтенсивності світла, що потрапляє на них, і т.д.

На рис. 2.8 зображено схему підключення панелі керування.

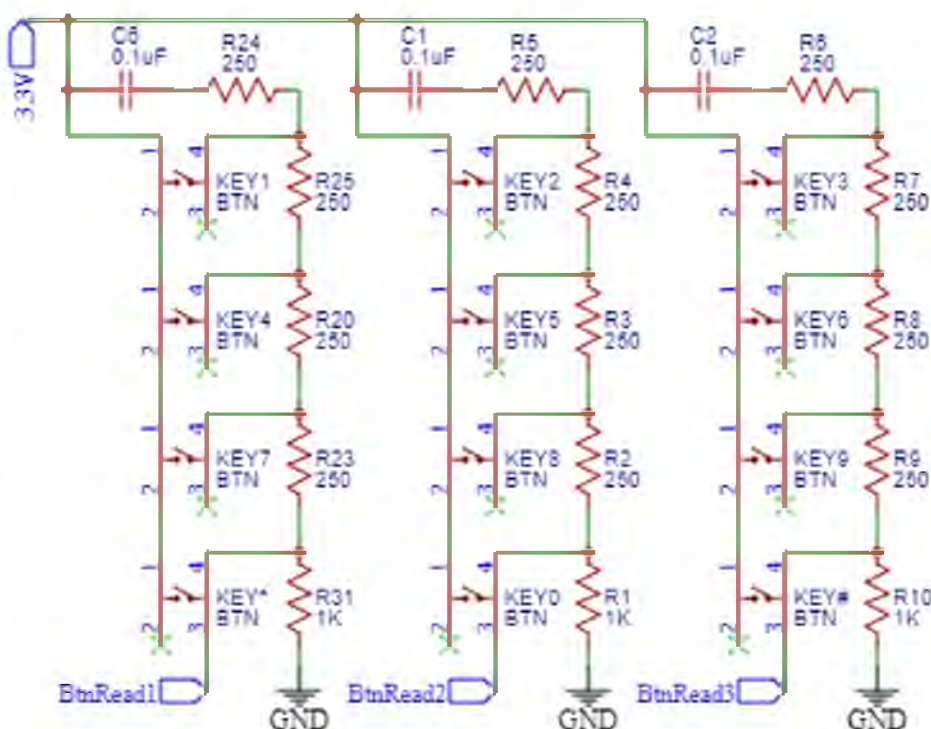


Рис. 2.8. Частина принципової схеми з панеллю керування

Дільник напруги можна як дві послідовних ділянки ланцюга, звані плечима, сума напруг у яких дорівнює вхідному напрузі. Плечо між нульовим потенціалом і середньою точкою називають нижнім (з нього зазвичай знімається вихідна напруга дільника), а інше верхнім. Розрізняють лінійні та нелінійні

дільники напруги. У лінійних вихідна напруга змінюється за лінійним законом залежно від вхідного. Такі дільники використовуються для задання потенціалів та робочих напруг у різних точках електронних схем. У нелінійних дільниках

вихідна напруга залежить від коефіцієнта а нелінійно. Нелінійні ділянки напруги застосовують у функціональних потенціометрах. Опір може бути як активним, так і реактивним, а також зовсім нелінійним, як, наприклад, в параметричному стабілізаторі напруги.

Також було додано конденсатор для усунення «брякоту контактів»

Брязкотіння контактів - явище, що відбувається в електромеханічних комутаційних пристроях і апаратах (кнопках, реле, герконах, перемикачах, контакторах, магнітних пусках та ін), що триває деякий час після замикання електричних контактів. Після замикання відбуваються багаторазові

неконтрольовані замикання та розмикання контактів за рахунок пружності матеріалів та деталей контактної системи - деякий час контакти відскакують один від одного при зіткненнях, розмикаючи та замикаючи електричний ланцюг.

На рис. 2.9. зображено схему підключення стабілізаторів напруги, які є ділянками напруги, на вхід яких подається вхідна (нестабільна) напруга, а вихідна (стабілізована) напруга знімається з нижнього плеча діляника. Стабілізація здійснюється шляхом зміни опору одного з плечі діляника: опір постійно підтримується таким, щоб напруга на виході стабілізатора знаходилася у встановлених межах.

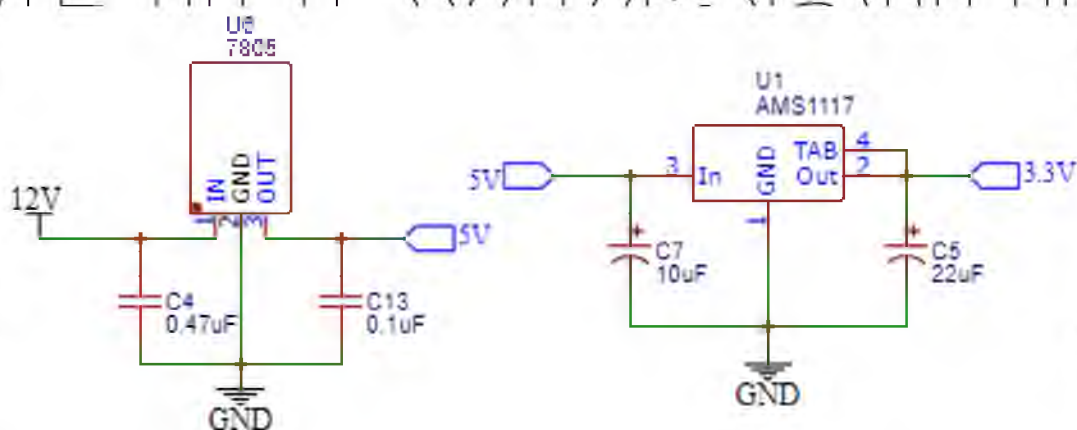


Рис. 2.9. Частина принципової схеми з стабілізаторами напруги

Після створення принципової схеми, є можливість спроектувати друковану плату. Друкована плата (PCB) – пластинка з діелектрика, на поверхні та/або в

обсязі якої сформовані електропровідні ланцюги електронної схеми. Друкована плата призначена для електричного та механічного з'єднання різних електронних компонентів. Електронні компоненти на друкованій платі з'єднуються своїми висновками з елементами малюнку, що проводить зазвичай пайкою.

Друкована плата розроблялась з урахуванням усіх необхідних компонентів, які вказані в таблиці 2.3

Таблиця 2.3

Опис компонентів системи

ID	Назва	Опис	К-сть	Назва від виробника	Виробник
1	0.1 μ F	C1,C2,C6,C13	4		
2	0.47 μ F	C4	1		
3	22 μ F	C5	1	22 μ F 16V 4*5	ValuePro
4	10 μ F	C7	1	10 μ F 16V 5*11	ValuePro
5	100nF	C8	1		
6	4Pin	H1,H3	2	210S-1*4P L=11.6MM Gold-plated black	Ckmtw
7	Boot	H2	1	210S-1*4P L=11.6MM Gold-plated black	Ckmtw
8	MHz	H4	1	210S-1*4P L=11.6MM Gold-plated black	Ckmtw
9	BTN	KEY0,KEY1,KEY2,KEY3,KEY4,KEY5,KEY6,KEY7,KEY8,KEY9,KEY#,KEY*	12	button 6*6*8	ReliaPro
10	rc522	P1	1	Female header HDR1X8-2.54	BOOMEL E
11	BC337	Q3,Q4	2	BC337	CJ
12	S8050D AF	Q8	1	S8050DAF	FeiHong
13	1K	R1,R10,R14,R15,R22,R31	6		
14	250	R2,R3,R4,R5,R6,R7,R8,R9,R20,R23,R24,R25	12		
15	10K	R16,R17,R18,R19	4		

Продовження таблиці 2.3

ID	Назва	Опис	К-сть	Назва від виробника	Виробник
16	BtnReset	S1	1	3x6x2.5mm	ReliaPro
17	BtnBot	S2	1	3x6x2.5mm	ReliaPro
18	Buzzer	SG2	1		
19	ESP32-WROOM-32	U1	1	ESP32-WROOM-32	Espressif Systems
20	0.91" OLED	U3	1	598	Adafruit
21	AMST 117.5 VOLT	U4	1		
22	AT24C256N	U5	1	AT24C256N	HGSEMI
23	7805	U6	1	7805	Foshan Blue Rocket Elec

На відміну від навісного монтажу, на друкованій платі електропровідний малюнок виконаний з фольги, повністю розташованої на твердій ізолюючій основі. Друкована плата містить монтажні отвори та контактні майданчики для монтажу похідних або планарних компонентів. Крім того, друковані плати мають перехідні отвори для електричного з'єднання ділянок фольги, розташованих на різних шарах плати. З зовнішніх сторін на плату зазвичай нанесені захисне покриття («паяльна маска») та маркування (допоміжний малюнок та текст згідно з конструкторською документацією).

