МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

ИНСТИТУТ ЭНЕРГЕТИКИ И ЭЛЕКТРОТЕХНИКИ
(наименование института полностью)
Промышленная электроника
(наименование кафедры)
11.03.04 Электроника и наноэлектроника
(код и наименование направления подготовки, специальности)
Промышленная электроника
(направление (профиль)/специальность)

БАКАЛАВРСКАЯ РАБОТА

на тему Система контроля доступа в аудиториях кафедры "Промышленная электроника"

Студент(ка) Л.С. Поляков _____
(И.О. Фамилия) (личная подпись)

Руководитель М.В. Позднов _____
(И.О. Фамилия) (личная подпись)

Допустить к защите

Заведующий кафедрой, к.т.н., доцент А.А. Шевцов _____
(ученая степень, звание, И.О. Фамилия) (личная подпись)

«_____»_____20_____г.

Тольятти 2019

Аннотация

Объем 87 с., 45 рис., 6 табл., 20 источников, 3 прил.

СИСТЕМА КОНТРОЛЯ ДОСТУПА В АУДИТОРИЯХ КАФЕДРЫ “ПРОМЫШЛЕННАЯ ЭЛЕКТРОНИКА”

Цель работы – разработка системы контроля доступа ограничивающий свободный проход посторонним лицам в аудиторию кафедры “Промышленная электроника”

Задачи работы заключались в:

1) Анализе современных решений
2) Разработки структурной и электрической принципиальной схемы
3) Трассировки и создание платы
4) Разработки программного кода микроконтроллера
5) Сборки устройства
6) Отладки устройства

Работа состоит из четырех глав, в которых решены упомянутые задачи.

Разработка устройства осуществлялась с помощью пакетов DipTrace, КомпасV16, sPlan 7.0. Написание программного кода велась в среде разработки Arduino IDE.

В процессе работы был создан рабочий прототип системы контроля доступа, для экспериментальных исследований и отладки работы устройства.

Область применения данного устройства являются кабинеты, огороженные помещения, аудитории.

Annotation

The volume of 87 p., 45 fig., 6 tab., 20 sources, 3 adj.

ACCESS CONTROL SYSTEM IN THE DEPARTMENT OF “INDUSTRIAL ELECTRONICS” DEPARTMENTS

The purpose of the work is the development of an access control system restricting free passage to unauthorized persons in the auditorium of the “Industrial Electronics” department

The tasks of the work were to:

1) Analysis of modern solutions

2) Structural and electrical concept development

3) Tracing and board creation

4) Develop the software code of the microcontroller

5) Build device

6) Debugging device

The work consists of four chapters in which the mentioned problems are solved.

The development of the device was carried out using the packages DipTrace, KompasV16, sPlan 7.0. Writing software code was carried out in the Arduino IDE development environment.

In the process, a working prototype of the access control system was created for experimental research and debugging the operation of the device.

The scope of this device are cabinets, fenced premises, and the audience.

СОДЕРЖАНИЕ

Введение 6
1. Постановка задач 8
1.1. Формирование задачи, определения целей, проверка актуальности проекта 8
1.2. Анализ современных решений, сравнения стоимости аналогов 9
1.3. Преимущества проекта над аналогами 12
2. Техническая часть 13
2.1 Разработка электрической структурной схемы 13
2.2. Выбор и обоснование элементной базы 16
2.3. Разработка электрической принципиальной схемы 25
2.4. Расчет системы контроля доступа 26
2.5. Расчёт надёжности системы контроля доступом 33
2.6. Расчёт стоимости печатного узла 36
2.7. Разработка печатной платы 39
2.8. Разработка программного обеспечения 43
2.8.1. Анализ и разработка алгоритма работы ПО 43
2.8.2. Написание программного кода 45
3. Технологическая часть 46
3.1. Сборка устройства 46
3.2. Запись программного обеспечения на микроконтроллер 49
4.Экспериментальная часть 50
4.1 Наладка готового устройства 50
4.2 Испытание готового устройства 51
4.3 Методика контроля технического состояния 53
4.4 Инструкция по эксплуатации устройства 64
Заключение 67
Список используемой литературы 68
Приложение А 70

Приложение Б 71
Приложение В 72

Введение

У человечества, за всё время существования, имелись потребности в контроле доступа на определённую территорию. С развитием науки и техники, инженерами разрабатывались всё более защищённые и технически сложные механизмы. Так, например, во втором тысячелетия до н.э. Египтяне изобрели первый замок, который блокировался посредством вертикальных штифтов-задержек [1]. На рисунке 1 изображена схема первого замка.

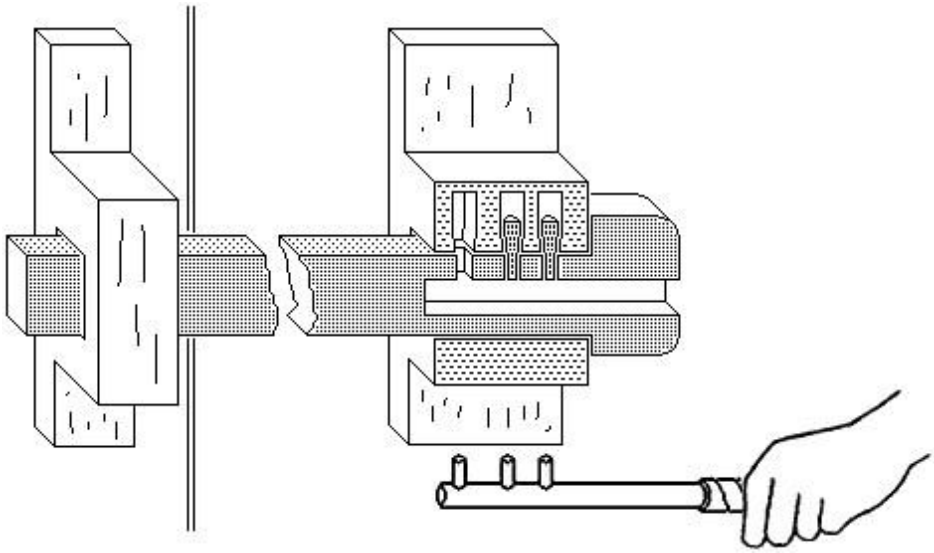


Рисунок 1 – изображение первого замка.

Первый навесной замок с охранной пружиной на дуге появился в 265-420 году н.э. в Китае. Использовался замок императорами, а также богатыми людьми и служил не только как средство защиты от воров, но и как вещью показывающее статус владельца. В культурах разных стран навесным замкам часто придавали формы различных амулетов, животных и оберегов. Считалось, что они способны привлекать счастье, удачу и богатство в дом.

В современных систем доступа применяется биометрия, как инструмент распознавания человека. Биометрия – это набор методов распознавания людей по разным физическим признакам и чертам. Например, для распознавания

могут использовать: роговицу человеческого глаза; отпечаток пальца; объёмное сканирование лица; запись голоса и т.д. [2].

Выпускная квалификационная работа (ВКР) на тему «Система контроля доступа в аудиториях кафедры "Промышленная электроника"» посвящен созданию устройству, которое предоставляет доступ в помещение определённым людям. В устройстве будет применён один из методов биометрии, а именно распознавание человека по отпечатку пальца. Помимо биометрии устройство так же будет предоставлять доступ по RFID меткам и обладать беспроводным доступом благодаря технологии Wi-Fi. Устройство не обойдётся и не без механических способов доступа, открыть дверь можно будет при помощи обычного ключа.

ПОСТАНОВКА ЗАДАЧ

1.1 Формирование задачи, определения целей, проверка актуальности проекта.

Системы доступа, будь то обычный амбарный замок или сложная биометрическая система по определению пользователя, всегда были актуальны у человечества, так как не позволяли посторонним лицам проникать на закрытую территорию. Особенно система доступа актуальна в местах с большим скоплением людей и препятствует как случайно зашедшему, так и злоумышленнику.

Главной целью работы является разработка системы контроля доступа ограничивающий свободный проход посторонним лицам в аудиторию кафедры “Промышленная электроника”

Устройство должно обладать как электронным, так и механическим устройством доступа, иначе в случае разряда аварийного аккумулятора или непредвиденного выхода из строя электронной части устройства, открыть замок будет возможно без помощи ключа. В электронную же часть будет входить: биометрия отпечатка пальца; доступ по RFID меткам; удалённый доступ по сети Wi-Fi.

Необходимо выполнить следующие задачи для достижения главной цели работы:

1) Анализ современных решений
2) Разработка структурной и электрической принципиальной схемы
3) Трассировка платы
4) Разработка программного кода микроконтроллера
5) Сборка устройства
6) Наладка устройства

1.2 Анализ современных решений, сравнения стоимости аналогов

В настоящее время в России нет дефицита электронных устройств доступа, поэтому были выбраны, для анализа, ходовые модели с похожими характеристиками, что и разрабатываемое устройство. Так же были учтены соотношения цены и качества.

На Российском рынке популярна система контроля и управления доступом (СКУД) фирмы Ашробот. У них имеются наборы комплектов с разных функционалов. Комплект под номером 27 подходит по функционалу по отношению к разрабатываемому устройству, соответственно анализировать будем его.

СКУД комплект 27 имеет сканер отпечатка пальца для опознавания пользователей. Так же получить доступ возможно при помощи RFID карты и обычного ключа. СКУД не имеет бесперебойного источника питания, его нужно докупать отдельно [3]. На рисунке 2 изображён СКУД комплект 27. В комплект входит: электромоторный врезной замок DJ05ST; биометрический считыватель со встроенным контроллером; блок питания.

Рисунок 2 – СКУД комплект 27.

Цена комплекта 11600 рублей. Так же к комплекту можно докупить Wi-Fi контроллер и иметь беспроводной доступ. Цена контроллера 4000 рублей.

Недостатком данного устройства является расположение управляющей платы устройства, которая находится в передней панели, что понижает защищённость. К недостаткам относится и отсутствие бесперебойного питания.

Из зарубежных аналогов дверных замков популярная модель Aqara Smart Lock производителя Xiaomi. Устройство выполнено в одном корпусе и питается от четырёх батареек типа АА. Имеет сканер отпечатка пальца, панель ввода цифрового пароля. Так же получить доступ возможно при помощи RFID карты и обычного ключа. На рисунке 3 изображён электронный замок Xiaomi Aqara Smart Lock. К преимуществам можно отнести множество способов получения доступа. Недостатком является питание от батареек. Цена составляет 13000рублей.

Рисунок 3 – Электронный замок Xiaomi Aqara Smart Lock.

Если же сравнивать в низком ценовом диапазоне, то тут будет модель фирмы Galo. Электронный замок от фирмы Galo открывается при помощи

RFID метки и обычного ключа. В некоторых версиях имеется панель ввода пароля и дистанционный радио пульт. Электронная часть питается от внешнего источника питания и не имеет бесперебойной части. На рисунке 4 изображён электронный замок от фирмы Galo. Преимущество данного электронного замка заключается в невысокой цене. К недостаткам относится отсутствие бесперебойного источника питания, отсутствие сканера отпечатка пальца. Цена обычной версии составляет 2900 рублей.

Рисунок 4 – Электронный замок от фирмы Galo.

1.3. Преимущества проекта над аналогами

Анализ выявил основные преимущества и недостатки современных устройств доступа. Распространённым недостатком современных устройств доступа является отсутствие бесперебойного питания, при отключении электроэнергии, попасть в помещении возможно лишь при помощи обычного ключа. Но бывают ситуации, что ключ отсутствует или ключ остался в закрытом помещении и тогда придётся искать иные способы попасть в помещение. Наличие в лицевой панели управляющей платы также является недостатком. Это решение увеличивает вероятность успешного взлома замка злоумышленником, так как при снятии лицевой панели ему предоставляется доступ к управляющей плате.

Основные преимущества разрабатываемого устройства:

1) Автономность будет обеспечивать li-ion аккумулятор, который будет питать устройство не менее 10 часов эксплуатации. Встроенный контроллер заряда будет заряжать li-ion аккумулятор и поддерживать его в заряженном состоянии при основном питании.
2) Плата управления и необходимая оснастка, разрабатываемого устройства, находится на тыловой части двери. Это решение не даст свободного доступа к плате управления и остальным частям устройства, повышая тем самым защищённость.
3) Наличие четырёх способов открытия двери.
4) Питать устройство можно от любых 5В USB зарядок с выдаваемым током 2 и более ампер.

2 ТЕХНИЧЕСКАЯ ЧАСТЬ

2.1. Разработка электрической структурной схемы

Схема электрическая структурная даёт общий вид на принцип работы устройства. На электрической структурной схеме отображают основные функциональные блоки устройства, их предназначение и линии связи между ними.

Обычно проектирования устройств начинается с создания структурной схемы, это позволяет на ранних стадиях разработки определять ошибки проектирования и изменять требования к основным узлам разрабатываемого устройства. Создание структурной схемы упрощает создание электрической принципиальной схемы устройства [4].

Структурная схема даёт наглядное понятие обо всех блоках устройства. Блоки выполняются в виде:

1) Прямоугольников
2) Условно-графических обозначений
3) Упрощённых внешних очертаниях

В итоги была составлена электрическая структурная схема системы контроля доступом. Составленную структурную схему мысленно можно разделить на две части:

1) Часть электропитания
2) Управляющая часть

Часть электропитания состоит из шести блоков. Основное питание ~220В подаётся на блок “Блок питания на 5В”, после чего понижается до постоянного напряжения 5В. Полученное напряжение 5В подаётся на два блок, на блок “Контроллер заряда” и блок “Цепь переключения питания”. Блок “Контроллер заряда” отвечает за контролируемый заряд аккумулятора обозначенным блоком “Аккумулятор”. Сам же блок “Аккумулятор” подключён всё к тому же блоку “Цепь переключения питания”. Блок “Цепь переключения питания” отвечает за мгновенные переключения основного питания на питание с аккумулятора при

автономном режиме работы (при пропаже ~220В). После чего питание поступает на два блока: блок” Линейный стабилизатор напряжения 3.3В” и блок “Повышающий DC-DC преобразователь”. На двух последних блоках и формируются два основных напряжения, 3.3В и 12В, для питания управляющей части.

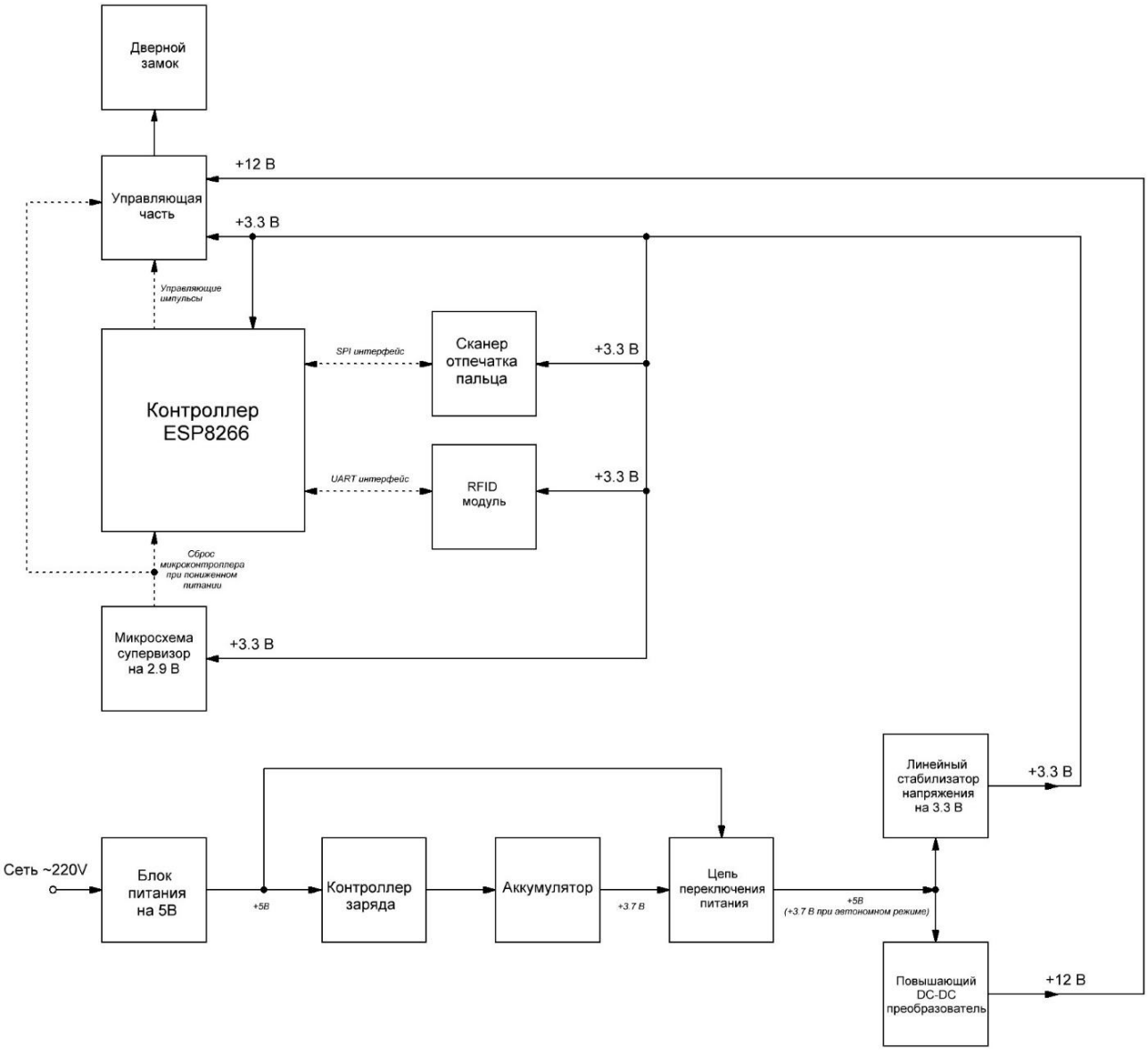


Рисунок 5 - Электрическая структурная схема системы контроля доступом.