Залежно кількості шарів з електропровідним малюнком друківані плати поділяють на:

- односторонні (ОПП): є лише один шар фольги, наклеєної на один бік листа діелектрика;

- двосторонні (ДПП): два шари фольги;

- багатошарові (МПП): фольга не лише на двох сторонах плати, а й у внутрішніх шарах діелектрика. Багатошарові друківані плати виходять склеюванням кількох односторонніх чи двосторонніх плат.

Для проекту було вирішено робити двосторонню плату з мідним провідником.

На рис. 2.10 та 2.11 зображені готові друківані плати для подальшого виробництва на заводі виробника.

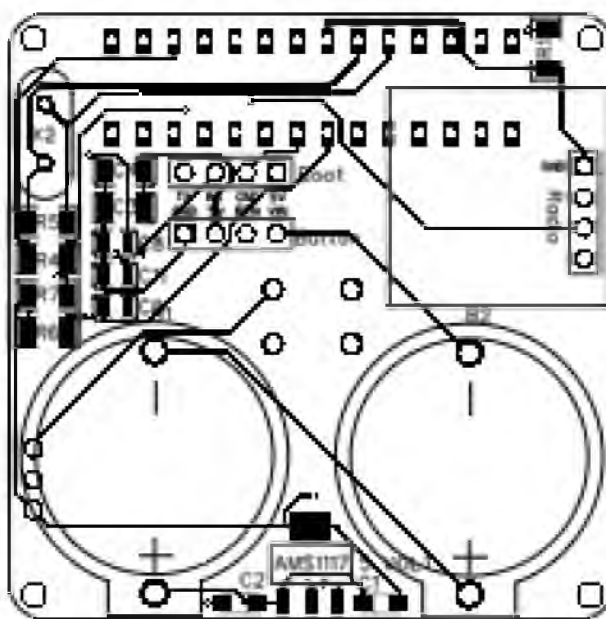


Рис. 2.10. Друківана плата допоміжного пристрою

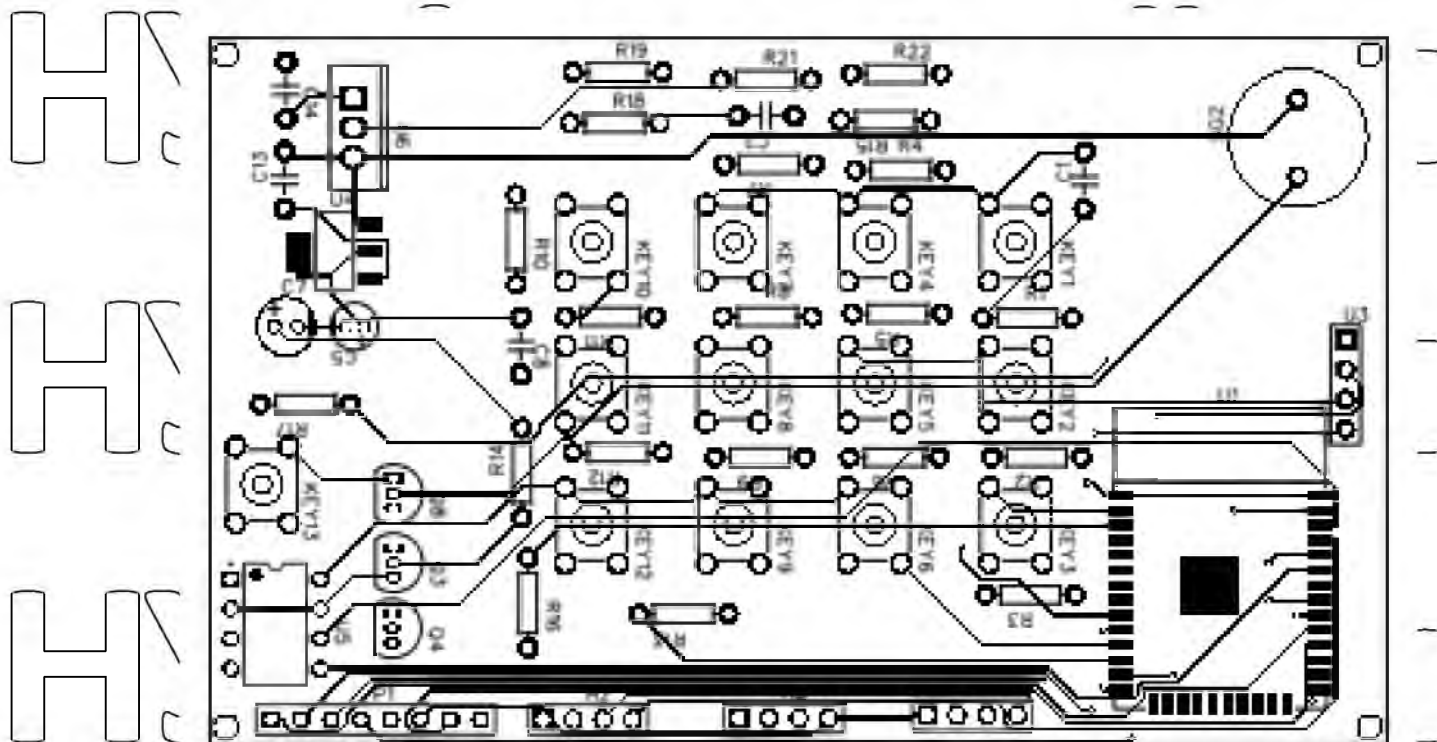


Рис. 2.11. Друкована плата головного пристрою

2.4 Монтаж компонентів на друковану плату

Щоб зробити замовлення на виробництво друкованих плат, необхідно надіслати електронною поштою готовий PCB проєкт плати в форматі Gerber-файлів (RS-274X) і файлів свердління в форматі Excellon [14].

Вимоги до файлів друкованих плат:

конструкторська документація на виробництво друкованих плат приймається в електронному вигляді.

- завжди пишуть на платі дату, її назву або номер в тому ж шарі, що і провідники;

- габарити плати малювати лінією товщиною 0.3 мм по всьому периметру друкованої плати;

- бажано вказати в тексті і кількість отворів на платі (наприклад, 3/2 мм — 4 шт., 1.2 мм — 4 шт., інші — 0.9 мм).

Після замовлення плат на ресурсі jlcpcb.com, було отримано готові друковані плати, що зображені на рис. 2.12 та 2.13.

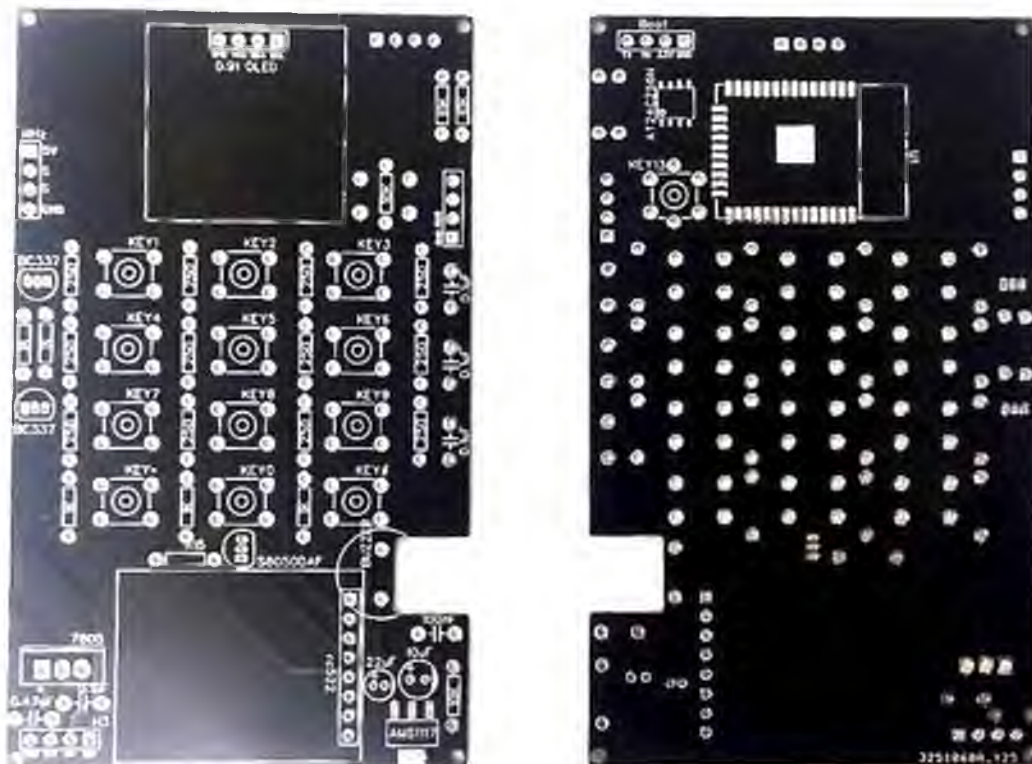


Рис. 2.12 Готова друкована плата головного пристрою



Рис. 2.13. Готова друкована плата допоміжного пристрою

Після отримання друкованих плат та компонентів можна провести монтаж. Монтаж компонентів на друковану плату – процес складається з механічного

з'єднання деталей та електронних компонентів у послїдовностї, що забезпечує їх необхідне розташування та взаємодїю для забезпечення встановлених технічних вимог.

Загальнї вимоги до робочого мїсця. Основи безпеки:

- робоче мїсце (стїл) не має бути захарашений. На вільному столї працювати приємнїше та ефектнвнїше. Крім того, радїодеталї не зможуть легко зарубитися в навколишньому мотлоху;

- робоче мїсце має бути добре освїтлене;

- пїд час паяння передбачте хорошу вентиляцїю робочого мїсця;

- не можна допускати дотику пальцїв до жалу паяльнїка або фєну;

- пїсля паяння потрїбно змити увесь фїюс є плати.

Результат готової розпаянї плати зображено на рис. 2.14.



Рис. 2.14. Розпаянї пристрої

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ МІКРОКОНТРОЛЕРІВ

Основною метою будь-якого розробника програмних систем є написання ефективної програми. Ефективного у всіх сенсах: надійного, зручного, розширюваного. Тому, чим зручніший і зрозуміліший підхід, тим менше доведеться розробнику зробити помилок, менше витратити часу на написання програми, простіше її зрозуміти і розширити.

На сьогоднішній день для написання будь-яких програм часто використовується об'єктно-орієнтований підхід, що дозволяє абстрагуватися розробнику від алгоритму, як послідовності вказівок. Об'єктно-орієнтований підхід пропонує програмісту представляти абстрактні сутності своєї програми як взаємодіючих між собою об'єктів. Такий підхід ближчий і зрозумілий людині, оскільки найбільш зручно відбиває навколишній реальний світ. Об'єктне мислення дозволяє представляти і розуміти дедалі складніші системи.

У гонитві за подальшою ефективністю та зниженням часу розробки програмних систем, одного існування такого підходу та відповідних об'єктно-орієнтованих мов програмування виявляється мало. Потрібні спеціальні засоби, що допомагають продумати, промоделювати систему, уникнути фатальних помилок, що виявляються, коли безліч рядків програмного коду вже написано. Потрібні засоби, що дозволяють візуально зобразити об'єкти та їх взаємодії, потрібні методики, що дозволяють послідовно знайти та вивчити взаємодіючі об'єкти, продумати процес розробки та адаптувати його до потреб, що змінюються.

Моделі дозволяють наочно продемонструвати структуру та поведінку системи. Вони необхідні для візуалізації та управління архітектурою системи, мінімізації ризиків. Моделі дозволяють досягти кращого розуміння систем, що призводить до їх спрощення та можливостей повторного використання. Система може бути описана з різних точок зору, навіщо використовуються різні моделі,

кожна з яких є семантично значущою абстракцією системи. Розрізняють структурні моделі, що представляють організацію системи, та поведінкові, що відображають її динаміку.

Необхідність універсализації підходів до побудови моделей призвела до створення у 1997 році спеціальної мови для опису. Універсальна мова моделювання (UML = Unified Modeling Language) набула широкого поширення і була вчасно стандартизована [15].

З появою стандарту в моделюванні виникли CASE-засоби, що дозволяють візуалізувати цей процес, поєднати моделі з документацією і навіть генерувати частини програмного коду. Одним із найвідоміших засобів цього типу є Rational Rose компанії Rational. Створений авторами UML цей CASE-засіб найповніше підтримує нотатції мови та її діаграми. Воно дозволяє організувати генерацію заголовків класів та деяких найпростіших функцій на основі створених моделей, що дозволяє вносити навіть суттєві зміни до існуючої структури системи. Універсальність мови моделювання дозволяє згенерувати дрілянки коду та описів будь-якою об'єктно-орієнтованою мовою C#, Java, C++, PHP та інших.

3.1. Моделювання поведінки системи

Для моделювання системи була використана діаграма DFD, яка наочно відображає перебіг інформації в межах процесу чи системи. Для зображення вхідних та вихідних даних, точок зберігання інформації та шляхів її пересування між джерелами та пунктами доставки у таких діаграмах застосовуються стандартні фігури, такі як прямокутники та кола, а також стрілки та короткі текстові мітки. Діаграми DFD варіюються від найпростіших нарисів процесів (включаючи намальовані вручну) до детальних багаторівневих схем з глибоким аналізом способів обробки даних. Діаграми DFD застосовуються для аналізу існуючих та моделювання нових систем. У найкращих традиціях візуалізації

даних діаграми DFD часто наочно «розповідають» про процеси, які складно пояснити словами і дозволяють ефективно донести інформацію до «фізиків», і до «ліриків», тобто до всіх учасників організації від розробників до генеральних директорів. Ось чому діаграми DFD не втратили популярності за довгі роки існування. Однак варто згадати, що хоча діаграми DFD відмінно підходять для програм і систем потоків даних, в наші дні вони далеко не завжди відповідають вимогам програмного забезпечення та систем, орієнтованих на інтерактивність, роботу в реальному часі та бази даних [16].

Насправді DFD близька до IDEF, оскільки також використовує принцип ієрархії та декомпіляції, але на відміну від IDEF є основним засобом моделювання функціональних вимог проєктованої системи. Метою створення такої діаграми є демонстрація того, як кожен процес перетворює свої вхідні дані на вихідні, а також виявити відносини між цими процесами. Ця діаграма зображує систему може «як буде». Перша діаграма з номером 0 називається контекстною і описує модельовану систему в загальному вигляді.

СКУД, крім здійснення автоматизованого пропускового режиму, надає не менш важливу для будь-якого підприємства функцію – обліку робочого часу працівників за допомогою формування звітів. Саме цей процес було узято за основний у контекстній діаграмі, зовнішніми сутностями в даному випадку будуть електронна перепустка та турнікет з одного боку, оператор СКУД з іншого. Діаграма DFD0 представлена рис. 3.1.



Рис. 3.1. DFD0 діаграма

Далі виконується процес декомпозиції, що полягає у розбитті основного процесу на функціональні підсистеми з дотриманням принципу ієрархії. Щоб система змогла сформувати звіт, повинен відбутися наступний ланцюжок подій: зчитування інформації з електронного пропуску співробітника, реєстрація події в системі (вхід/вихід або відмова) і після запиту на формування звіту відбувається безпосередньо сам процес формування звіту. Отже, після декомпозиції діаграми DFD0 ми отримуємо діаграму DFD1, представлену рис. 3.2 і ще з трьох підсистем.



Рис. 3.2. DFD діаграма 1 рівня

Процес формування звіту декомпозуємо до DFD діаграми другого рівня, в якій покажемо, що для створення звіту необхідні критерії пошуку співробітників і зазначений період. Діаграма DFD2 показана рис. 3.3.