Управляющая часть состоит из 6 блоков, главным из которых является блок “Контроллер ESP8266”. К нему подключается по SPI интерфейсу сканер

отпечатка пальца, а также, по UART интерфейсу, RFID модуль. Дабы избавить микроконтроллер от зависаний, возникший в результате скачка питания 3.3В, был добавлен блок “Микросхема супервизор 2.9В”. Для того что бы микроконтроллер мог управлять дверным замком (блок “Дверной замок”), между ними, был установлен блок “Управляющая часть”. Этот блок не только коммутирует дверной замок, но и предотвращает ложные срабатывания, путём контроля питания 3.3В и ножки сброса контроллера “RESET”. На рисунке 5 изображена электрическая структурная схема системы контроля доступом. Так же в приложении А приведена электрическая структурная схема системы контроля доступом в формате А3.

2.2 Выбор и обоснование элементной базы

Одной из основных частей системы контроля доступом является микроконтроллер. Микроконтроллер принимает данные от считывателя RFID карт и от сканера отпечатка пальца, после обрабатывает и решает давать доступ пользователю или нет. Так же контроллер должен обладать интерфейсами: UART, SPI, Wi-Fi. По UART и SPI подключается модуль RFID и модуль сканера отпечатка пальца. Wi-Fi служит для настройки и управлением системой контроля доступом беспроводным путём [5].

Для выполнения задач хорошо подойдёт микроконтроллер ESP8266 фирмы производителя Espressif. На рисунке 6 изображён микроконтроллер ESP8266.

Рисунок 6 - Микроконтроллер ESP8266.

Основные характеристики ESP8266 [6]:

1) Питание от 2.2В до 3.6В, максимальное потребление - 215мА, номинальное потребление – 70мА, поддерживаются три режима пониженного потребления.

2) Тактовая частота 80МГц, разрядность 32 бита, память 4Мб.

3) Имеется 11 портов ввода-вывода, из них: один порт АЦП и 10 цифровых.

4) Интерфейсы: UART, SPI, I^2S, Wi-Fi.

Для обеспечения доступа по RFID меткам был выбран модуль RC522. Модульная система позволяет уменьшить время разработки устройства, а также конечную стоимость устройства. RFID (радиочастотная идентификация) - это набор методов автоматической идентификации предметов, посредством записи и считывания радиосигналов в RFID-метках. Модуль RC522 работает на частоте 13,56МГц. Для подключения к микроконтроллеру используется SPI интерфейс (модуль имеет так же UART и I^2S). На рисунке 7 изображён модуль RFID RC522. На рисунке 8 изображены RFID метки.

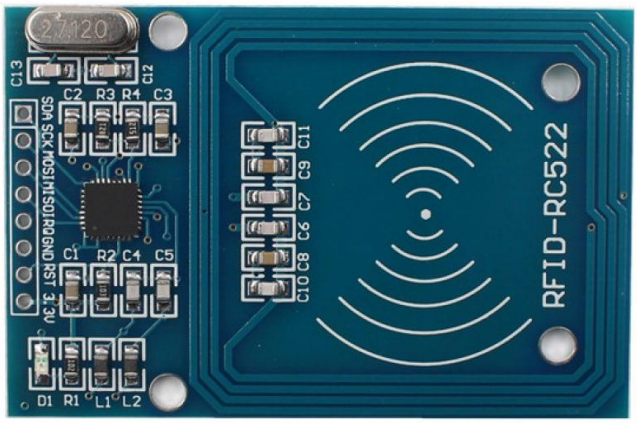


Рисунок 7 - модуль RFID RC522.

Рисунок 8 - RFID метки.

Основные характеристики RC522:

1) Питание от 2.5В до 3.6В, максимальное потребление - 50мА,
2) Рабочая частота 13.56МГц
3) Скорость передачи до 10Мбит/С
4) Дальность считывания 0-60мм
5) Интерфейсы: SPI (основной), UART, I^2C
6) Размеры 40мм х 60мм

Для идентификации пользователя был выбран сканер отпечатка пальца фирмы Diymore модель FPM10A. Сканер имеет оптический сенсор, что позволяет делать фотографии рисунка кожи на пальце и заносить в базу. В сканере также присутствует собственная память и чип. Чип обрабатывает фотографии, производит необходимые расчёты для обнаружения соответствия между записанным в базе и проверяемым отпечатком. В базу вмешается до 162 отпечатков. Для подключения к микроконтроллеру, присутствует UART интерфейс. На рисунке 9 изображён сканер отпечатка пальца FPM10A.

Рисунок 9 - Сканер отпечатка пальца FPM10A.

Основные характеристики:

1) Питание от 3В до 6В, максимальное потребление 140мА, номинальное потребление 120мА.
2) Максимальное количество сохранённых отпечатков – 168.
3) Интерфейс подключения UART, скорость передачи данных: 9600, 19200, 28800, 38400, 57600 (по умолчанию).
4) Время обработки изображения отпечатка: < 1.0 секунды.
5) Размеры: 56 x 20 x 21.5 мм.

Разрабатываемое устройство должно питаться от 5В, но микроконтроллер питается от 3.3В, для этого был выбран линейный стабилизатор напряжения ADP3338AKCZ-3.3. Его главным плюсом является малое падение напряжение на выходе. Малое падение напряжения на стабилизаторе позволит питать его, при переключении в автономный режим, напряжением 3,7В от Li-Ion аккумулятора [7]. Данный стабилизатор был выбран благодаря выходному току в 1А при достаточном компактном корпусе SOT-223. На рисунке 10 изображена микросхема ADP3338AKCZ-3.3

Основные характеристики:

1) Входное напряжение от 2.7В до 8В, выходное напряжение 3.3В
2) Максимальный выходной ток 1А
3) Падение напряжение 0.2В при 1А
4) Корпус SOT-223

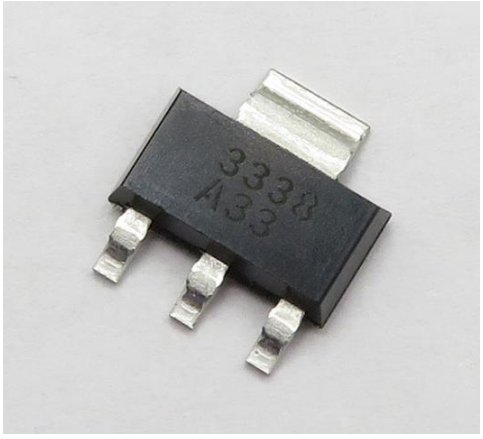


Рисунок 10 - Микросхема ADP3338AKCZ-3.3.

Помимо 3.3В в устройстве необходимо и 12В питание для открывания электромагнитного замка, для этого необходимо повысить 5В в 12В. С этой задачей прекрасно справится распространённая микросхема MC34063. Микросхема MC34063 является импульсным преобразователем напряжения и в зависимости от схемы подключения может быть, как повышающей, так и понижающей [8]. В нашем случае она будет использоваться как повышающий преобразователь напряжения. Так как электромагнитный замок управляется импульсом (для открытия подаётся импульс в 200-500мс), то микросхеме нужно лишь зарядить электролитические конденсаторы, после чего она работает на холостом ходу. Длительность заряда конденсаторов не велика, соответственно нет большой нагрузки на микросхему. На рисунке 11 изображена микросхема MC34063.

Основные характеристики:

1) Входное напряжение от 3В до 40, выходное напряжение от 1.25В до 38В
2) Максимальный выходной ток 1.5А
3) Рабочая частота до 100кГц
4) Тип преобразователя – повышающий/понижающий
5) Корпус микросхемы SO-8

Рисунок 11 - Микросхема MC34063.

Что бы устройство было автономным необходимо использовать аккумулятор. Что бы в корпус устройства поместить аккумулятор, при этом он мог выдавать ток 2-3А без просадок по напряжению, был выбран Li-ion аккумулятор фирмы Sanyo UR18650FM с номинальной ёмкостью 2500мАч [9]. На рисунке 12 изображён аккумулятор Sanyo UR18650FM.

Рисунок 12 - Аккумулятор Sanyo UR18650FM.

Основные характеристики:

1) Минимальное напряжение 2.5В, максимальное 4.2В.

2) Ёмкость 2500мАч.

3) Номинальный выходной ток 2.6А, максимальный 5.2А, номинальный ток заряда 1.6А.

4) Количество циклов заряда/разряда: более 500

5) Типоразмер 18650.

Для защиты аккумулятора от перенапряжения, глубокого разряда и коротких замыканий была применена микросхема защиты li-ion аккумуляторов DW01A фирмы Fortune Semiconductor Corporation. При превышении 4.2В, на аккумуляторе, микросхема отключает аккумуляторы от платы устройства, аналогичное действие происходит и при падении напряжения, на аккумуляторе, ниже 2.5В. На рисунке 13 изображён контроллер защиты DW01A.

Рисунок 13 - Контроллер защиты DW01A.

Микросхема DW01A служит только как защита в экстренных случаях. Для постоянного контроля напряжения и правильного заряда аккумулятора применён контроллер заряда TP4056 [10]. Это изделие с линейным зарядом по принципу постоянное напряжение/постоянный ток для одноэлементных li-ion аккумуляторов. В неё имеется встроенный термодатчик, отключающий контроллер при перегреве. Есть регулировка тока заряда аккумулятора до 1А. На рисунке 14 изображён контроллер заряда TP4056

Процесс зарядки состоит из нескольких этапов:

1) Контроль напряжения подключенного аккумулятора (постоянно);

2.) Зарядка током 1/10 от запрограммированного резистором Rprog (100мА при Rprog= 1.2к) до уровня 2.9 В (если требуется);

3) Зарядка максимальным током (1000мА при Rprog = 1.2к);

4) При достижении на батарее 4.2В применяется стабилизация напряжения на уровне 4.2В. Ток падает по мере зарядки;

5) При достижении тока 1/10 от запрограммированного резистором Rprog (100мА при Rprog = 1.2к) зарядное устройство отключается. Переход к п. 1

Основные характеристики:

1) Напряжение питания +4,5...+8,0
2) Ток заряда 1,0 Ампер (1000 мА), легко программируется изменением значения резистора Rprog (от 1,2k до 30k);

3) Напряжение окончания заряда аккумулятора: 4,2 вольта;
4) Светодиодная индикация заряда, светодиодная индикация окончания заряда;
5) Корпус SO-8

Рисунок 14 - Контроллер заряда TP4056.

По мимо электронной части, в системе контроля доступом должна быть и механическая (для открытия двери обычным ключом в экстренных случаях). Для открытия механическим и электронным способом был выбран электромеханический замок FE-2369 накладного типа [11]. Также на замке установлена кнопка блокировки. Питание замка от 9В до 12В. На рисунке 15 изображён электромеханический замок FE-2369.

Рисунок 15 - Электромеханический замок FE-2369.

Основные характеристики электромеханического замка FE-2369:

1) Напряжение питания от 9В до 12В, мощность 12Вт
2) Тип монтажа – накладной
3) Тип ключа – зубчатый ключ
4) Основной материал – сталь
5) Вес 1.58 кг

Самым эффективным и дешевым способом контроля над напряжением питания при разработке микропроцессорных систем является использование внешней микросхемы супервизора питания. Она позволяет не только поддерживать контроллер в состоянии сброса перед его пуском (функция POR — power on reset), но и контролировать уровень и стабильность питания во время выполнения программы (функция BOR — brown out reset), выполнять функции сторожевого таймера (WDT). Был выбран супервизор фирмы Maxim Semiconductor модель MAX809SEUR. При понижении напряжения питании, ниже 2.9 В, супервизор подтягивает контакт reset микроконтроллера к нулю, тем самым перезагружая его.

Основные характеристики микросхемы MAX809SEUR :

1) Пороговое напряжение 2.93 В
2) Тип сброса – active low
3) Сброс ожидание – 140мс
4) Корпус – SOT-23

2.3. Разработка электрической принципиальной схемы

Электрическая принципиальная схема – схема, показывающая весь состав компонентов и их связей, также она даёт представление о работе изделия

Имея структурную схему и выбранную элементную базу, была разработана электрическая принципиальная схема системы контроля доступом. При разработке так же изучалась документация на основные компоненты [12]. Разработанная электрическая принципиальная схема находится в приложении Б в формате А3.

2.4. Расчет системы контроля доступа

Что бы управлять замком необходим каскад из двух биполярных транзисторах. Это необходимо из-за того, что вывод микроконтроллера ESP8266 имеет слабый выходной ток 12мА, а коммутируемый ток составляет 3А. Схема каскада управления замка изображена на рисунке 16.

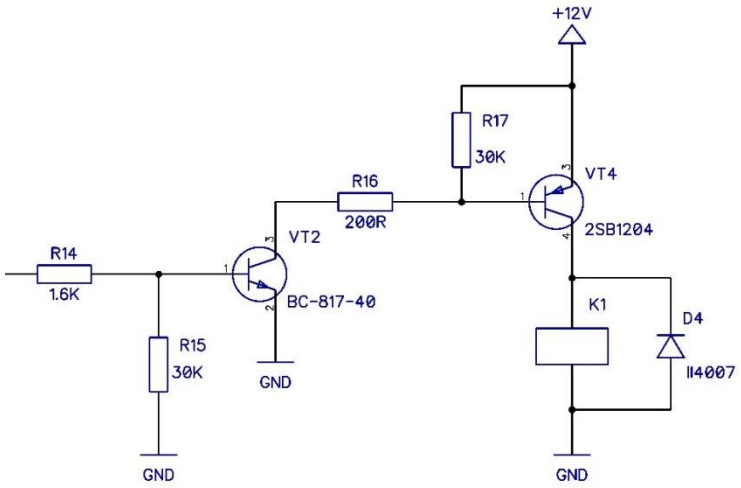


Рисунок 16 - Каскад управлением замком.

Для начала нам необходимо выбрать силовой транзистор VT4, после будет выбираться транзистор VT2. Из документации к замку нам известно, что при 12В питания ток нагрузки составляет 3А. Соответственно у выбранного транзистора должен быть запас по току, для долговечной работы и лучшего рассеивания тепла. В результате подбора силового транзистора, из справочных материалов, был выбран транзистор 2SB1204 в корпусе TO252. Основные характеристики:

1) Структура PNP
2) Максимально допустимый ток коллектора– 8А
3) Максимальное напряжение к-э – 50В
4) Коэффициент передачи тока – 140
5) Граничная частота коэффициента передачи тока – 130МГц

Рассчитаем сопротивление резистора R16, влияющий на работу силового транзистора VT4 в ключевом режиме. Вначале рассчитаем ток базы $I_б$, при помощи формулы 1.

$$I_б = \frac{I_к}{h_{21e}} = \frac{8}{140} = 0{,}057\text{А}. \quad (1)$$

где $I_б$ – ток базы транзистора, Ампер;

$I_к$ – максимальный ток коллектора, Ампер;

h_{21e} – коэффициент усиления.

Необходимо произвести расчёт напряжения на базе транзистора $U_б$, при помощи формулы 2.

$$U_б = U_п - U_{пбэ} = 12 - 0{,}6 = 11{,}4 \text{ В}. \quad (2)$$

Где $U_{пбэ}$ –падение напряжения на переходе б-э, В;

$U_б$ – напряжение базы транзистора, В;

$U_п$ – напряжение питания, В;

Согласно закону Ома, рассчитывает резистор базы $R_б$ (R16), по формуле (3).

$$R_б = \frac{U_б}{I_б} = \frac{11{,}4}{0{,}057} = 200 \text{ Ом}. \quad (3)$$

где $R_б$ – сопротивление резистора базы, Ом;

Согласно расчёту, выбираем резистор из стандартного ряда E24 равному 200 Ом.

Рассчитываем мощность рассеивания на резисторе R16, по формуле 4.

$$P_{рас} = I_б^2 \cdot R_б = 0{,}057^2 \cdot 200 = 0{,}649 \text{ Вт}. \quad (4)$$

где $P_{рас}$ – рассеиваемая мощность резистора, Вт;

$I_б$ – ток базы транзистора, А;

$R_б$ – токоограничивающий резистор, Ом.

В результате расчётов мощности подойдёт резистор типоразмера 2010 имеющий мощность рассеивания 0.75Вт.

Для управления силовым транзистором VT4, по схеме, используется транзистор VT2. Мы знаем протекающий ток базы VT4, который составляет 0.057А, соответственно можем подобрать нужный нам транзистор. В результате

подбора транзистора VT2, был выбран транзистор BC-817-40. Основные характеристика транзистора BC-817-40:

1) Структура NPN
2) Максимально допустимый ток коллектора – 0.5А
3) Максимальное напряжение к-э – 45В
4) Коэффициент передачи тока – 300
5) Граничная частота коэффициента передачи тока – 100МГц

Рассчитаем резистор R14. Для начала рассчитываем ток базы $I_{б,}$ при помощи формулы 5.

$$I_б = \frac{I_к}{h_{21e}} = \frac{0,5}{300} = 0,0016А. \quad (5)$$

Рассчитываем напряжение на базе транзистора VT2, по формуле (6).

$$U_б = U_п - U_{пбэ} = 3.3 - 0,7 = 2,6 В. \quad (6)$$

Согласно закону Ома, рассчитывает резистор базы $R_б$ (R14) транзистора VT2, по формуле (7).

$$R_б = \frac{U_б}{I_б} = \frac{2.6}{0,0016} = 1625 Ом. \quad (7)$$

Согласно расчёту, выбираем резистор из стандартного ряда E24 равному 1.6 кОм. Рассчитываем мощность рассеивания на резисторе R14, по формуле 8.

$$P_{рас} = I_б^2 \cdot R_б = 0,0016^2 \cdot 1600 = 0,004 Вт. \quad (8)$$

В результате расчётов мощности подойдёт резистор типоразмера 0805 имеющий мощность рассеивания 0.125Вт. Расчёт каскада управлением замка закончен.

Заряд li-ion аккумулятора должен осуществляться, стабилизированным током и напряжением. Микросхема TP4056 контролирует уровень напряжения на аккумуляторе во время заряда и так же ограничивает ток заряда. Номинальный зарядный ток может быть изменен подбором резистора Rprog из таблицы 1. Аккумулятор Sanyo UR18650FM рассчитан на ток заряда 1C, но максимальная долговечность аккумулятора достигается уменьшением током заряда до 0.1C-0.3C. Так как большой необходимости в быстром заряде

аккумулятора нет, то вполне подойдёт ток заряда 0.2С. Рассчитаем ток заряда аккумулятора Iзар по формуле 9.

$$I_{зар} = 0.2 \cdot C = 0.2 \cdot 2.5 = 0.5A. \quad (9)$$

Где С – ёмкость аккумулятора, А/час.

Выбираем резистор Rprog, из таблицы 1, из расчёта тока заряда 0.5А. Было выбрано среднее значение, между 2кОм и 3кОм, а именно 2.5кОм.

Таблица 1 – Таблица подбора сопротивления Rprog.

Резистор (кОм)	Ток заряда (мА0
30	50
20	70
10	130
5	250
4	300
3	400
2	580
1.66	690
1.5	780
1.33	900
1.2	1000

Импульсный повышающий преобразователь напряжения (ИППН) должен выдавать на выходе стабилизированное напряжение 12В. Выходное напряжение устанавливается путём подбора двух резисторов R21 и R22 изображённых на рисунке 17. Подбор номиналов резисторов осуществляется из расчета, что на входе компаратора должно быть напряжение равное 1,25В. Резистор R22 выбирается любого номинала из 1-100кОм, по ряду Е24. Был выбран резистор R22 номиналом 13 кОм. Рассчитываем резистор R15 по формуле 10.

$$R1 = \frac{R_2}{\frac{U_{вых}}{1.25} - 1} = \frac{13000}{\frac{12}{1.25} - 1} = 1511 \text{ Ом.} \quad (10)$$

Где Uвых – заданное выходное напряжение, В;

R1 – Сопротивление резистора R21, Ом;

R2 – Сопротивление резистора R22, Ом.

Резистор R15 подбираем из ряда E24. Был выбран ближайший по значению резистор номиналом R21 = 1500 Ом.

Выполняем проверку выходного напряжения по формуле (11).

$$U_{вых} = 1.25 \left(1 + \frac{R_2}{R_1}\right) = 1.25 \left(1 + \frac{13000}{1500}\right) = 12.08\text{В} \quad (11)$$

Где Uвых – заданное выходное напряжение, В;

R1 – Сопротивление резистора R21, Ом;

R2 – Сопротивление резистора R22, Ом.

Напряжение на выходе импульсного преобразователя составляет 12.08В при R21 = 1.5 кОм, R22 = 13 кОм.

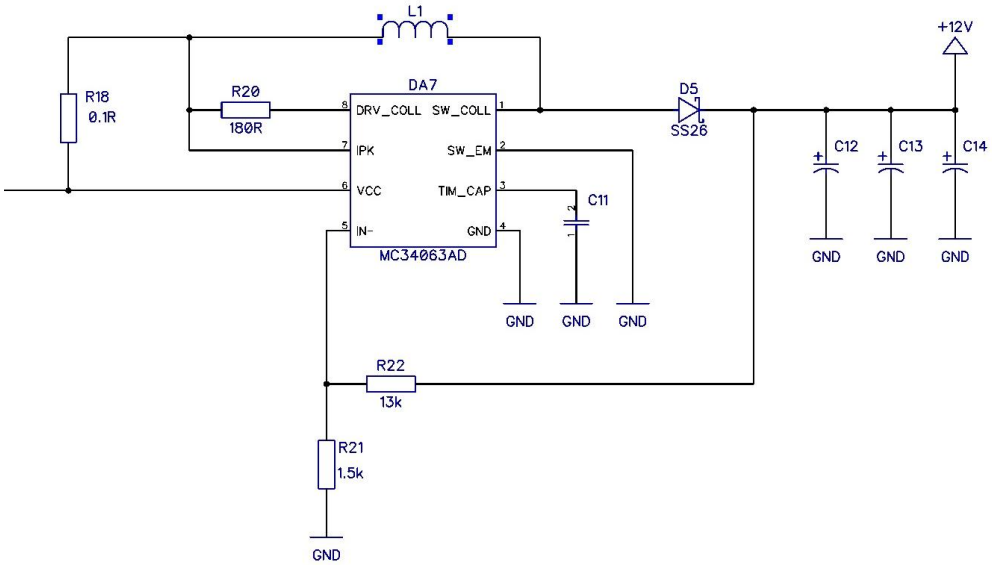


Рисунок 17 - Импульсный преобразователь напряжения.

Рассчитываем шунт R18 цепи ограничения тока ИППН. Для быстрой зарядки конденсаторов C12, C13, C14 выбираем максимальный номинальный ток ИППН из документации он составляет $I_{мак} = 1.5\text{А}$. Рассчитываем сопротивление шунта по формуле (12)

$$R_{\text{шунт}} = \frac{U_{\text{оп}}}{I_{\text{мак}}} = \frac{0.3}{1.5} = 0.2 \text{ Ом}. \quad (12)$$

Где :$U_{\text{оп}}$ – опорное напряжение цепи ограничения тока, равное $U_{\text{оп}}$ = 0.3В.

$I_{\text{мак}}$ – максимальный ток, А.

Рассчитываем мощность рассеивания на шунте R18, по формуле 13.

$$P_{\text{рас}} = I_{\text{мак}}^2 \cdot R_{\text{шунт}} = 1.5^2 \cdot 0.2 = 0.45 \text{ Вт}. \quad (13)$$

Где: $P_{\text{рас}}$ мощность рассеивания на шунте.

Согласно расчёту, выбираем шунт номиналом равному 0.2 Ом.

В результате расчётов мощности подойдёт шунт типоразмера 2010 имеющий мощность рассеивания 0.75Вт.

Что бы работал ИППН необходим диод шоттки D5. Диод подбирается исходя из максимального тока, а также выходного напряжения. Для увеличения надёжности параметры берутся с 30% запасом. Был выбран диод шоттки SS26, с максимальным прямым током 2А, максимальное обратное напряжение 60В.

В документации на микросхему MC34063, работающей в повышающим режиме, указана рекомендованная индуктивность дросселя L1 номиналом 170мкГн.

Так же нужно выполнить расчёт четырёх светодиодов: HL1, HL2, HL3, HL4. По справочным материалам [13] была подобрана модель светодиода зелёного цвета FYL-5002UBC1F. Основные характеристики FYL-5002UBC1F:

1) Номинальный ток 20мА
2) Рабочее напряжение 2.1-2.3В
3) Потребляемая мощность 0.5 мВт
4) Сила светового потока, 45-50Лм

Рассчитаем токоограничивающий резистор для светодиода HL1 по формуле 14.

$$R_{\text{то}} = \frac{(V_{\text{п}} - V_{\text{пн}})}{I_{\text{св}}} = \frac{(5-2.1)}{0{,}02} = 145 \text{ Ом}. \quad (14)$$

Где $R_{\text{то}}$ – токоограничивающий резистор, Ом;

$V_{\text{п}}$ – питающее напряжение, В;

$V_{\text{пн}}$ – падение напряжения светодиода, В;

$I_{\text{св}}$ – ток светодиода, А.

Выбираем из ряда E24 ближайший по значению, а именно $R_к$ = 150 Ом. Найдём мощность рассеивания токоограничивающего резистора для светодиода HL1 по формуле 15.

$$P_{рас} = I_{рсв}^2 \cdot R_{то} = 0.02^2 \cdot 150 = 0.06 \text{ Вт}. \quad (15)$$

Где $R_{то}$ - токоограничивающий резистор, Ом;

I_{pcb} – номинальный ток светодиода, А;

$P_{рас}$ – рассеиваемая мощность, Вт.

В результате расчётов мощности подойдёт резистор типоразмера 0805 имеющий мощность рассеивания 0.125Вт. Так как светодиоды HL1, HL2, HL3 подключены к одному питанию, то расчёт одного токоограничивающего резистора можно применить и к оставшимся двум.

Рассчитаем токоограничивающий резистор для HL4 по формуле 16.

$$R_к = \frac{(V_п - V_{пн})}{I_{св}} = \frac{(3.3-2.1)}{0,02} = 60 \text{ Ом} \quad (16)$$

Выбираем из ряда E24 ближайший по значению и получаем Rк = 56 Ом.

Найдём мощность рассеивания токоограничивающего резистора для HL4 по формуле 17.

$$P_{рас} = 0.02^2 \cdot 56 = 0.022 \text{ Вт} \quad (17)$$

После расчётов мощности подойдёт резистор типоразмера 0805 имеющий мощность рассеивания 0.125Вт. По итогу расчётов токоограничивающих резисторов были выбраны резисторы:

1) Для HL1, HL2, HL3 – 0805 150 Ом.
2) Для HL4 – 0805 60 Ом.

2.5 Расчёт надёжности системы контроля доступом

На этапе разработки устройства производят расчёт надёжности для нахождения времени наработки на отказ. Надёжность рассчитывается по известным данным об интенсивности отказов элементов, к числу основных показателей надёжности относятся [14]:

1) Долговечность
2) Сохраняемости
3) Безотказность
4) Ремонтопригодность

Безотказность – это свойство изделия (машины, агрегата и т.п.) выполнять заданные функции, сохраняя свои эксплуатационные показатели в заданных пределах в течение требуемого промежутка времени или требуемой наработки в конкретных условиях и режимах эксплуатации этого изделия. Показателями безопасности является:

1) Наработка на отказ
2) Средняя наработка (до появления неисправностей)
3) Вероятность безотказной работы
4) Интенсивность отказов

В справочных данных берутся показатели интенсивности отказов компонентов. В таблице 2 показана интенсивность отказов компонентов.

Интенсивность отказов каждой группы рассчитаем согласно формуле:

$$\lambda_{\Sigma} = \mathrm{Ni} \cdot \lambda_{0\mathrm{i}} \cdot 10^{-6}, \quad (18)$$

где λ_{Σ} – интенсивность отказов группы;

N_i – количество элементов;

λ_{0i} – интенсивность отказов.

Суммарную интенсивность отказов модуля рассчитываем по следующей формуле:

$$\lambda_{\Sigma} = \quad \mathrm{N_i} \cdot \lambda_{0\mathrm{i}} \cdot 10^{-6} \ . \quad (19)$$

Повышающий преобразователь напряжения, микроконтроллер, стабилизатор, выполнены в виде интегральных микросхем.

Испытания блока индикации проводятся в лабораторных условиях.

Таблица 2 - Интенсивность отказов компонентов системы контроля доступом.

Наименование элемента	Обозначение по схеме	Кол-во элементов N_i, шт.	Интенсивность отказов, $\lambda_{0i} \cdot 10^{-6}$	Интенсивность отказов группы, $\lambda_{\Sigma} \cdot \cdot 10^{-6}$
Конденсаторы неполярные	C	4	0,1	0,4
Конденсаторы полярные	C	8	0,1	0,8
Микросхемы	DA	7	0,3	2,1
Микроконтроллер	DD	1	0,1	0,1
Светодиоды	HL	4	1,1	4,4
Дроссель	L	1	0,05	0,05
Резисторы постоянные	R	20	0,3	6
Биполярный транзистор	Q	2	0,1	0,2
Полевой транзистор	Q	4	0,1	0,4
Диод	VD	5	0,5	2,5
Переключатели	S	3	1,4	4,2
Сканер отпечатка пальца	M	1	1,9	1,9
Модуль RFID	BL	1	1,9	1,9
Предохранитель	F	1	0,05	0,05
Аккумулятор	GB	1	5	5
Разъёмы	X	5	1,0	5
Паяные соединения	-	239	0,01	2,39
				λ_{Σ} = 37,39

Рассчитаем наработку на отказ, $T_{0,}$ по формуле 19:

$$T_0 = \frac{1}{\lambda_\Sigma} \frac{1}{37{,}39 \cdot 10^{-6}} \approx 26 \text{ тыс. часов.} \quad (19)$$

Вероятность безотказной работы, P(t), при времени работы равной t = 1000 часов составит (расчёт осуществляется по формуле 20):

$$P(t) = e^{-\lambda_\Sigma \cdot t}\, e^{-37{,}39 \cdot 10^{-6} \cdot 1000} = 0{,}96.$$

(20)

Примерный срок службы прибора составит около 7 лет, при ежедневной работе по 10 часов в сутки. При работе 24 часа в день срок службы составит 3 года.

2.6. Расчёт стоимости печатного узла

Себестоимость, это один из основных показателей выполненной работы. Себестоимость показывает успехи и недостатки в работе. Это все затраты на изготовление и реализацию товара выраженные в денежном виде. По способу отнесения на себестоимость затраты делятся на прямые и косвенные. Затраты связанные с выпуском изделия называются прямыми. Затраты связанные с работой всего предприятия называется косвенными.

Расчет стоимости материалов складывается из стоимости основных, вспомогательных материалов и стоимости покупных полуфабрикатов. Стоимость основных, вспомогательных материалов рассчитывается исходя из норм расхода, цена за единицу. Стоимость покупных полуфабрикатов, то есть, стоимость электрорадиоэлементов рассчитывается исходя из количества радиоэлементов каждого наименования и цены. Количество и перечень радиоэлементов определяется по принципиальной электрической схеме, которая прилагается к ВКР. Расчет стоимости материалов сводится в таблицу 3:

Таблица 3 - Расчет стоимости покупных полуфабрикатов.

Тип, марка	ГОСТ	Количество (шт)	Цена за шт. (руб.)	Сумма (руб.)
Конденсатор К50-35	ожо.464.214 ту	3	14	42
Конденсатор К50-35	ожо.464.214 ту	2	10	20
Конденсатор К50-35	ожо.464.214 ту	2	5	10
Конденсатор К50-35	ожо.464.214 ту	1	2	2
Конденсатор К10-17В	ожо.460.107 ту	2	3	6
Конденсатор К10-17В	ожо.460.107 ту	2	3	6
Диод Шоттки SS26		3	9	18
Диод ll4007		1	5	5
Защитный диод SMBJ18		1	13	13

Микросхема TP4056		1	7	7
Микросхема DW01A		1	5	5
Микросхема ADM809			33	33
Микросхема ADP333AKCZ-3.3		1	150	25
Микросхема MC34063AD		1	30	30
Микросхема MC74AC08D		1	34	34
Микросхема MAX809		1	30	30
Микроконтроллер ESP8266		1	200	200
Светодиод 3mm		4	4	16
Предохранитель керамический 2A		1	8	8
Дроссель 220мкГн		1	30	30
Чип резисторы 0805		20	0.8	16
Кнопка тактовая		2	5	10
Кнопочный переключатель		1	11	11
Транзистор 2SB1204		1	21	21
Транзистор BC-817-40		1	3	3
Транзистор IRF7342			23	23
Транзистор FS8205A		1	14	14
Гнездо питания		1	16	16
Гнездо PBS-16		1	7	7
Вилка PLS-16		1	9	9
Модуль RFID		1	170	170
Сканер отпечатка		1	700	700

пальца				
Аккумулятор Li-ion 3.7В 1500мА/ч		1	350	350
Электромеханический замок		1	1300	1300
Стеклотекстолит односторонний		1	140	140
Корпус		1	60	
Итого:				3390

Себестоимость системы контроля доступом составила 3390 рублей.

2.7. Разработка печатной платы

Для разработки печатных плат [15], в настоящее время, активно используют системы автоматического проектирования – САПР. Одними из распространённых программ являются: Sprint-Layout, Splan, DipTrace.