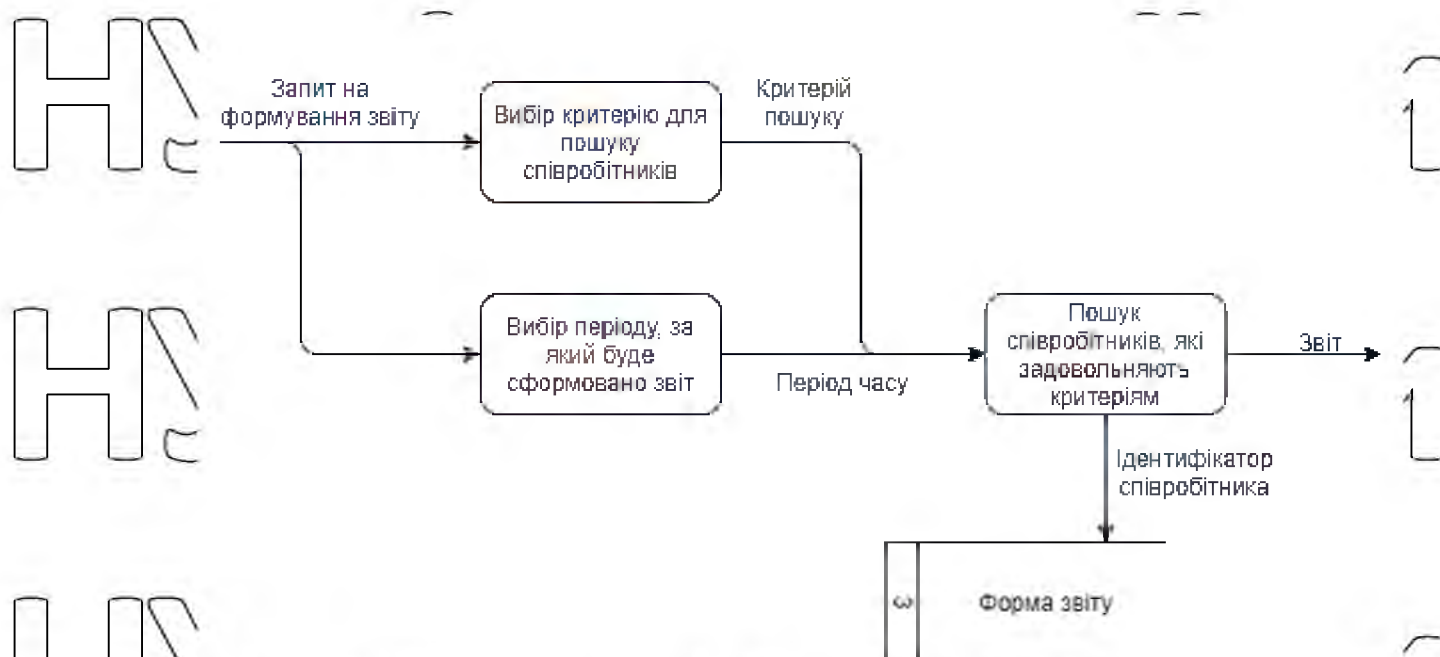


Рис. 3.3. DFD діаграма 2 рівня

3.2. Моделювання структури системи

Як уже зазначалося вище, основним напрямом розвитку СКУД є їх інтелектуалізація, агрегування максимально можливої кількості функцій зі збору, обробки інформації та прийняття рішень. Системи КУД здатні автоматизувати безліч процесів, що з організацією доступу до ресурсу Сюди входять реєстрація суб'єктів (користувачів та персоналу) та об'єктів (ресурсів) СКУД, безпосереднє надання доступу до ресурсу, організація контролю роботи персоналу, збирання та надання статистики про функціонування системи та багато іншого. У автоматизації маркетингу системи контролю доступу розширюють свої можливості з допомогою організації системи рахунків і платежів використання ресурсу.

СКУД, орієнтовані на обслуговування великої кількості клієнтів, зазвичай мають модульну структуру, що дозволяє організувати робочі місця різних служб, які забезпечують ефективне функціонування системи. Модульна схема

забезпечується за рахунок використання архітектури клієнт-сервер. Кількість та функціональність модулів залежать від призначення системи та виробника. Наприклад, набір модулів може бути таким:

- «Бюро перепусток» – робоче місце менеджера по роботі з клієнтами.

Ця служба займається реєстрацією нових клієнтів, видачею електронних карток, створенням рахунків, призначенням прав доступу групам та окремим користувачам.

- «АРМ касира» – спеціалізоване робоче місце, призначене прийому платежів використання послуг, наданих системою.

- «АРМ оператора» – робоче місце оператора (охоронця). Оператор відповідає за коректне функціонування системи протягом своєї зміни. Він контролює роботу системи, реагує на позаштатні ситуації.

- «АРМ адміністратора» – Адміністратор здійснює налаштування системи, прийом карт-ідентифікаторів, реєстрацію персоналу.

- «Генератор звітів» – Використовується для створення різних звітів про роботу системи.

Для невеликих систем, де роль обслуговуючого персоналу грає лише одна людина, вся необхідна функціональність може бути зведена в єдиний модуль.

Модулі (службові додатки) взаємодіють із центральним сервером СКУД (див. рис.3.4), який виконує роль деякого диспетчера, який обробляє запити додатків та події контролерів, датчиків та інших виконавчих пристроїв.

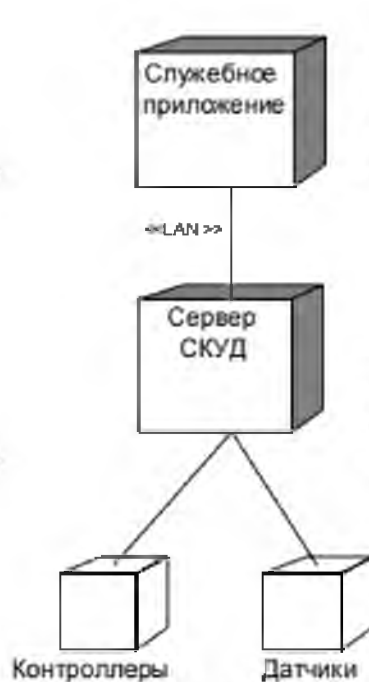


Рис. 3.4. Діаграма розгортання СКУД

Як середовище взаємодії службових додатків та сервера СКУД можуть виступати локальна обчислювальна мережа або адресний простір одного комп'ютера.

3.2 Розгортання та налаштування системи

Щоб почати працювати з ESP32, потрібно встановити драйвер CH340 і Arduino IDE або VSCode. Перед тим, як завантажити програму, потрібно встановити режим роботи мікроконтролера завантаження коду (Upload Using), задати потрібну частоту (CPU frequency), вибрати розмір флеш пам'яті (Flash Size), задати швидкість передачі (Upload Speed) і вибрати потрібний порт.

Налаштування зображені на рис. 3.5.

Можна додатково завантажити і встановити привласни кодів для мікроконтролера. Для цього скачайте файл з кодами потрібно розпакувати за адресою `\arduino\examples\`. Потрібно перезавантажити Arduino IDE, і в файл

НУБІП України

прикладі з'являться нові коди, які можна використовувати в своїх проектах. Щоб перевірити, чи правильно все підключено, можна запустити скетч для миготіння світлодіодом.

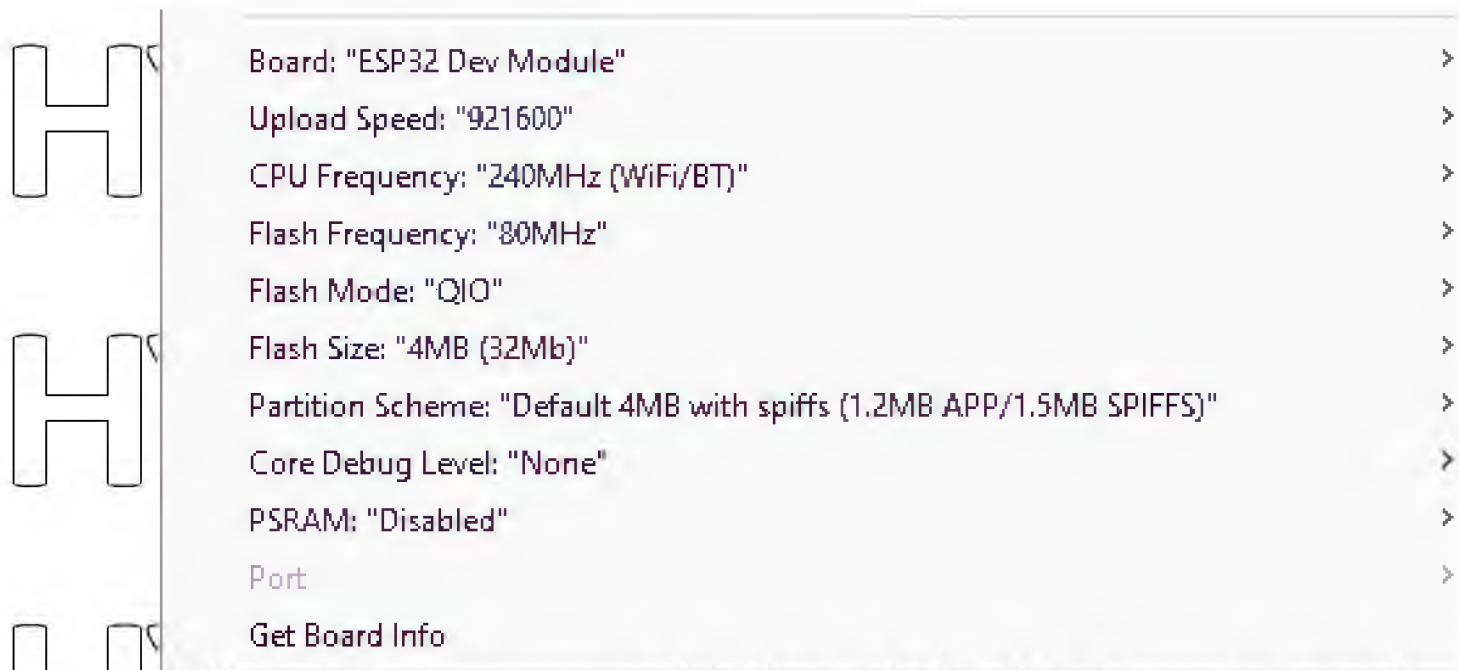


Рисунок 3.5 – Налаштування Arduino IDE

Після налаштування середовища розробки системи, можна завантажити програму. Але для завантаження треба додатково встановити усі необхідні бібліотеки, які можна завантажити у вільному доступі на ресурсі GitHub, а саме:

- бібліотеку для роботи с JSON;
- бібліотеку для роботи с SPI протоколом;
- бібліотеку для роботи с HTTP запитами;
- бібліотеку для роботи с RFID картками;
- бібліотеку для роботи с LED екраном

На рис. 3.6 приведено приклад підключення бібліотек.