Для создания печатной платы системы контроля доступом использовался пакет программ DipTrace. У программы имеется бесплатная версия, которая практически не урезает функционал. Также включает в себя несколько подпрограмм, которые полностью удовлетворяют все потребности на этапах разработки печатной платы. В набор входит:

1) Shematic Capture - программа для построения электронных схем.
2) PCB Layout – программа трассировщик
3) Component Editor – редактор компонентов

Ранее при создании принципиальной схемы была использована подпрограмма Shematic Capture. В ней есть собственная библиотека компонентов с привязкой к корпусу, поэтому разрабатывая принципиальную схему заранее можно выбрать компоненту необходимый корпус. На рисунке 18 приведена программа с разработанной электрической принципиальной схемой.

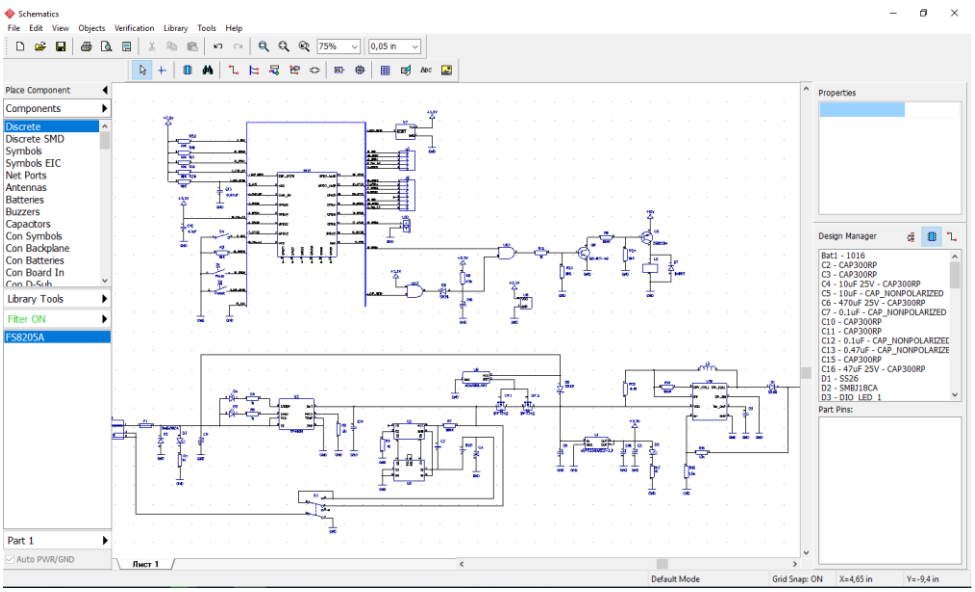


Рисунок 18 - Программа DipTrace Shematic Capture во время разработки электрической принципиальной схемы.

Далее после завершения разработки электрической принципиальной схемы, выполняется этап преобразования в плату. Для этого в программе есть команда “преобразовать в плату”, после нажатия автоматически запустится подпрограмма PCB Layout. И все компоненты, с принципиальной схемы, автоматически перенесутся в PCB Layout вместе со связями контактов. На рисунке 19 изображена программа DipTrace PCB Layout после выполнения команды “преобразовать в плату” в DipTrace Shematic.

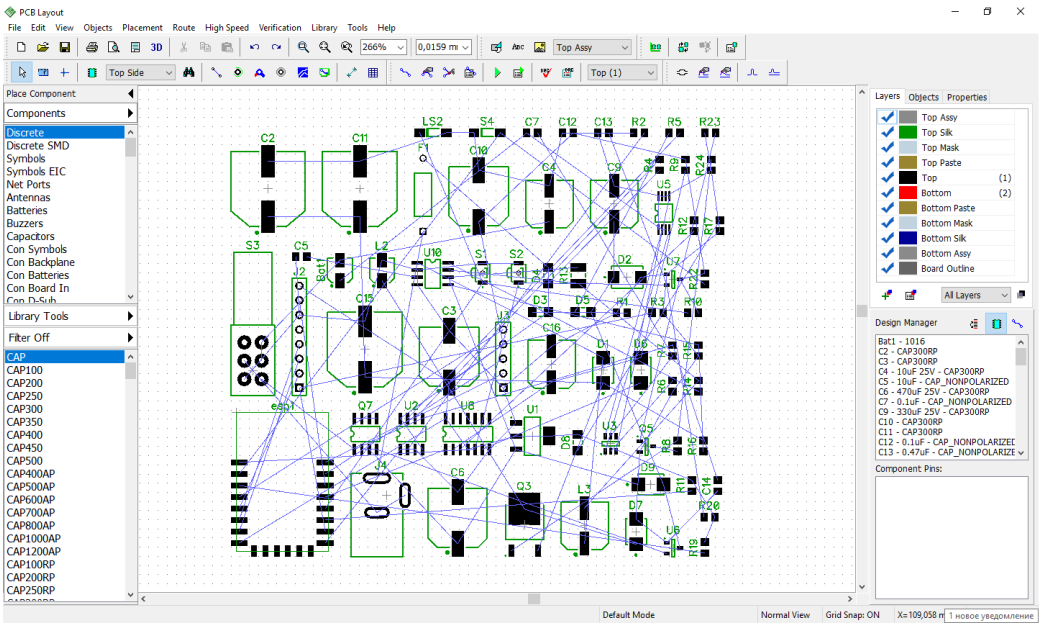


Рисунок 19 - Программа DipTrace PCB Layout после выполнения команды “преобразовать в плату” в DipTrace Shematic.

Следующим шагом было эффективное расположение компонентов на плате, дабы уменьшить число перемычек и общую длину проводников. Так же был выбран поверхностный монтажа компонентов SMD (Surface Mount Device). Это решение позволило упростить создание печатной платы и разместить все компоненты на одной стороне платы.

Далее необходимо трассировать плату, то есть провести токопроводящие дорожки между компонентами [16]. Существует 3 способа трассировки:

1) Автоматическая трассировка – это трассировка платы, осуществляющаяся при помощи алгоритмов программ САПР, без участия оператора.

2) Ручная трассировка – трассировка платы осуществляется оператором в ручную, без помощи программ САПР.

3) Комбинированный метод – трассировка платы осуществляется как при помощи оператора, так и программы САПР. Метод подразумевает автоматическую трассировку дорожек, автоматически выбирается ширина шины [17] и настройки прочих параметров, но окончательный вид печатная плата принимает только после проверки и доработки оператором. Так же бывает, что САПР не в состоянии проложить все требуемые дорожки, то выполняется ручная прокладка отдельных связей.

В итоге, для уменьшения габаритов и подгонки платы под готовый корпус, был выбран ручной метод трассировки платы. В конечном счете, мы получили трассированную печатную плату, изображённую на рисунке 20. На рисунке 21 изображена готовая плата

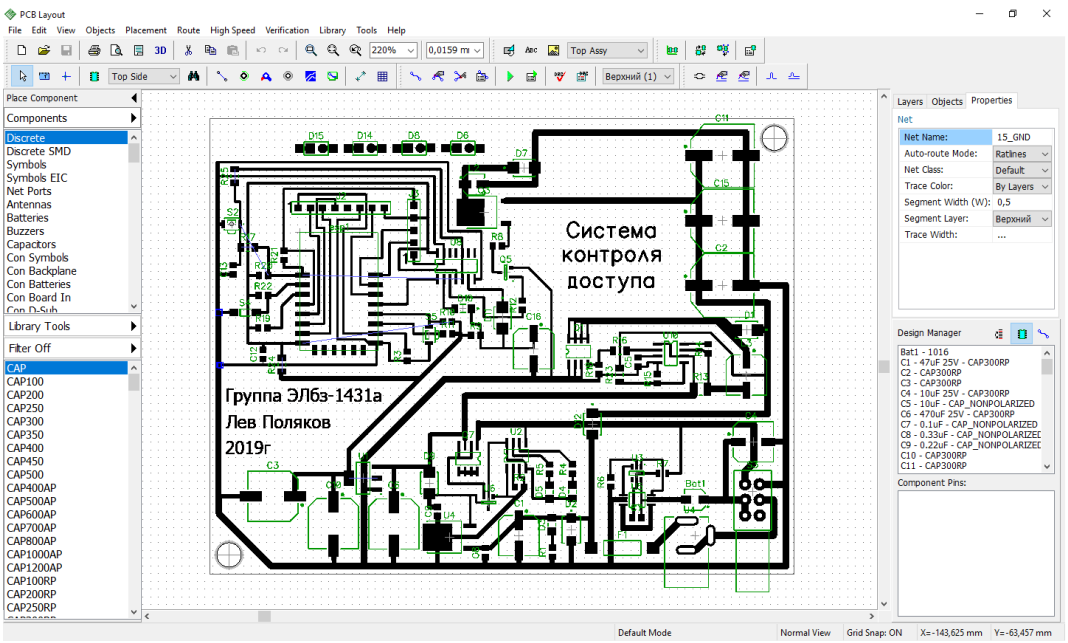


Рисунок 20 - Вид трассированной платы в программе DipTrace PCB Layout.

Программа DipTrace PCB также обладает полезной функцией выведения 3D изображения готовой платы. С помощью этой функцией можно увидеть как будет выглядеть уже изготовленная плата и позволяет вносить поправки платы во время разработки. На рисунке 21 изображена программа DipTrace PCB с 3D отображением платы. Для 3D отображения платы так же можно установить 3D корпуса компонентов, это позволит увидеть как, будет выглядеть собранное устройство.

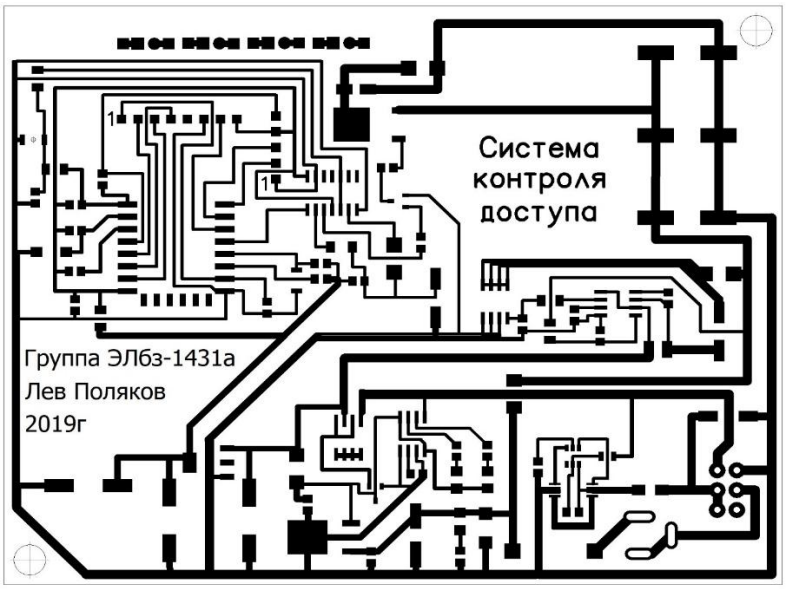


Рисунок 20 - Окончательный вид печатной платы системы контроля доступом.

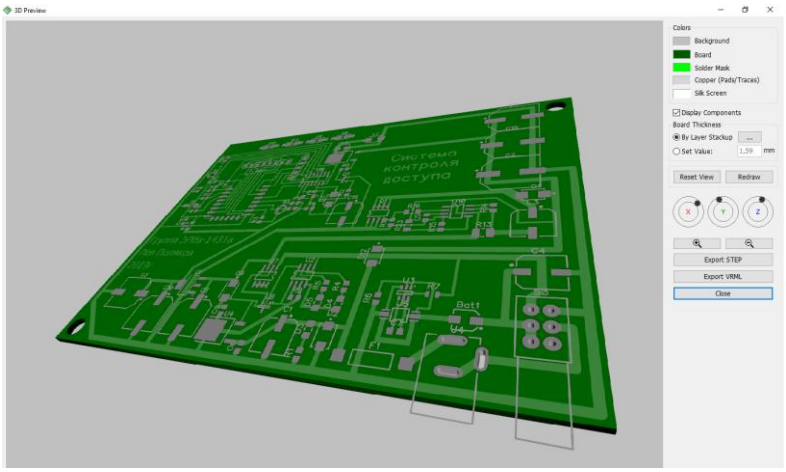


Рисунок 21 - Программа DipTrace PCB с 3D отображением печатной платы.

2.8 Разработка программного обеспечения

2.8.1. Анализ и разработка алгоритма работы ПО

Любому микроконтроллеру необходима программа действий, а именно программное обеспечение (ПО). Для написания программного кода была использована среда разработки Arduino ID. На сегодняшний день Arduino ID это одна из популярных сред разработки использующий язык программирования C++ (с некими особенностями, упрощающее новичкам написания первых программ). Распространяется бесплатно, разрабатывается как программное обеспечение с открытым исходным кодом.

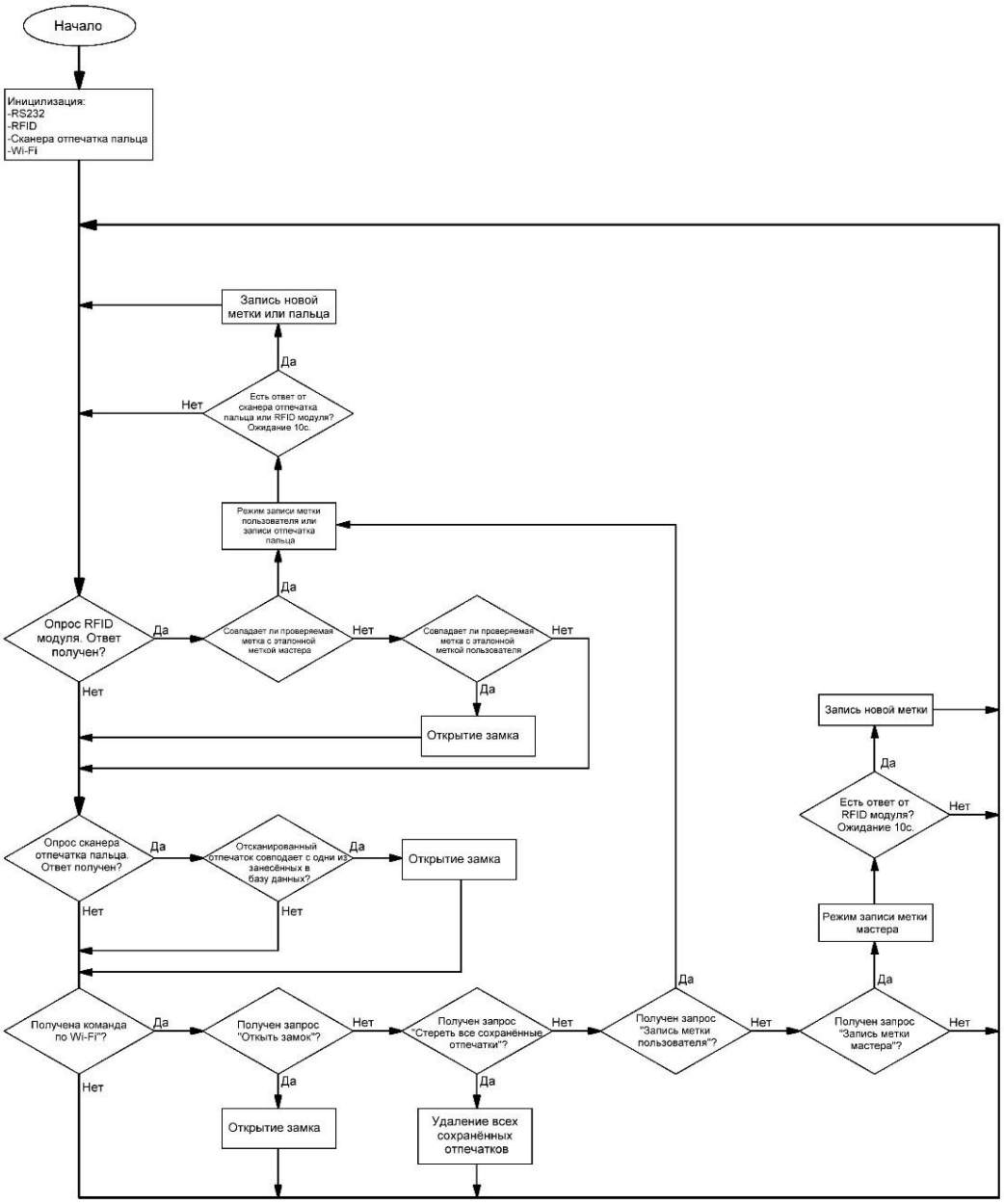


Рисунок 22 – Блок-схема работы программного обеспечения.

Что бы упростить и ускорить написание программного кода, была составлена схема алгоритма работы программы. На рисунке 22 изображена блок-схема работы программного обеспечения.

2.8.2. Написание программного кода

Написание программного кода осуществлялось на высокоуровневом языке программирования C++ [18], в среде разработки Arduino ID. На рисунке 23 изображён процесс написания программы. Для удобства, программа была разбита на несколько вкладок:

1) Security – подключение библиотек.
2) A_SETTING – установка констант.
3) A_STRUCT – структура программы.
4) B_OBJECT
5) Device – основной корень программы.
6) IMPRINT – часть отвечающая за сканер отпечатка пальца.
7) Lock – настройка времени открывания замка.
8) RFID – часть отвечающая за RFID модуль.
9) WiFi – часть отвечающая за Wi-Fi.