НУБІП України

```

1 //Created by Habatynchik
2 //-----
3
4 #include <Arduino.h>
5 #include <SPI.h>
6 #include <Wire.h>
7 #include <Adafruit_GFX.h>
8 #include <Adafruit_SH1106.h>
9 #include <Tone32.h>
10 #include <MFRC522.h>
11 #include <RCSwitch.h>
12 #include <HTTPClient.h>
13 #include <ArduinoJson.h>
14
15 //-----
16
17 #define btnSignal1 25
18 #define btnSignal2 26
19 #define btnSignal3 27
20 #define buzzer 33
21
22 //-----
23

```

Рис. 3.6. Приклад роботи бібліотеки QRCode

Для того, щоб плата мала можливість відправляти запити на сервер треба встановити модуль `HTTPClient.h`, після чого мікроконтролер отримує можливість передавати інформацію на сервер та навпаки – отримувати її. Зв'язок відбувається за допомогою REST API запитів. Так називається спосіб взаємодії та обміну даними сервера. Більшість великих компаній розробляють цей інтерфейс для внутрішнього використання або своїх клієнтів. Така технологія здатна забезпечити повідомлення між двома системами. Зараз цей підхід зумів витіснити практично решту, включаючи дизайни, які були засновані на SOAP.

Це англійська аббревіатура, яка розшифровується та перекладається як передача стану уявлення. Web-служби, які користуються системою `Representational State Transfer`, застосовують термін `RESTful`. Відмінність цього архітектурного стилю від інших полягає в тому, що він не має єдиного стандарту, проте при цьому допустимо використовувати XML, HTTP, JSON та URL.

Суть роботи алгоритму полягає у парі дій, залежно від типу запиту. Від роботи сервера залежить функціонал та можливості архітектури. Є 4 основних види щодо інформації:

- get – отримання, просто передача;
- delete - Видалення, надалі вони не відображаються;
- post - реєстрація або додавання, реєстрація;
- update - оновлення, регулярна операція, бази стають актуальними та свіжими.

Лістинг 3.1 Передача Get запитів

```
#include <HTTPClient.h>

const char* ssid = "WiFi SSID";
const char* password = " WiFi Password";

void loop() {
  Serial.println(getRequest(hostingURL));
  //getCurrentTime();
  delay(5000);
}

String getRequest(String request){
  HTTPClient http;
  http.begin(request);
  String response = "null";
  int httpCode = http.GET();
  if (httpCode > 0) { //Check the returning code
    response = http.getString(); //Get the request response
  }
  http.end(); //Close connection
  return response;
}
```

Як пакет зазвичай відправляється JSON масив на вказану конкретну URL-адресу. Там спрацьовує так звана функція, а залежно від вже надісланих даних та поточного запиту починається певна дія. При цьому не має значення, з якого пристрою надіслана інформація - мобільний додаток або браузер комп'ютера.

Після написання коду потрібно підключити SSD1306 до мікроконтролера. На рис. 3.7 зображено взаємозв'язок між мікроконтролера та екрану. Також для роботи з дисплеєм потрібно встановити бібліотеку Adafruit.

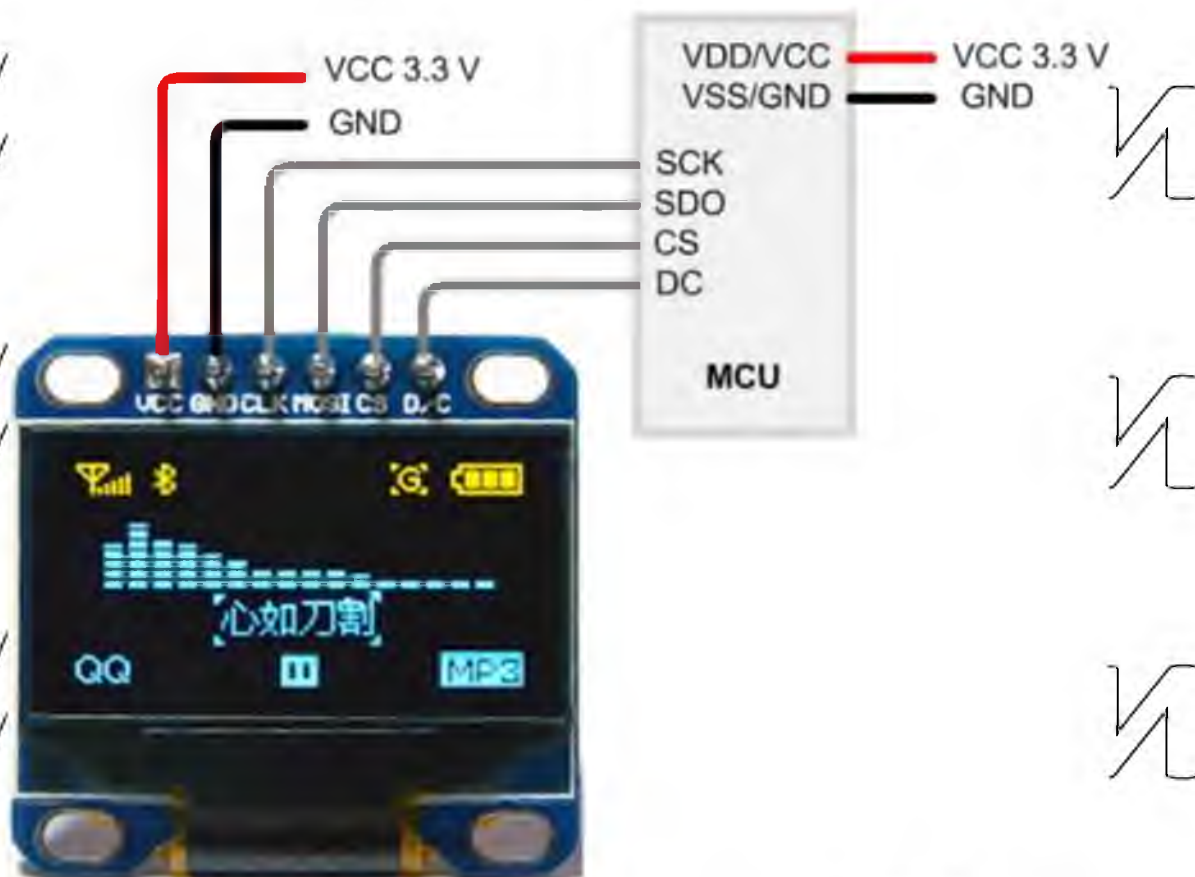


Рис. 3.7. Схема підключення SSD1306 до мікроконтролера

SSD1306 є єднокристальним OLED/PLED драйвером з контролером для органічних/полімерних світлодіодних графічних дисплеїв. SSD1306 має вбудоване керування контрастністю, оперативною пам'яттю та внутрішній генератор, що знижує кількість зовнішніх компонентів та споживання енергії. 256 кроків регулювання яскравості. Має 6800/8000 серії сумісний паралельний інтерфейс, послідовний периферійний інтерфейс I2C або SPI. Це підходить для багатьох компактних портативних пристроїв, таких як допоміжний дисплей для мобільних телефонів, MP3-плеєр та калькулятори, і т.д [17].

Останнє, що потрібно налаштувати в системі, це зберігання користувачів у пам'яті мікроконтролера. Зберігання відбувається за допомогою чипу AT24C256N.

EEPROM – пам'ять, до якої маємо повний доступ із виконуваної програми, тобто, можна під час виконання читати та писати туди дані, і ці дані не скидаються під час перезавантаження. Існує декілька варіантів використання пам'яті:

- зберігання параметрів, що змінюються “з меню” пристрою, без перепрошивки;
- калібрування, збереження калібрувальних даних;
- використання як додаткової SRAM пам'яті у разі її нестачі;
- “чорна скринька” – постійний запис показань із датчиків для подальшого розшифрування збоїв;
- запис стану робочого процесу відновлення роботи після раптового перезавантаження.

Єдиний недолік: EEPROM має ресурс за кількістю перезапису комірок пам'яті. Виробник гарантує 100 000 циклів запису кожної комірки, за фактом ця кількість залежить від конкретного чіпа та температурних умов, незалежні тести показали 3-6 мільйонів циклів перезапису при кімнатній температурі до появи першої помилки, тобто, заявлені 100 тисяч взяті з дуже великим запасом. Але є невелике уточнення – при заявлених 100 000 циклах перезапису гарантується збереження записаних даних протягом 100 років за температури 24°C, якщо перезаписувати по мільйону – дані зіпсуються швидше. У той же час кількість читань кожної комірки необмежена [18].

Після вивчення після підключення усіх модулів та налаштування їх в мікроконтролеру можна зробити висновок, що система функціонує згідно з технічним завданням, усі модулі виконують свої функції та не конфліктують один з одним. Надалі для закріплення результату потрібно провести тестування системи.

3.4 Тестування працездатності системи

Для тестування системи було прийнято рішення провести тестування на серверній стороні для того, щоб перевірити зв'язок мікроконтролера та серверного додатку.

Тести повинні бути стабільними і мінімально залежати від оточення. Якщо при внесенні змін, тести падають, швидше за все вони хороші. І навпаки, якщо не падають, ймовірно вони не дуже хороші.

3 коробки Laravel підтримує 3 типи тестів:

```
Browser;
Feature;
Unit.
```

У більшості випадків треба писати окремих модулний тест для кожного випадку, визначеного у запиті форми тестування. Це призводить до великої кількості тестів, таких як `test_request_without_title` і `test_request_without_content`. Всі ці методи реалізуються точно так же, тільки викликаючи вашу кінцеву точку з деякими іншими даними. Це призведе до великої кількості дублікатів коду [19].

Для цього прикладу я буду робити запит форми, щоб зберегти продукт.

Лістинг 3.2 – Створення тесту

```
php artisan make:request SaveProductRequest
```

Згенерований файл класу буде розміщена в `App/HTTP` і запити. Після чого треба оголосити набір правил перевірки для цієї форми запиту. Параметр `title` повинен являти собою рядок, що містить не більше 50 символів. Параметр `price` повинен бути числовим. На даний момент це єдині два правила перевірки [20].

Клас запити `SaveProduct` наведено у лістингу 3.3.

Листинг 3.8 Приклад Unit тесту

```

<?php
namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class SaveProductRequest extends FormRequest
{
    public function authorize()
    {
        return true;
    }

    public function rules()
    {
        return [
            'title' => 'required|string|max:50',
            'price' => 'required|numeric',
        ];
    }
}

```

В рамках методу авторизації можете перевірити, чи є у користувача дозвіл на виконання цього запиту. Наприклад, можливо перевірити, чи є користувач адміністратором. На рис. 3.8 зображено результат тестування.



```

Time: 741 ms, Memory: 26.00 MB
OK 16 tests, 13 assertions

```

Рис. 3.8. Результат виконання Unit тесту

В тесті, треба вказати дані, з якими викликається наша кінцева точка. Потім треба запустити кожен з цих масивів даних через клас валідатора Laravel, щоб перевірити, чи проходять правила валідації.

Найважливіше тут – це структура постачальника даних. У Ключі масиву постачальників даних треба вказати ім'я тесту. У цьому масиві є два атрибути: пройдені тести і дані. Переданий атрибут – це логічне значення з очікуваним результатом роботи валідатора. Атрибут data містить дані, які ми хочемо відправити в кінцеву точку.

Листинг 3.4. Приклад валидатора validationProvider

```

<?php
namespace Tests\Feature\App\Http\Requests;
use App\Http\Requests\SaveProductRequest;
use Faker\Factory;
use Tests\TestCase;
use Illuminate\Foundation\Testing\RefreshDatabase;
class SaveProductRequestTest extends TestCase
{
    use RefreshDatabase;
    /** @var \App\Http\Requests\SaveProductRequest */
    private $rules;

    /** @var \Illuminate\Validation\Validator */
    private $validator;
    public function setUp(): void
    {
        parent::setUp();
        $this->validator = app()->get('validator');
        $this->rules = (new SaveProductRequest()->rules());
    }
    public function validationProvider()
    {
        /** WithFaker trait doesn't work in the dataProvider */
        $faker = Factory::create(Factory::DEFAULT_LOCALE);
        return [
            'request_should_fail_when_no_title_is_provided' => [
                'passed' => false,
                'data' => [
                    'price' => $faker->numberBetween(1, 50)
                ]
            ],
            'request_should_fail_when_no_price_is_provided' => [
                'passed' => false,
                'data' => [
                    'title' => $faker->word()
                ]
            ],
            'request_should_fail_when_title_has_more_than_50_characters' => [
                'passed' => false,
                'data' => [
                    'title' => $faker->paragraph()
                ]
            ],
            'request_should_pass_when_data_is_provided' => [
                'passed' => true,
                'data' => [
                    'title' => $faker->word(),


```

```

        'price' => $faker->numberBetween(1, 50)
    ];
}
/**
 * @test
 * @dataProvider validationProvider
 * @param bool $shouldPass
 * @param array $mockedRequestData
 */
public function validation_results_as_expected($shouldPass, $mockedRequestData)
{
    $this->assertEquals(
        $shouldPass,
        $this->validate($mockedRequestData)
    );
}
protected function validate($mockedRequestData)
{
    return $this->validator
        ->make($mockedRequestData, $this->rules)
        ->passes();
}

```

На рис. 3.9 зображено результат тестування.



```

Time: 666 ms, Memory: 24.00 MB
OK (6 tests, 6 assertions)

```

Рис. 3.9. Результат виконання Unit тесту

Для тестування системи на апаратній стороні було використано Google C++ Testing, що дозволило переконатися у тому, що система повністю працездатна та не потребує змін.

Google C++ Testing Framework – бібліотека для модульного тестування мовою C++.

Google Test побудована на методології тестування xUnit, тобто, коли окремі частини програми (класи, функції, модулі) перевіряються окремо один від

НУБІП УКРАЇНИ
одного, в ізоляції. Бібліотека сама по собі розроблена з активним застосуванням тестування, коли при додаванні будь-яких частин до офіційної версії, крім коду самих змін, необхідно написати набір тестів, що підтверджують їх коректність [21].

НУБІП УКРАЇНИ
Результат тестування дає інформацію, що всі тести проходять, але дублювання скорочується, а ремонтпридатність поліпшується.

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

ВИСНОВКИ

НУБІП України

В роботі спроектовано пристрій для контролю доступу до приміщення, розроблено програмний продукт для взаємодії з серверним додатком, написаний мовою програмування C++, встановлений на мікроконтролері ESP32, представлено основні схеми, структуру, схеми та вихідний код програмного забезпечення.

НУБІП України

Забезпечення найвищого рівня безпеки всіх виробничих процесів – важливе завдання для функціонування будь якого підприємства. Несанкціоновані проходи, порушення пропускового режиму і трудової дисципліни, нецільове використання робочого дня – це несе потенційну загрозу, здатну призвести до істотних матеріальних витрат.

НУБІП України

Для забезпечення контролю доступу на підприємстві необхідно впровадження сучасної СКУД, до складу якої входять різні засоби для аутентифікації (зчитувачі карток, біометричні зчитувачі та ідентифікатори), контролери та виконавчі пристрої (турнікети, ворота тощо). Кожному працівнику визначається рівень доступу, видаються персональні ідентифікатори, за допомогою яких вони можуть проходити на територію банку та до приміщень, де мають право перебувати.

НУБІП України

Автоматизація дозволяє знизити витрати, підвищити ефективність роботи, досягти можливостей, недоступних раніше під час використання паперового документообігу. Користь від автоматизації діяльності відчувається у разі, якщо автоматизується вся діяльність підприємства, весь комплекс його завдань.

НУБІП України

У ході роботи досліджено можливості використання сучасних технологій для проектування та використання мікроконтролерів в системах контролю доступу до приміщень. Було проведено аналіз, проектування, розробку та впровадження автоматизованої системи, що відповідає вимогам сучасного ринку.

НУБІП України

НУБІП України

ЦЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

НУБІП України

1. Access control units and systems. Classification. General technical requirements. Test methods URL: <https://docs.cntd.ru/document/1200071688> (дата звернення 01.11.2021).

НУБІП України

2. Rule-based Access Control URL: <https://www.idcubesystems.com/rule-based-access-control/> (дата звернення: 10.11.2021).

3. PERCo-Web Access control system URL:

НУБІП України

<https://www.perco.com/products/perco-web-access-control-system/> (дата звернення: 15.11.2021).

4. Wiegand interface URL: https://en.wikipedia.org/wiki/Wiegand_interface

(дата звернення: 15.10.2021).

5. Life Cycle Management URL: <https://www.lifecycleinitiative.org/starting-life-cycle-thinking/life-cycle-approaches/life-cycle-management/>

НУБІП України

(дата звернення: 15.10.2021).

6. ESP32 Technical Reference Manual URL:

НУБІП України

[https://www.espressif.com/sites/default/files/documentation/esp32_technical_referen](https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf)
[ce_manual_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_technical_referen) (дата звернення: 10.11.2021).

7. Гамаюнов Н.Д., Редько В.П., Ляхно В.А. Підключення багатьох модулів до ESP32 по протоколу SPI [Електронний ресурс] // Міжнародна науково-практична конференція студентів, аспірантів та молодих вчених

"Інформаційні технології: економіка, техніка, освіта": [сайт]. [2020]. URL: <https://bit.ly/39HYQV3>

8. AT24C256N-10SI-2.7 Datasheet URL:

НУБІП України

<https://eu.mouser.com/datasheet/2/268/doc0670-1180619.pdf> (дата звернення: 10.11.2021).

9. Резисторы постоянные неспроволочные общего применения С2-33

НУБІП України

URL: https://www.erkon-nn.ru/upload/model_library/S2-33/s2-33_1.pdf (дата звернення: 25.10.2021).