Весь готовый программный код, для системы контроля доступа, представлен в приложении В.

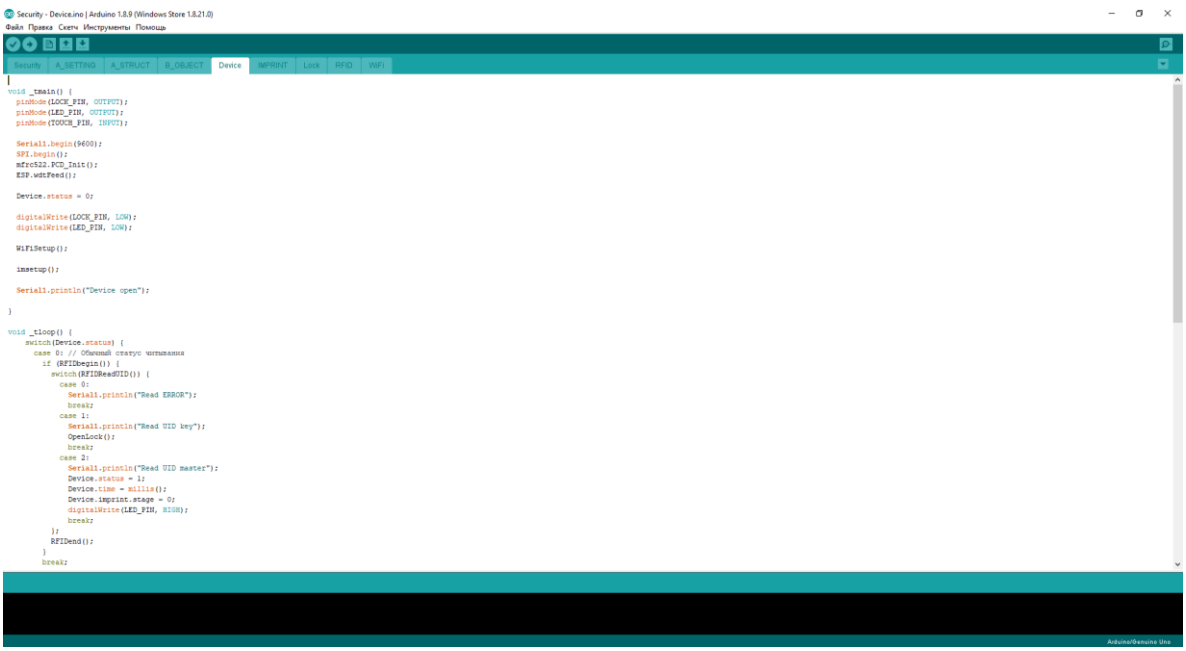


Рисунок 23 - Процесс написания программы.

3.ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

3.1. Сборка устройства

Сборка устройства начинается с изготовления печатной платы (ПП). Плата была изготовлена при помощи лазерно-утюжной технологией (ЛУТ). В ЛУТ используют глянцевую бумагу с нанесённой, при помощи лазерного принтера, рисунком платы см. рисунок 24.

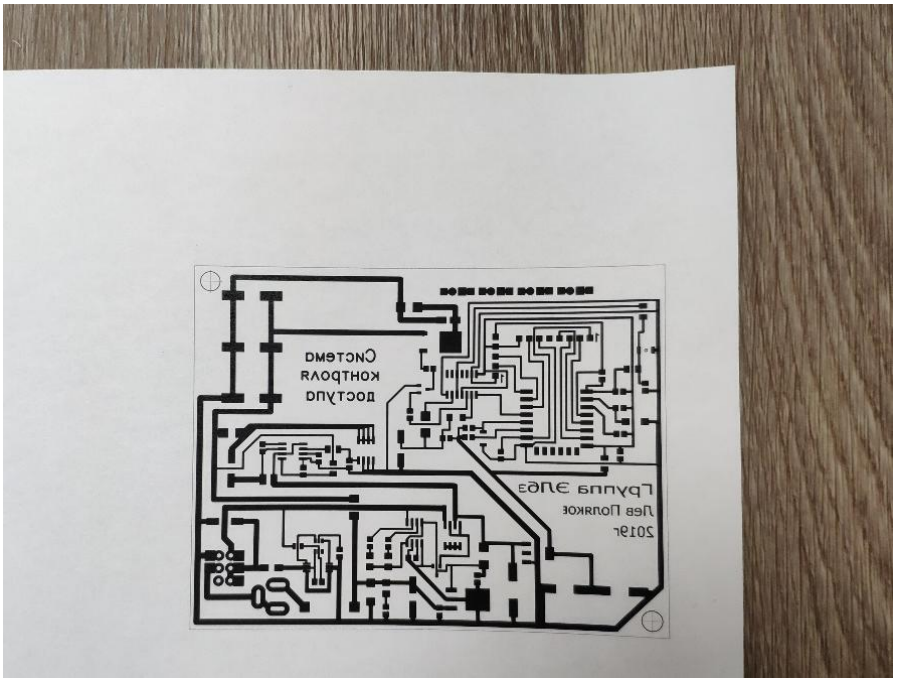


Рисунок 24 - Лист глянцевой бумаги с нанесённым рисунком ПП.

Распечатанный рисунок прикладывают к фольгированному текстолиту и после проглаживают нагретым утюгом, переводя рисунок с бумаги на текстолит. Далее отмывают водой остатки бумаги и на плате остаются защищённые участки дорожек от раствора для травления (см. рисунок 25).

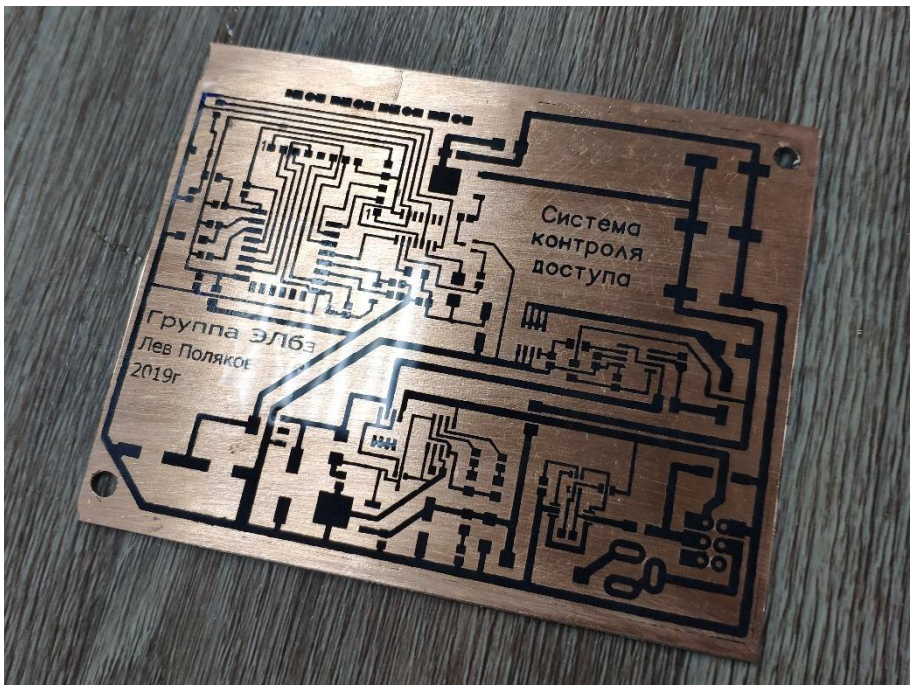


Рисунок 25 - ПП с перенесённым рисунком.

После плату помещают в раствор для травления, в результате не защищённые участки платы вытравливаются. В итоге получается плата (см. рисунок 26) с разведёнными дорожками готовая к следующему этапу.

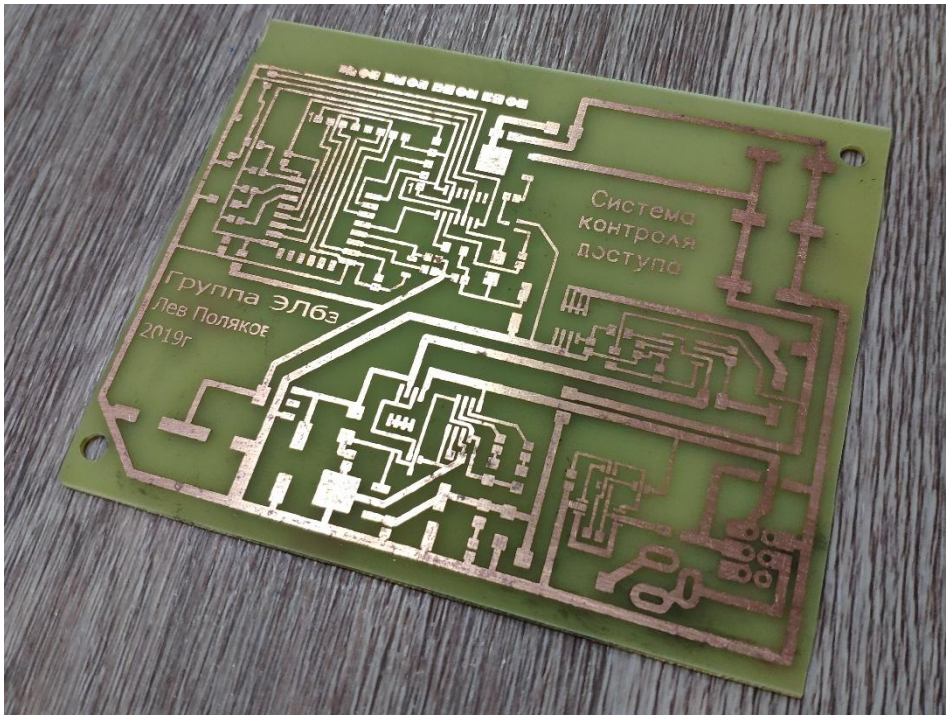


Рисунок 26 - Вытравленная плата.

Далее идёт этап монтажа электронных компонентов. На этом этапе лудят все дорожки, дабы в будущем. избежать коррозии. И устанавливают электронные компоненты. На рисунке 27 изображена плата с монтированными электронными компонентами.

Рисунок 27 - Плата с монтированными электронными компонентами.

3.2. Запись программного обеспечения на микроконтроллер

Написанное программное обеспечение необходимо записать в память микроконтроллера ESP8266. Запись в микроконтроллер ESP8266 производится при помощи UART интерфейса. Что бы подключить микроконтроллер к персональному компьютеру (ПК) используется программатор модели CH340G [19]. Программатор CH340G является двухсторонний преобразователь USB-UART. Позволяет программировать устройства с питанием 3.3В или 5В. Имеет самовосстанавливающий предохранитель на 0.5А. На рисунке 28 изображена схема подключения программатора к устройству контроля доступом

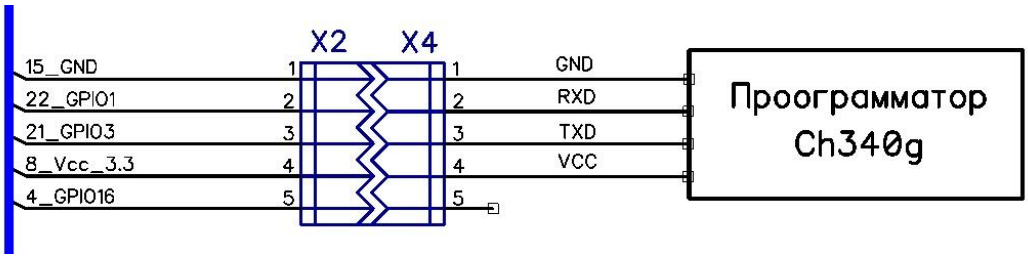


Рисунок 28 - Схема подключения программатора к устройству контроля доступом.

Для настройки программы Arduino нам нужно:

1) Заходим в настройки и в поле “Доп. ссылки для Менеджера плат” вводим “ http://arduino.esp8266.com/stable/package_esp8266com_index.json ”.
2) Заходим в “Менеджер плат” и устанавливаем “esp8266 by ESP8266 Community version 2.0.0”
3) Переходим в “Платы” и выбираем “Generic ESP8266 Module”

Для программирования разрабатываемого устройства, необходимо подключить программатор согласно схеме (см. рисунок 28). Необходимо перевести микроконтроллер в режим программирования. Зажимается кнопка “FLASH”, далее зажимается кнопка “Reset”, после отпускается кнопка “Reset” и кнопка “FLASH”. Следующим шагом загружается ПО, нажатием, в программе Arduino, на кнопку “Загрузить”. Программное обеспечение загружено на микроконтроллер.

4. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ

4.1. Наладка готового устройства

После сборки и загрузки программного обеспечения, устройство должно быть налажено на стабильную работу. Устройство, на стадии проектирования, разрабатывалось так, что бы минимизировать количество наладочных операций, но всех операций убрать не удалось.

Для начальной наладки необходимо отрегулировать длительности импульса открытия замка. Регулировка осуществляется в программной части устройства. Стандартное значение импульса открытия замка составляет 500мс. Отладкой данного параметра не является обязательным, а лишь помогает увеличить срок службы устройства и повысить время нахождения в автономном режиме. Для регулировки длительности импульса необходимо постепенно убавлять задержку (функцию delay) по 50мс, до появления неисправности в виде не открытия замка. После проявления этой неисправности вернуть значение задержки до стабильного открытия замка. Расположением “delay” является функция “void OpenLock” которая находится во вкладке Lock.

Под наладку попадает и электромеханический замок FE-2369. После монтажа электромеханического замка, зависимости от веса и силы запирания двери, дверь может либо плохо закрываться или самопроизвольно захлопываться. Для этого нужно отрегулировать пружину натяжного ригеля. Сняв крышку замка отрегулируйте с помощью втулки с шестигранной гайкой необходимое усилие взводной пружины замка.

4.2. Испытание готового устройства

Для выявления скрытых неисправностей необходимо тестировать все узлы разработанного устройства при повышенной эксплуатации. Питаем устройство от блока питания 5В и включаем кнопку питания, после чего должны засветится два светодиода PowerLed 1 и PowerLed 2, а также должна появится индикация на сенсоре отпечатка пальца. PowerLed 1 показывает наличие входного напряжения 5V, PowerLed 2 показывает наличие внутреннего напряжения 3.3V. На рисунке 29 изображена готовая система контроля доступом.

Рисунок 29 - Готовая система контроля доступом.

Далее проверяем запись отпечатков и меток. Прикладываем метку “Мастер” и записываем палец (необходимо прикладывать палец 3 раза к сканеру). Прикладываем ещё раз метку “Мастер”, после чего можно приложить пустую метку и на неё запишется электронный ключ (после этого она станет меткой “пользователь”).

Проверяем срабатывание замка, для этого подносим метку “пользователь” к передней панели, после чего замок должен открыться. Прикладываем не записанный палец к сканеру, соответственно замок должен остаться закрытым. Прикладываем записанный палец к сканеру, в случае успешного распознавания замок откроется, если нет пробуем приложить палец под другим углом.

Необходимо проверить автономный режим, отключаем внешний разъём питания 5В. Светодиод PowerLed 1 должен потухнуть, а светодиод PowerLed 2 остаться в рабочем состоянии. В таком режиме проверяем срабатывание замка и работоспособность остальных узлов. Оставляем устройство в работе в течении 10 часов, и каждые 10-30 минут делаем по несколько открытий замка. Проверяем работоспособность узлов после 10 часов тестирования. Подключаем внешнее питание 5В, появляется индикация PowerLed 1 и вместе с ней индикация заряда аккумулятора ChargOnLed. Ориентировочно через 5-6 часов индикатор ChargOnLed должен потухнуть и вместо него должна появится индикация ChargOffLed информирующая, что аккумулятор заряжен.

4.3. Методика контроля технического состояния

Устройство системы контроля доступом, так же, как и другое электронное устройство, требует регулярной проверки по техническому обслуживанию [20]. Под одним из видов работ стоит контроль технического состояния (КТС). КТС это проверка контрольных параметров и исправности устройства. У устройства есть два режима работы:

1) Штатное – когда питание устройства осуществляется от внешних 5В

2) Автономное – когда питание устройство осуществляется от встроенного аккумулятора.

Для проверки работы устройства на электрической принципиальной схеме были выставлены контрольные (диагностические) точки. На рисунке 30 изображена электрическая принципиальная схема с контрольными точками для диагностики устройства. По контрольным точкам, проверяется уровень и форма сигнала в определённым момент времени работы устройства при помощи измерительного оборудования: вольтметра; осциллографа.

Проверка начинается с более простых измерений уровня напряжения. Устройство должно быть включено. Необходим вольтметр измеряющий постоянное напряжение. Минусовым (чёрным) щупом касаемся контрольной точки GND. Не убирая чёрный щуп с точки GND, прикасаемся плюсовым (красным) щупом к точкам: F2, H1, C1, B2, C2.

1) Контрольная точка F2 – в штатном режиме на контрольной точке F2 напряжение составляет 5В±5%. В автономном режиме – 0В±5%. Контрольная точка показывает уровень входного напряжения устройства.

2) Контрольная точка H1 – в штатном режиме на контрольной точке H1 напряжение составляет 4.8В±5%. В автономном режиме – от 3,5 до 4.2В±5%.