10. Конденсаторы оксидно-электролитические алюминиевые К50-35, К50-35А, К50-35Б, ОЖ0.464.214 ТУ URL: <https://eandc.ru/pdf/kondensator/k50-35.pdf> (дата звернення: 25.10.2021).

11. Транзистор BC337 URL: <https://shematok.ru/transistor/bc337> (дата звернення: 18.09.2021).

12. S8050 circuit Datasheet URL: <https://pdf1.alldatasheet.com/datasheet-pdf/view/172696/UTC/S8050.html> (дата звернення: 17.09.2021).

13. AMS1117 integrated circuit Datasheet URL: <http://www.advanced-monolithic.com/pdf/ds1117.pdf> (дата звернення: 22.08.2021).

14. Gerber format URL: https://en.wikipedia.org/wiki/Gerber_format (дата звернення: 23.10.2021).

15. Unified Modeling Language documentation URL: https://en.wikipedia.org/wiki/Unified_Modeling_Language (дата звернення: 17.09.2021).

16. What is a Data Flow Diagram URL: <https://www.lucidchart.com/pages/data-flow-diagram> (дата звернення: 10.09.2021).

17. SSD1306 circuit - Adafruit Industries Datasheet URL: <https://cdn-shop.adafruit.com/datasheets/SSD1306.pdf> (дата звернення: 10.10.2021).

18. EEPROM URL: <https://en.wikipedia.org/wiki/EEPROM> (дата звернення: 15.11.2021).

19. Laravel documentation URL: <https://laravel.com/docs/8.x/artisan> (дата звернення: 11.10.2021).

20. Релько В.П., Гамаюнов Н.Д., Лахно В.А. Підключення технології WebSocket для обміну інформації в реальному часі [Електронний ресурс] // "Теоретичні та прикладні аспекти розробки комп'ютерних систем". Науково-практична конференція студентів і аспірантів.". [сайт]. [2021]. URL: <https://bit.ly/3uYRQuZ>

21. GoogleTest User's Guide URL: <https://google.github.io/googletest/> (дата звернення: 11.10.2021).

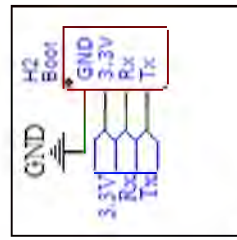
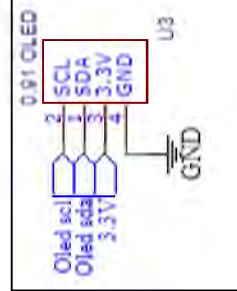
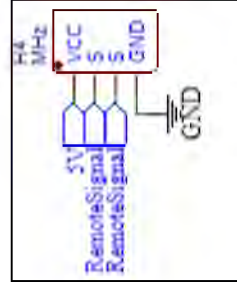
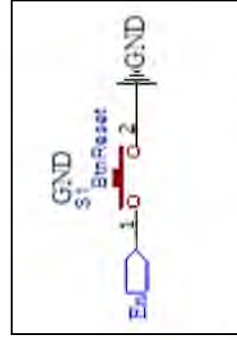
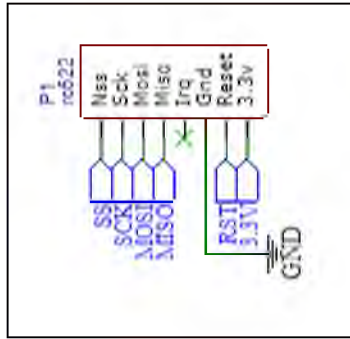
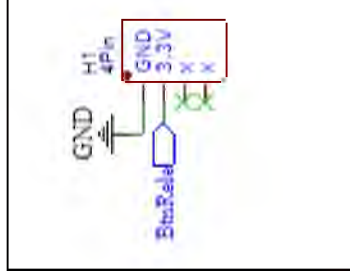
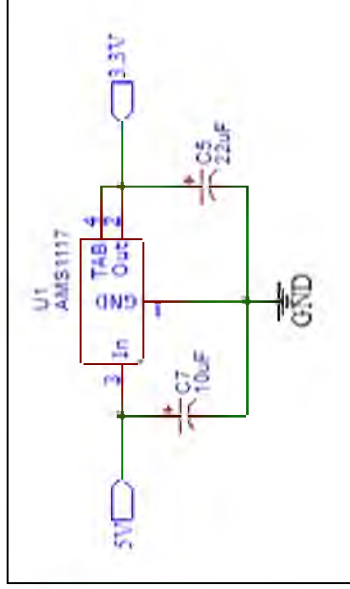
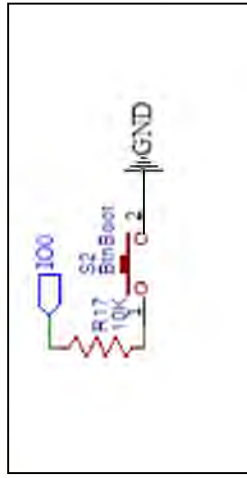
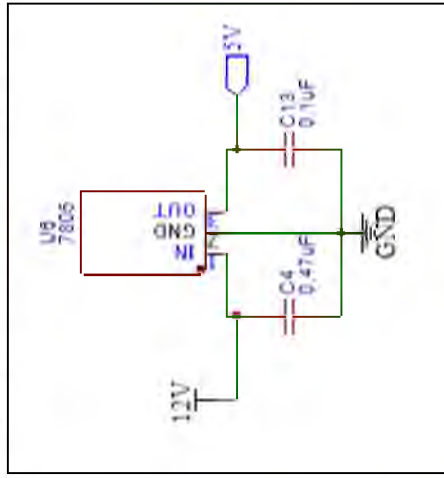
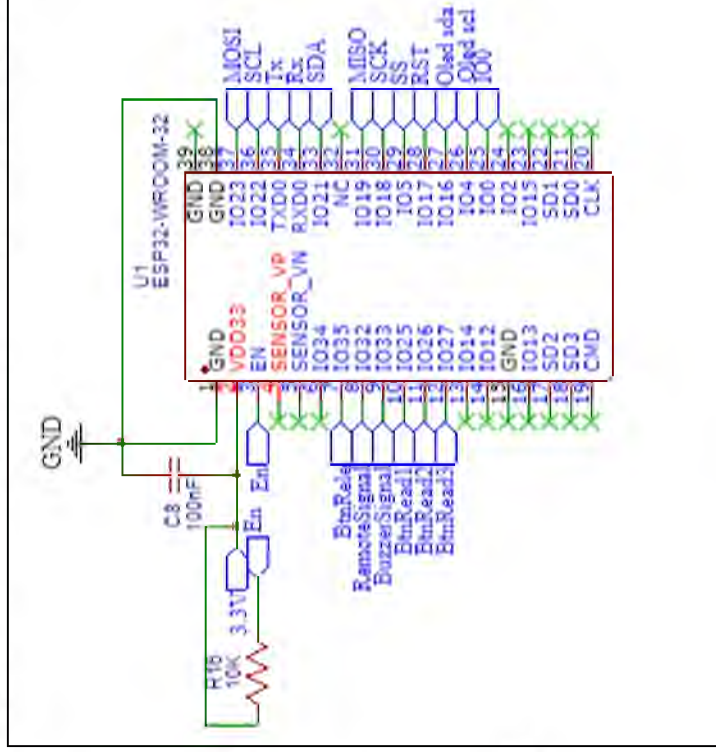
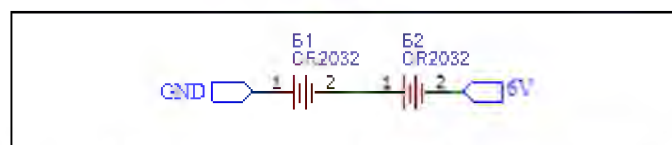
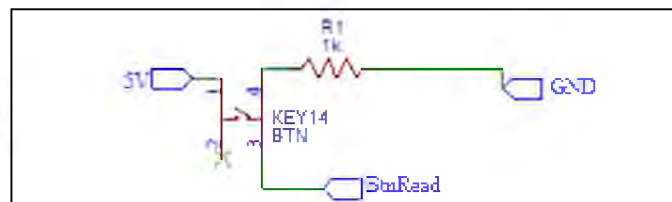
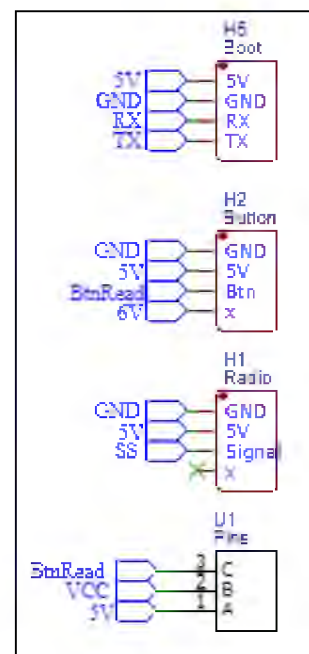
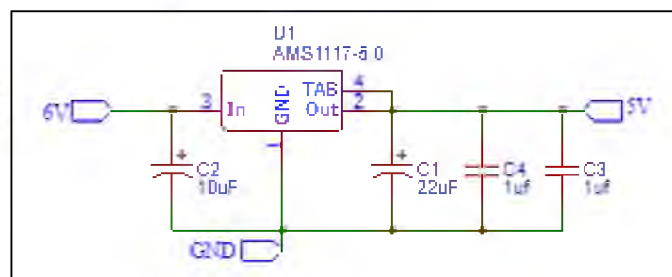
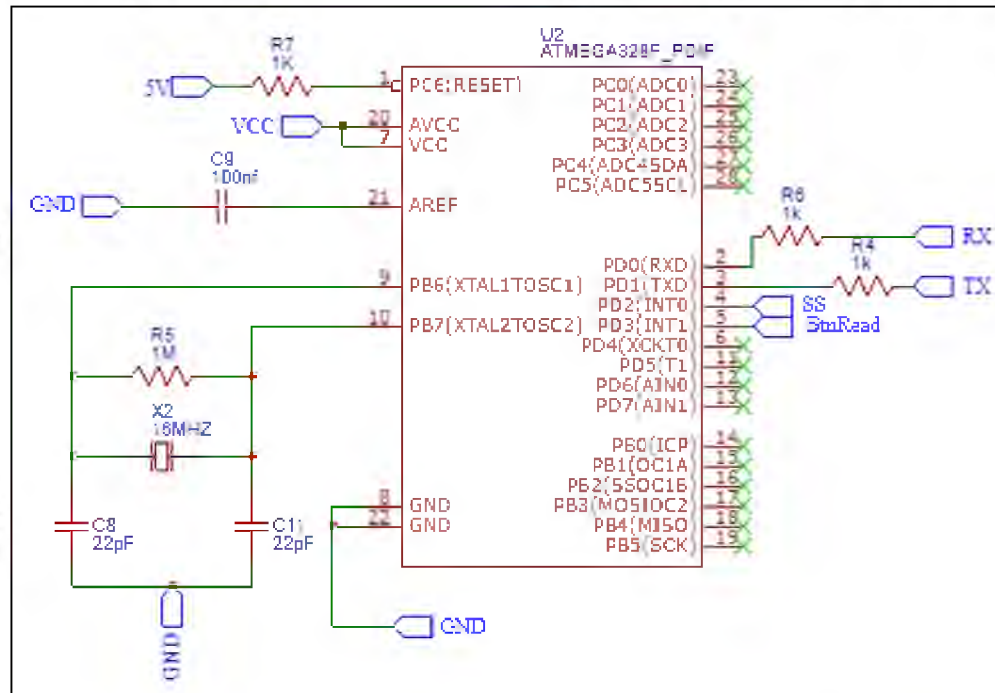


Схема електронної плати

Сторінка 1



НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП УКРАЇНИ

```
// Created by Habatynchik
// -----
```

```
#include <Arduino.h>
#include <SPI.h>
#include <Wire.h>
```

НУБІП УКРАЇНИ

```
#include <Adafruit_GFX.h>
#include <Adafruit_SSH106.h>
#include <Tone32.h>
#include <MFRC522.h>
#include <RCSwitch.h>
#include <HTTPClient.h>
#include <ArduinoJson.h>
```

НУБІП УКРАЇНИ

```
// -----
/*
#define OLED_SDA 12
#define OLED_SCL 14
```

```
SSD1306Wire display(0x3c, SDA, SCL);
```

НУБІП УКРАЇНИ

```
/*
#define btnSignal1 25
#define btnSignal2 26
#define btnSignal3 27
#define buzzer 33
```

НУБІП УКРАЇНИ

```
// -----
const char *ssid = "Republic of Belarus 2.4G";
const char *password = "Misterx13579";
const char *serverName = "http://ta-za-sho.me";
```

```
DynamicJsonDocument doc(1024);
```

НУБІП УКРАЇНИ

```
unsigned long lastTime = 0;
unsigned long timerDelay = 5000;
// -----
```

```
#define RST_PIN 17 // Configurable, see typical pin layout
above
```

```
#define SS_PIN 5 // Configurable, see typical pin layout
above
```

```
#define RemoteSignal 27 // Configurable, see typical pin layout
above
```

НУБІП УКРАЇНИ

```
const int piezoPin = 33;
MFRC522 rfid(SS_PIN, RST_PIN); // Instance of the class
```

NUBІП УКРАЇНИ

```
MFR522::MIFARE_Key key;  
byte nuidPICC[4]; // Init array that will store new NUID  
RCSwitch mySwitch = RCSwitch();  
/*
```

NUBІП УКРАЇНИ

```
void fillRect(void) {  
  uint8 t color = 1;  
  for (int16 t i = 0; i < display.getHeight() / 2; i += 3) {  
    display.setColor((color % 2 == 0) ? BLACK : WHITE); //  
    alternate colors  
    display.fillRect(i, i, display.getWidth() - i * 2,  
display.getHeight() - i * 2);  
    display.display();  
    delay(10);  
    color++;  
  }  
  // Reset back to WHITE  
  display.setColor(WHITE);  
}
```

NUBІП УКРАЇНИ

```
*/  
void setup()  
{  
  /*
```

NUBІП УКРАЇНИ

```
fillRect();  
delay(1000);  
display.clear();  
*/  
Serial.begin(9600);  
pinMode(piezoPin, OUTPUT);
```

NUBІП УКРАЇНИ

```
Serial.println("setup");  
// Transmitter is connected to Arduino Pin #00  
mySwitch.enableReceive(0); // Optional set pulse length.  
// mySwitch.setPulseLength(320);  
// Optional set protocol (default is 1, will work for most  
outlets)  
// mySwitch.setProtocol(2);
```

NUBІП УКРАЇНИ

```
// Optional set number of transmission repetitions.  
// mySwitch.setRepeatTransmit(15);  
SPI.begin(); // Init SPI bus  
rfid.PCD_Init(); // Init MFR522
```

NUBІП УКРАЇНИ

```
-----  
-----  
WiFi.begin(ssid, password);  
// Serial.println("Connecting");  
while (WiFi.status() != WL_CONNECTED)  
{  
  delay(500);  
}
```

```
Serial.print(".");
}
Serial.println("");
Serial.print("Connected to WiFi network with IP Address: ");
Serial.println(WiFi.localIP());
//-----
```

```
}
bool checkCard()
{
  if (!rfid.PICC_IsNewCardPresent())
    return 0;
```

```
  if (!rfid.PICC_ReadCardSerial())
    return 0;

  for (byte i = 0; i < 4; i++)
    nuidPICC[i] = rfid.uid.uidByte[i];
  rfid.PICC_HaltA();
  rfid.PCD_StopCryptol();
```

```
  return 1;
}
void soundError()
{
  tone(piezoPin, 1000, 100, 0);
  delay(50);
  tone(piezoPin, 1000, 200, 0);
```

```
}
void soundAccess()
{
  tone(piezoPin, 2400, 200, 0);
}

```

```
String arrayToString()
{
  int tmp;
  String myString = "";
  for (size_t i = 0; i < 4; i++)
  {
```

```
    tmp = nuidPICC[i];
    myString += tmp;
    //tmp += nuidPICC[i];
  }
  return myString;
}
boolean sendCard()
```

```
boolean sensor = 0;
if (WiFi.status() == WL_CONNECTED)
{
  HTTPClient http;
```

```
    // Your Domain name with URL path or IP address with path
    http.begin("http://ta-za-sho.me/api/open");
    // Specify content-type header
    http.addHeader("Content-Type", "application/json");
    doc["card_number"] = arrayToString();
```

```
    String requestBody;
    serializeJson(doc, requestBody);
    //int httpResponseCode = http.POST(requestBody);
    if (http.POST(requestBody) > 0)
```

```
    {
      deserializeJson(doc, http.getString());
      //String response = http.getString();
      sensor = doc["success"].as<boolean>();
      // Serial.print("HTTP Response code: ");
      // Serial.println(httpResponseCode);
      //Serial.println(response);
      //Serial.println(sensor);
    }
```

```
  http.end();
}
else
{
  // Serial.println("WiFi Disconnected");
}
```

```
return sensor;
}
void loop()
{
```

```
  //Serial.println( mySwitch.getReceivedValue() );
  /*if (mySwitch.available()) {
    long gg = mySwitch.getReceivedValue();
    Serial.print( gg );
    if (gg == 1488)
```



```
soundAccess();  
Serial.println("Open");  
delay(200);
```

НУБІП України

```
mySwitch.resetAvailable();  
}*/  
if (checkCard())  
{  
//soundAccess();  
arrayToString();  
//sendCard();  
Serial.println(arrayToString());  
//Serial.println(sendCard());  
delay(300);  
}
```

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України