3) Контрольная точка H2 – в штатном режиме на контрольной точке H2 напряжение составляет 3.3В±5%. В автономном режиме – 3.3В±5%. Контрольная точка показывает уровень внутреннего напряжения 3.3В, питающую управляющую часть устройства.

4) Контрольная точка B2 – в штатном режиме на контрольной точке B2 напряжение составляет 12В±5%. В автономном режиме – 12В±5%. Контрольная точка показывает уровень внутреннего напряжения 12В, питание необходимо для открытия замка.

5) Контрольная точка C2 – в штатном режиме на контрольной точке C2 напряжение составляет 3.3В±5%. В автономном режиме – 3.3В±5%. Точка C2 это логическое состояние ножки “Reset”, показывающая состояние работы микроконтроллера. 0В – работа микроконтроллера остановлена, 3.3В – микроконтроллер работает штатном режиме.

Дальше КТС производится при помощи осциллографа. Снимается осциллограмма контрольных точек с проверяемого устройства и далее сравниваются с заведомо правильным сигналом (осциллограммой). Для проверки понадобится осциллограф с двумя сигнальными входами.

Прикасаемся щупом канала 1 к контрольной точке A, а общим проводом щупа к контрольной точке GND. Сигнал записывается во время открытия замка. Измеренные сигналы должны совпадать с эталонной осциллограммой 1, обозначенной на рисунке 31. Сигнал 1 показывает уровень входного напряжения устройства во время открытия замка. По осциллограмме 1 видно просадку по напряжению.

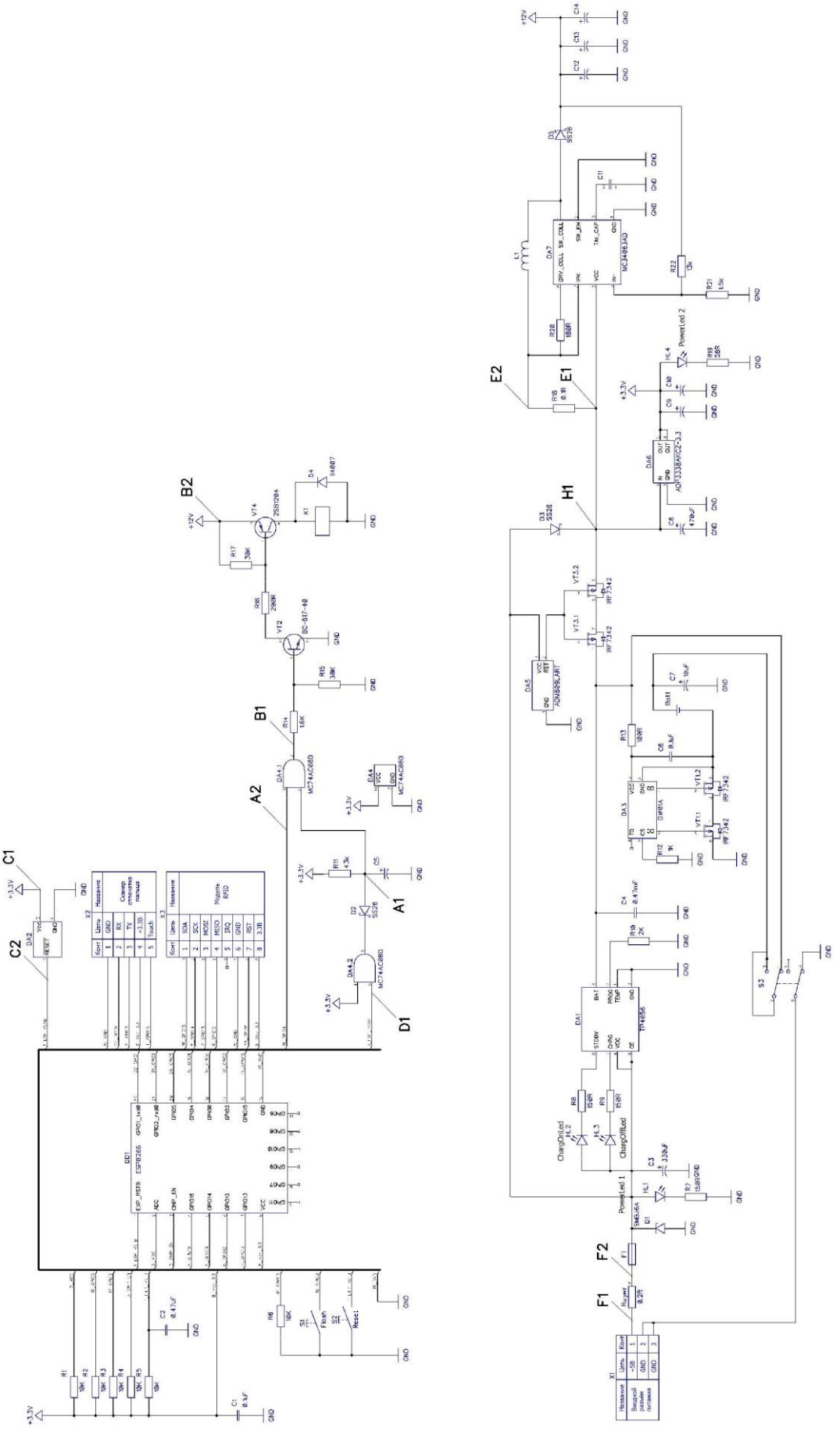


Рисунок 30 - Электрическая принципиальная схема с контрольными точками.

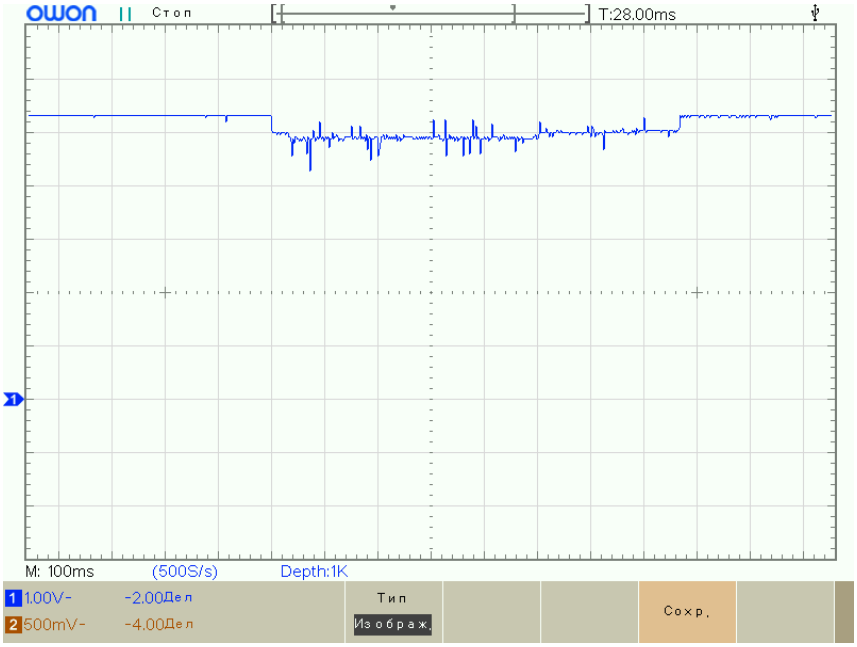


Рисунок 31 - Эталонная осциллограмма 1.

Прикасаемся щупом канала 1 к контрольной точке A1, а щупом канала 2 к контрольной точке A2. Подключаем два общих провода щупов к контрольной точке GND. Устройство должно быть отключено, после подключения щупов, включается регистрация сигналов на осциллографе и включается устройство. Измеренные сигналы должны совпадать с эталонной осциллограммой 2, обозначенной на рисунке 32. Сигнал 1 показывает уровень заряда конденсатора, как только напряжение превысит 1.5В (уровень логического срабатывания микросхемы), логическая микросхема разрешит управлением электромеханическим замком. Сигнал 2 это логический сигнал открытия замка. Как видно из эталонной осциллограммы на сигнале 2 присутствует короткий импульс (примерно 70мс), он возникает при включении устройства из-за особенностей микроконтроллера.

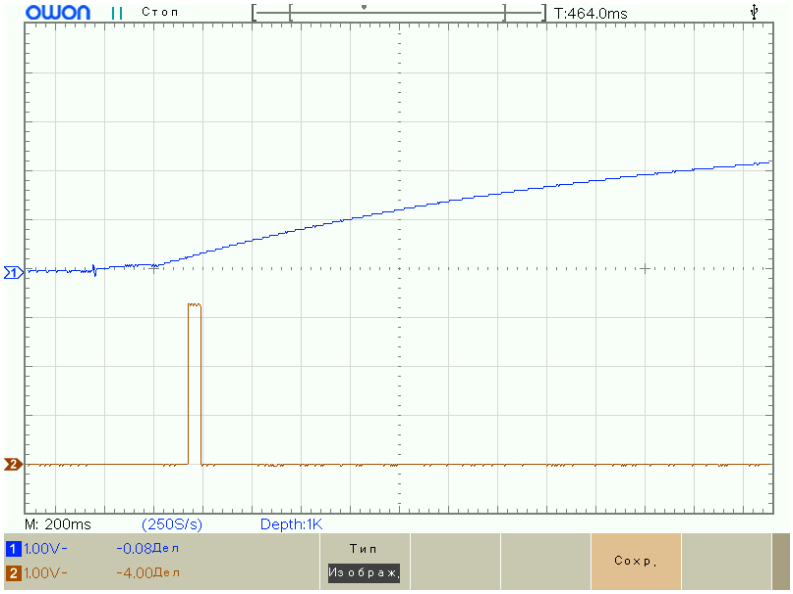


Рисунок 32 - Эталонная осциллограмма 2.

Прикасаемся щупом канала 1 к контрольной точке B1, а щупом канала 2 к контрольной точке B2. Подключаем два общих провода щупов к контрольной точке GND. Устройство должно находится в штатном режиме. Сигналы записываются во время открытия замка. Измеренные сигналы должны совпадать с эталонной осциллограммой 3, обозначенной на рисунке 33. Сигнал 1 - это логический сигнал открытия замка. Сигнал 2 показывает напряжение 12В питания во время открытия замка.

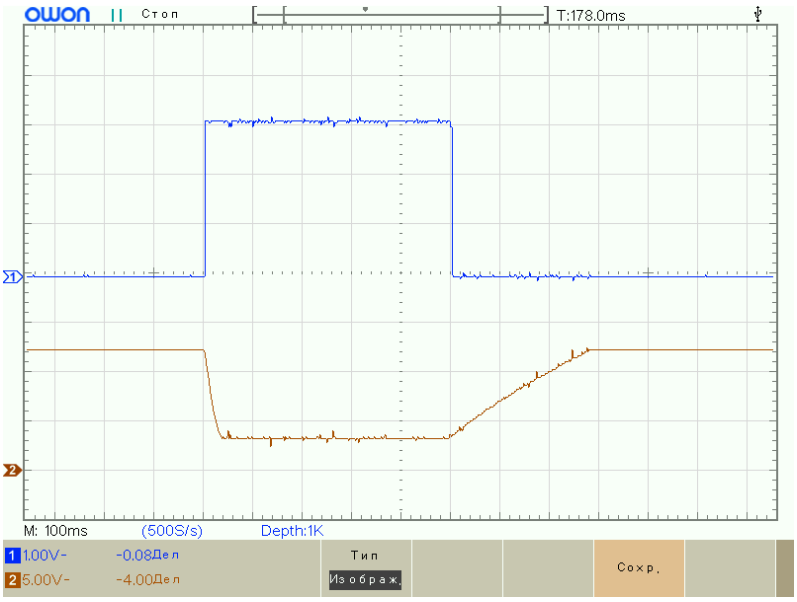


Рисунок 33 - Эталонная осциллограмма 3.

Прикасаемся щупом канала 1 к контрольной точке B1, а щупом канала 2 к контрольной точке B2. Подключаем два общих провода щупов к контрольной точке GND. Устройство должно находится в автономном режиме. Сигналы записываются во время открытия замка. Измеренные сигналы должны совпадать с эталонной осциллограммой 4, обозначенной на рисунке 34. Сигнал 1 - это логический сигнал открытия замка. Сигнал 2 показывает напряжение 12В питания во время открытия замка.

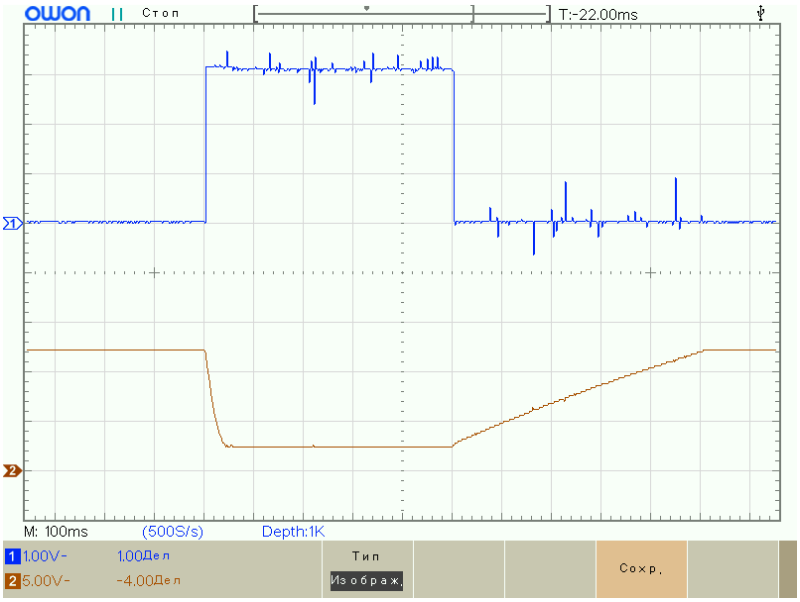


Рисунок 34 - Эталонная осциллограмма 4.

Прикасаемся щупом канала 1 к контрольной точке C1, а щупом канала 2 к контрольной точке C2. Подключаем два общих провода щупов к контрольной точке GND. Устройство должно находится в штатном режиме. Сигналы записываются во время пропажи питания 3.3В. Измеренные сигналы должны совпадать с эталонной осциллограммой 5, обозначенной на рисунке 35. Сигнал 1 это напряжение питания 3.3В. Сигнал 2 это логическое состояние ножки “Reset”, показывающая состояние работы микроконтроллера. Как видно из осциллограммы 5, при пропаже 3.3В сигнал “Reset” сразу опускается до логического 0. Это происходит благодаря микросхеме супервизора, которая перезагружает микроконтроллер при пониженном или повышенном питании.

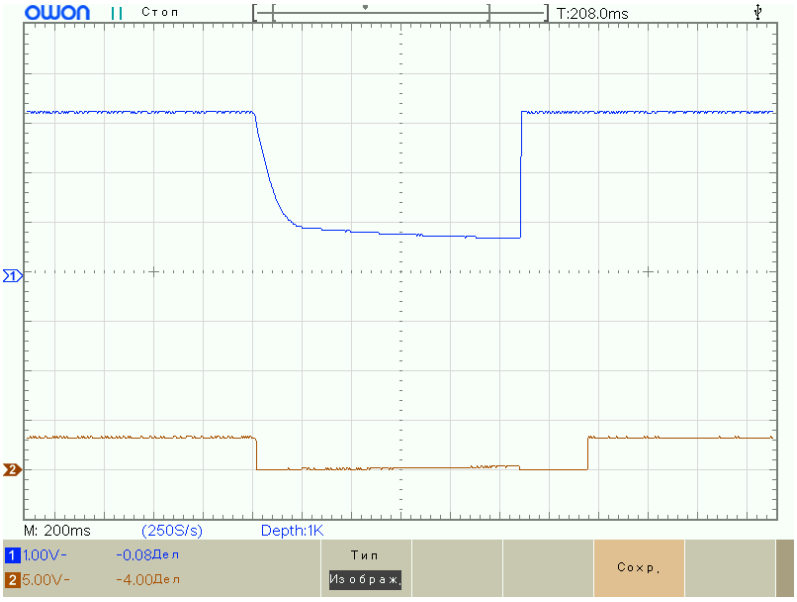


Рисунок 35 - Эталонная осциллограмма 5.

Прикасаемся щупом канала 1 к контрольной точке D1, а щупом канала 2 к контрольной точке A1. Подключаем два общих провода щупов к контрольной точке GND. Устройство должно находится в штатном режиме. Сигналы записываются при нажатии кнопки “RESET”. Измеренные сигналы должны совпадать с эталонной осциллограммой 6, обозначенной на рисунке 36. Сигнал 1 это логическое состояние ножки “Reset”, показывающая состояние работы микроконтроллера. Сигнал 2 это напряжение времязадающего конденсатора.

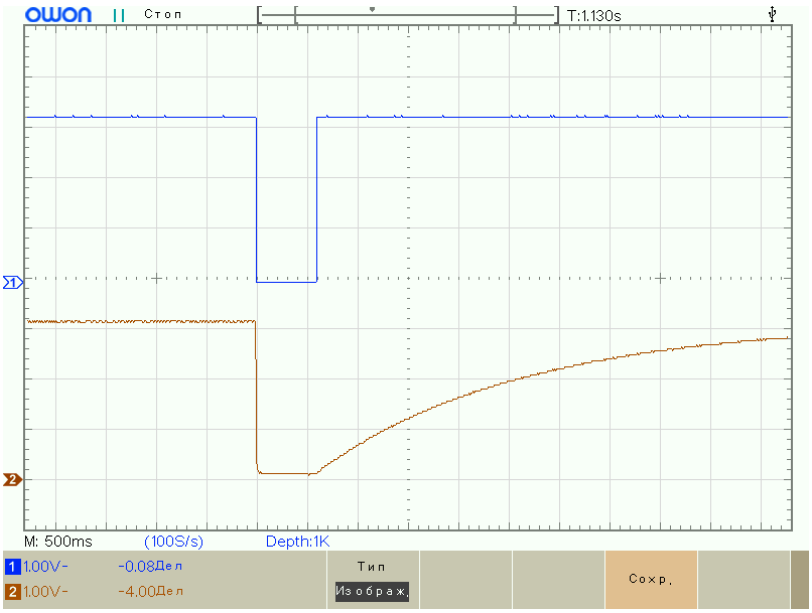


Рисунок 36 - Эталонная осциллограмма 6.

Прикасаемся щупом канала 1 к контрольной точке E1, а общим проводом к контрольной точке E2. Устройство должно находится в штатном режиме. Сигнал записывается во время открытия замка. Измеренные сигналы должны совпадать с эталонной осциллограммой 7, обозначенной на рисунке 37. Сигнал 1 это падение напряжения на шунте импульсного преобразователя напряжения во время открытия замка.

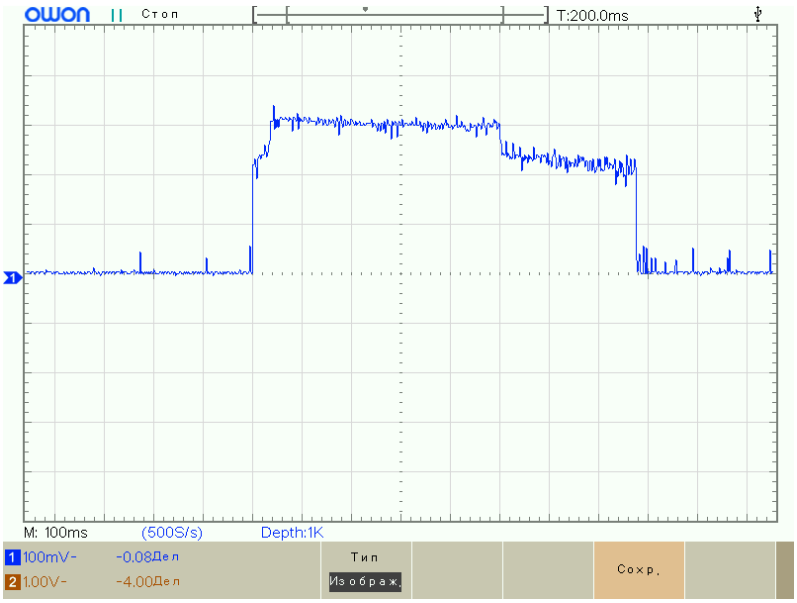


Рисунок 37 - Эталонная осциллограмма 7.

Прикасаемся щупом канала 1 к контрольной точке E1, а общим проводом к контрольной точке E2. Устройство должно находится в автономном режиме. Сигнал записывается во время открытия замка. Измеренные сигналы должны совпадать с эталонной осциллограммой 7, обозначенной на рисунке 38. Сигнал 1 это падение напряжения на шунте импульсного преобразователя напряжения во время открытия замка.

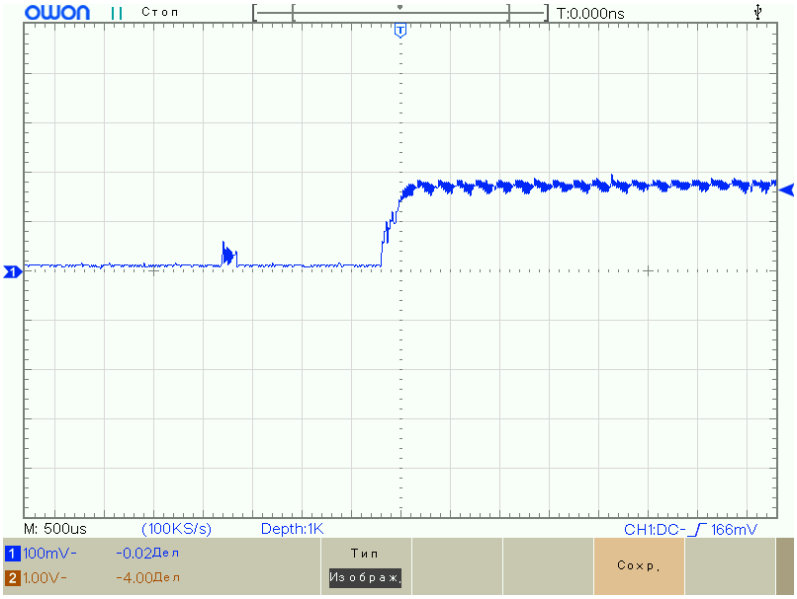


Рисунок 38 - Эталонная осциллограмма 7.

Прикасаемся щупом канала 1 к контрольной точке F1, а общим проводом к контрольной точке F2. Устройство должно находится в штатном режиме. Сигнал записывается во время открытия замка. Так же необходимо поставить шунт (резистор), номиналом в 0.2 Ом, в разрыв провода внешнего источника питания (по положительному проводу). Измеренные сигналы должны совпадать с эталонной осциллограммой 8, обозначенной на рисунке 39. Сигнал 1 это падение напряжения на шунте, установленным по входному питанию.

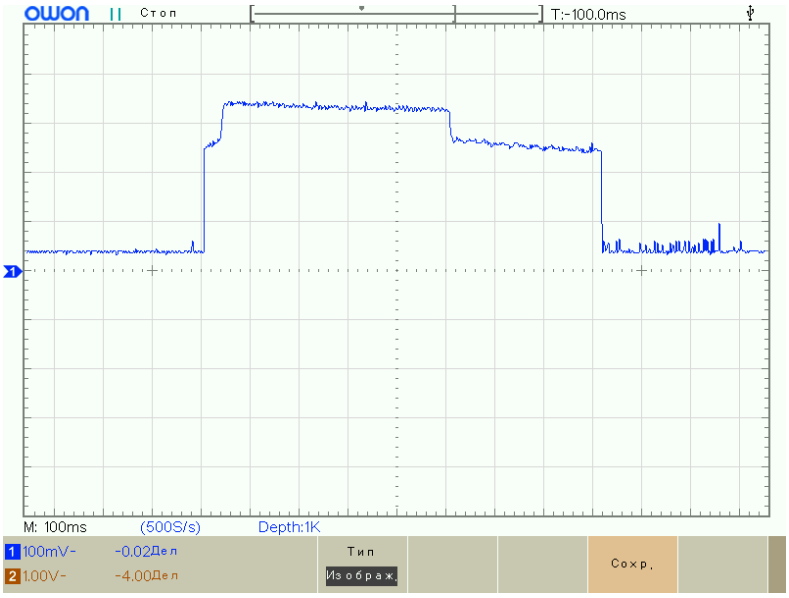


Рисунок 39 - Эталонная осциллограмма 8.

Прикасаемся щупом канала 1 к контрольной точке G1, а общим проводом щупа к контрольной точке GND. Сигнал записывается во время касания пальцем сканера. Измеренные сигналы должны совпадать с эталонной осциллограммой 9, обозначенной на рисунке 40. Сигнал 1 показывает нам передачу данных по UART интерфейсу.

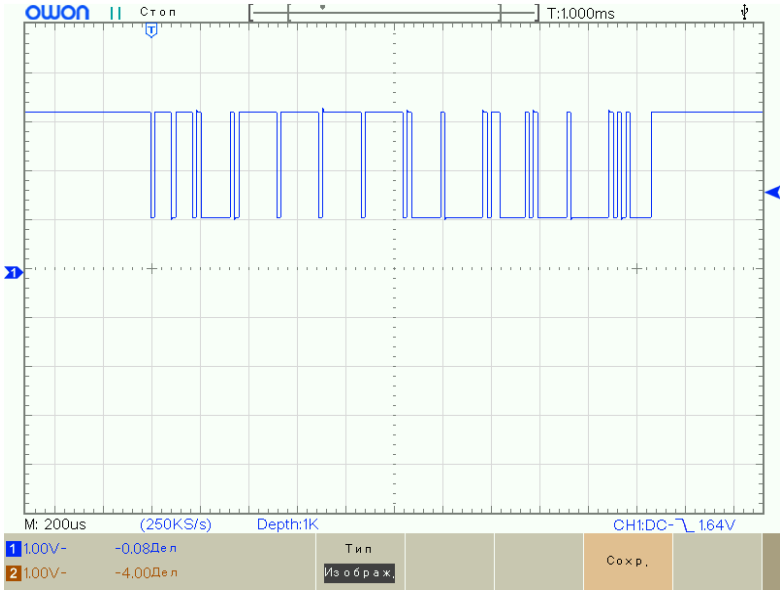


Рисунок 40 - Эталонная осциллограмма 9.

Прикасаемся щупом канала 1 к контрольной точке H1, а щупом канала 2 к контрольной точке C1. Подключаем два общих провода щупов к контрольной точке GND. Устройство должно находится в штатном режиме. Сигналы записываются при открытии замка. Измеренные сигналы должны совпадать с эталонной осциллограммой 10, обозначенной на рисунке 41. Сигнал 1 это напряжение после цепи переключения питания. Сигнал 2 это напряжение питания 3.3В.

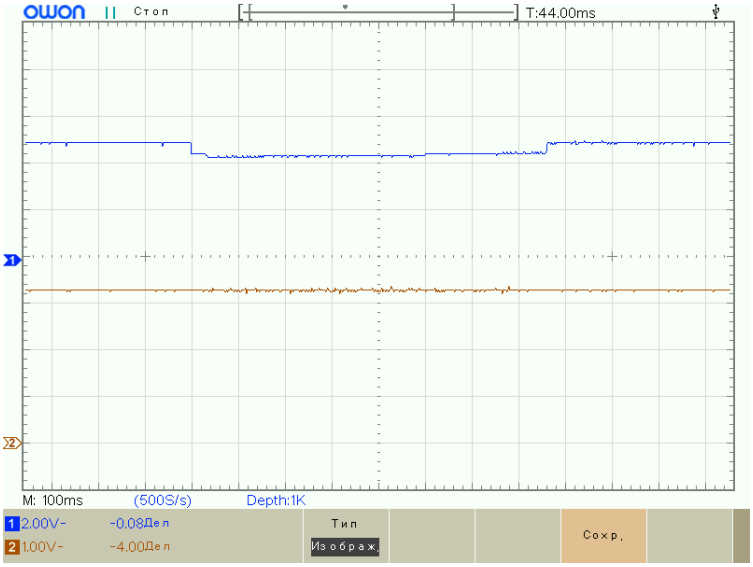


Рисунок 41 - Эталонная осциллограмма 10.

Прикасаемся щупом канала 1 к контрольной точке H1, а щупом канала 2 к контрольной точке C1. Подключаем два общих провода щупов к контрольной точке GND. Устройство должно находится в автономном режиме. Сигналы записываются при открытии замка. Измеренные сигналы должны совпадать с эталонной осциллограммой 11, обозначенной на рисунке 42. Сигнал 1 это напряжение после цепи переключения питания. Сигнал 2 это напряжение питания 3.3В.

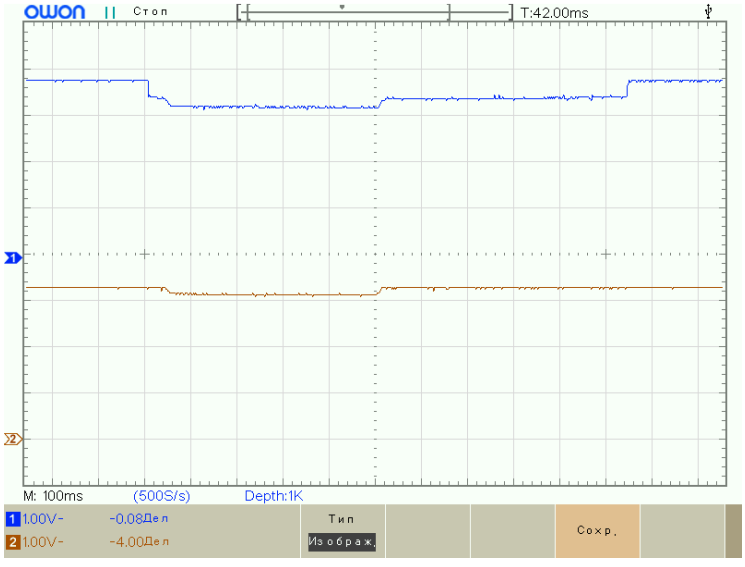


Рисунок 42 - Эталонная осциллограмма 11.

4.4. Инструкция по эксплуатации устройства

Система контроля доступом(СКД) питается от внешнего источника питания 5В 0.5А, но рекомендуется использовать блок питания на 5В 2А.

На передней панели имеются 3 индикатора:

1) PowerLed 1 (зелёный) наличие входного напряжения 5V

2) PowerLed 2 (зелёный) наличие внутреннего напряжения 3.3V

3) ChargOnLed (красный) индикатор заряда аккумулятора.

На рисунке 43 изображена передняя панель СКД с индикаторами.

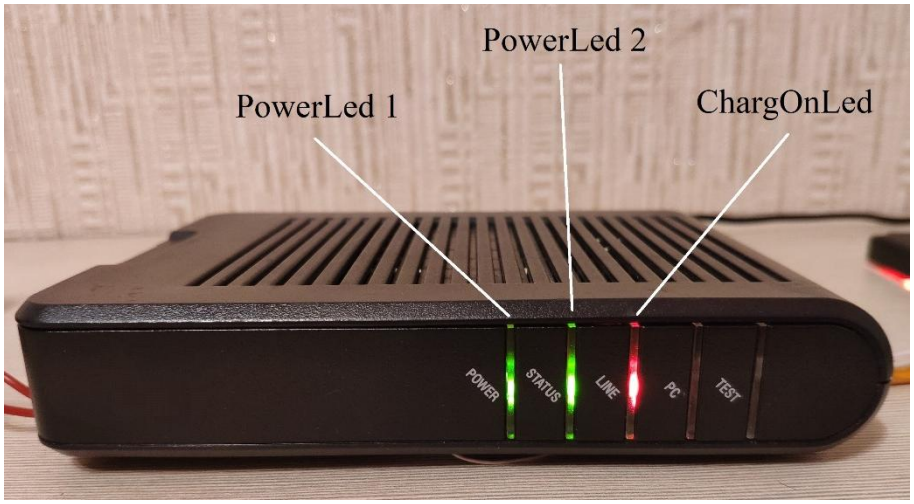


Рисунок 43 - Передняя панель СКД с индикаторами.

На задней части СКД расположен коннектор питания и кнопка включения. На рисунке 44 изображена задняя панель СКД.

Рисунок 44 - Задняя панель СКД.

СКД имеет 3 способа открытия замка: по метке (RFID); по отпечатку пальца; по удалённому способу, через Wi-Fi подключение.

Настройка доступа по Wi-Fi. Включаем СКД. Далее необходимо подключиться к Wi-Fi СКД (точке доступа), для этого запускаем поиск Wi-Fi, например, на смартфоне (управляющим устройством может быть любое устройство имеющее технологию Wi-Fi: ноутбук, планшет, смартфон и т.д.), находим точку доступа под названием “System Control Access”, и подключаемся к ней. При подключении потребуется ввести пароль. Как подключение будет осуществлено, заходим в браузер и в строке поиска вводим адрес “192.168.4.1” и нажимаем ввод. Далее появляется меню из 5 пунктов:

1) Запись мастера – команда, позволяющая записать на пустую RFID метку, ключ метки мастера.

2) Запись ключа – команда, записывает на пустую RFID метку, ключ пользователя, этаже команда позволяет записать палец в базу данных. После нажатия команды “запись ключа” приложить палец к сканеру три раза или приложить пустую метку к RFID части для записи ключа пользователя.

3) Открыть замок – команда, открывает замок СКД.

4) Стереть отпечатки – стирает базу данных отпечатков пальцев.

5) Стереть метку – команда, удаляет ключ, приложенной к RFID части, с метки пользователя или метки мастера.

Настройка RFID. RFID не нуждается в настройки. Для записи меток пользователя необходимо приложить и убрать метку мастера к RFID части, устройство войдёт в режим записи меток пользователя, далее необходимо приложить пустую метку. По завершению записи, метку можно использовать для открытия замка. Записать ключ можно и по Wi-Fi (2 пункт меню). Для открытия замка нужно поднести метку пользователя к RFID части.

Настройка сканера отпечатка пальца. Для добавления отпечатков пальцев в базу данных, необходимо приложить метку мастера к RFID части. Далее приложить палец 3 раза одной и той же стороной к сканеру отпечатка пальца. После завершения записи, сканер сможет узнавать пользователя и

соответственно открыть ему замок СКД. Записать палец можно и по Wi-Fi (2 пункт меню). Для открытия замка, необходимо приложить записанный палец к сканеру отпечатка пальца.

На рисунке 45 изображено расположение внутренних кнопок.

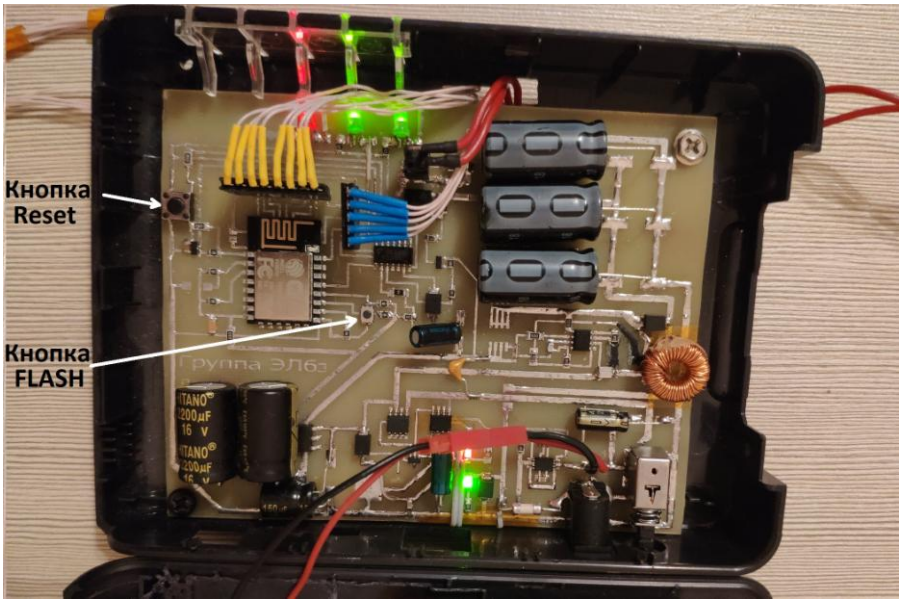


Рисунок 45 – Расположение внутренних кнопок.

ЗАКЛЮЧЕНИЕ

Выпускная квалификационная работа на тему «Система контроля доступа в аудиториях кафедры "Промышленная электроника"» разрабатывался в соответствии с требованиями всех разделов расчетно-пояснительной записки.

При создании проекта автором была осуществлена работа по анализу современных решениях и выбору основных параметрах устройства. Была разработана электрическая структурная схема, а также схема электрическая принципиальная, после чего произведён расчёт узлов устройства. На основе электрической принципиальной схемы разработана печатная плата. Разработано собственное программное обеспечение для работы микроконтроллера.

Был осуществлён расчёт надёжности устройства, который выявил время работы устройства на отказ в районе 3 лет. Посчитана стоимость устройства.

Результатом разработки устройства стало создание рабочего прототипа для демонстрации стабильной работы и верности произведённых расчётов. Для устройства так же была разработана наладка перед первым запуском. Составлен метод испытания для выявления неисправностей. Для дальнейшего обслуживания и ремонта была составлена методика контроля технического состояния.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Первые замки. // Александр Лаврус, Анна Лаврус 2002г. URL http://n-t.ru/tp/it/pz2.htm/. (Дата обращения: 02.04.2019).
2. Комаров А.И. ПОСТРОЕНИЕ СИСТЕМЫ БЕЗОПАСНОСТИ С ПОМОЩЬЮ QR-КОДА И БИОМЕТРИЧЕСКИХ ДАННЫХ ЧЕЛОВЕКА // Студенческий: электрон. научн. журн. 2018. № 9(29). URL: https://sibac.info/journal/student/29/105528 (дата обращения: 6.04.2019).
3. СКУД 27 с доступом по отпечатку пальца и карте с электромеханическим врезным замком. // hrobot.ru. 2016. https://www.hrobot.ru/product/komplekt-27-skud-s-dostupom-po-otpechatku-paltsa-i-karte-s-elektromekhanicheskim-vreznym-zamkom/. (дата обращения: 9.04.2019).
4. Деревянко Д.М., Бочаров Н.Г., Васин Д.А. [и др.] РАЗРАБОТКА ЭЛЕКТРИЧЕСКИХ СХЕМ С ПРИМЕНЕНИЕМ КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ // Научное сообщество студентов: МЕЖДИСЦИПЛИНАРНЫЕ ИССЛЕДОВАНИЯ: сб. ст. по мат. III междунар. студ. науч.-практ. конф. № 3. URL: sibac.info/sites/default/files/conf/file/stud_3_3.pdf. (дата обращения: 15.04.2019).
5. Ремизевич Т. В. Микроконтроллеры для встраиваемых приложений: от общих подходов – к семействам HC05 и HC08 фирмы Motorola. / под ред. Кирюхина И. С. – М.: ДОДЭКА, 2000. – 272с. (дата обращения: 17.04.2019).
6. ESP8266 datasheet and tutorial. // Technical Reference 2017. URL: https://www.espressif.com/sites/default/files/documentation/esp8266technical_reference_en.pdf. (дата обращения: 20.04.2019).
7. Арутюнян Т.В., Пузановский К.В., Шуткин И.Ю. [и др.] СХЕМОТЕХНИКА СТАБИЛИЗАТОРОВ НАПРЯЖЕНИЯ // Научное сообщество студентов XXI столетия: сб. ст. по мат. XLIX междунар. студ. науч.-практ. конф. № 1(48). URL: https://sibac.info/archive/technic/1(48).pdf (дата обращения: 23.04.2019).
8. MC34063 Datasheet. // STMicroelectronics 2001. URL: http://pdf1.alldatasheet.com/datasheetpdf/view/92958/STMICROELECTRONICS/MC34063.html. (дата обращения: 26.04.2019).
9. Batteries types, differences and features. // Copyright. 2007-2009. URL: http://www.powerinfo.com/accumulatortype.php. (дата обращения: 01.05.2019).
10. TP4056 datasheet and tutorial. // Chinagoods 2015. URL: http://chinaguds.ru/goods-with-aliexpress/modul-zaryada-micro-usb-tp4056-5v-1a-s-

zashhitoj-plata-kontrolya-zaryada-razryada-li-ion-akkumulyatora-18650.html. (дата обращения: 02.05.2019).

11. Электромеханический замок Falcon Eye FE-2369. // Falcon Eye 2014. URL: https://falconeye.su/manuals/FE-2369_2369i_2370.pdf. (дата обращения: 02.05.2019).

12. Титце У., Шенк К. Полупроводниковая схемотехника: Справочное ру-ководство. Пер. с нем. – М.: Мир, 2014. – 512. (дата обращения: 04.05.2019).

13. Светодиоды - маркировка, характеристика, подключение. // Светодиодный мир нашего века. 2012. URL: https://svetodiode.blogspot.com/2012/01/blog-post_19.html. (дата обращения: 05.05.2019).

14. Иванова А.В. АНАЛИЗ НАДЕЖНОСТИ ТЕХНИЧЕСКИХ СИСТЕМ НА ОСНОВЕ СТРУКТУРНО-ЛОГИЧЕСКИХ СХЕМ: ОСНОВНЫЕ МЕТОДЫ // сб. ст. по мат. XLVI междунар. студ. науч.-практ. конф. № 11(46). URL: https://sibac.info/archive/meghdis/11(46).pdf (дата обращения: 6.05.2019).

15. Емельянова Д.К. СОЗДАНИЕ ТОПОЛОГИИ ПЕЧАТНЫХ ПЛАТ КАК ВАЖНЫЙ ЭТАП ПРОЕКТИРОВАНИЯ ЭЛЕКТРОННЫХ УСТРОЙСТВ // Научное сообщество студентов XXI столетия. ТЕХНИЧЕСКИЕ НАУКИ: сб. ст. по мат. LX междунар. студ. науч.-практ. конф. № 12(59). URL: https://sibac.info/archive/technic/12(59).pdf (дата обращения: 08.05.2019).

16. Еремин А.Н. ОЧЕРЕДНОСТЬ ПРОКЛАДКИ СОЕДИНЕНИЙ ПЕЧАТНЫХ ПЛАТ // ТЕХНИЧЕСКИЕ НАУКИ: сб. ст. по мат. LXII междунар. студ. науч.-практ. конф. № 2(61). URL: https://sibac.info/archive/technic/2(61).pdf (дата обращения: 08.05.2019).

17. Расчет ширины дорожки печатной платы в зависимости от силы тока: [Электронный ресурс]. RadioProg. 2017. URL: http://radioprog.ru/post/257. (дата обращения: 08.05.2019).

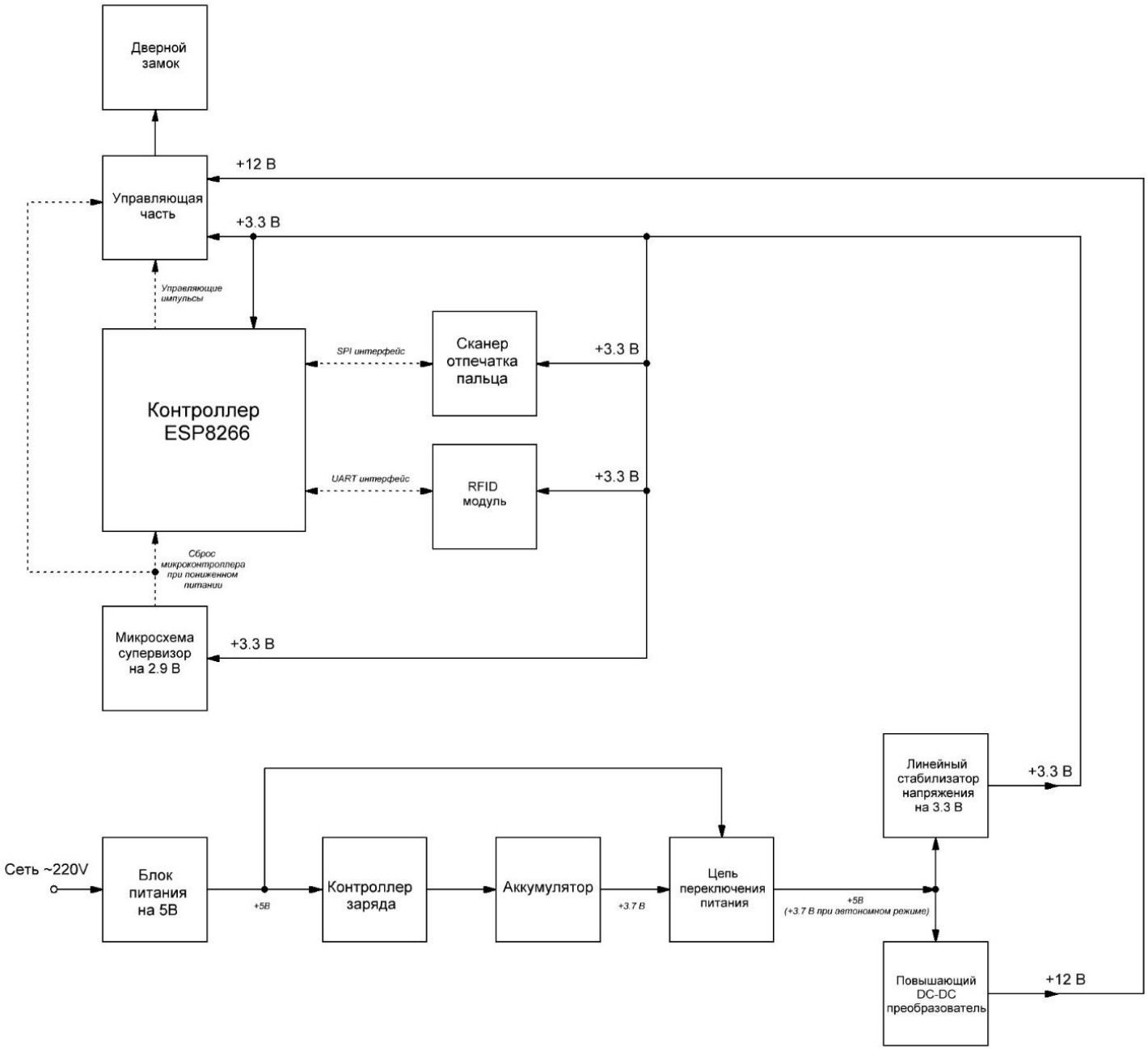
18. Боровский, А.Н. Qt4.7+. Практическое программирование на C++. / А.Н. Боровский. - СПб.: BHV, 2012. - 496 с. (дата обращения: 10.05.2019).

19. Модуль: CH340G. // www.5v.ru. 2013. URL: http://www.5v.ru/ch340g.htm. (дата обращения: 12.05.2019).

20. Дробышевский И.С., Полянский Н.А. СПОСОБЫ КОНТРОЛЯ, СНЯТИЯ И ПЕРЕДАЧИ ДАННЫХ С ПРИБОРОВ УЧЕТА ЭНЕРГОРЕСУРСОВ // Научное сообщество студентов XXI столетия. ТЕХНИЧЕСКИЕ НАУКИ: сб. ст. по мат. XLVII междунар. студ. науч.-практ. конф. № 10(46). URL: https://sibac.info/archive/technic/10(46).pdf (дата обращения: 12.05.2019)

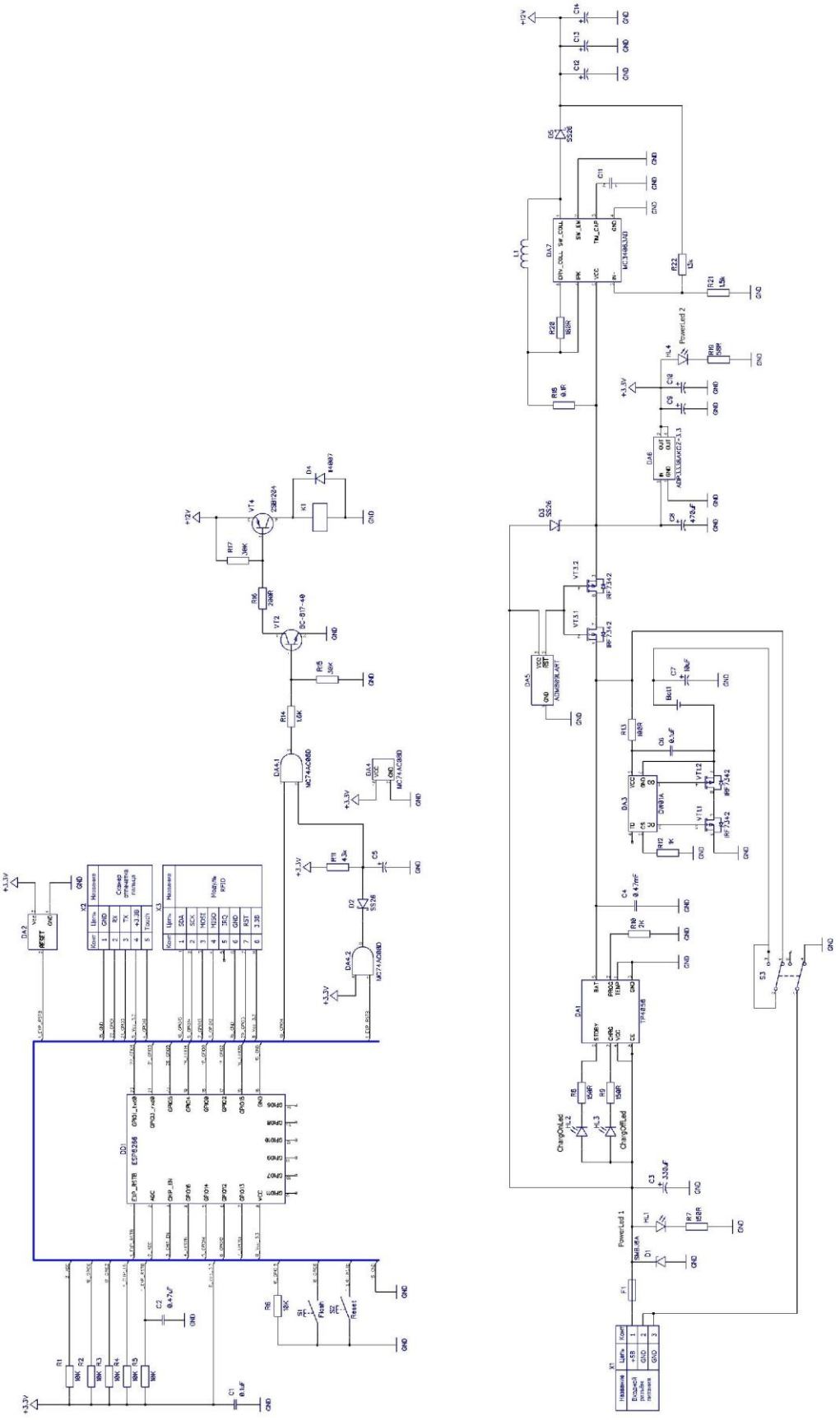
Приложение А

Электрическая структурная схема системы контроля доступом.



Приложение Б

Электрическая принципиальная схема системы контроля доступом.



Приложение В

Программное обеспечения микроконтроллера.

Вкладка “Security”:

```
#include <SPI.h>
#include <MFRC522.h>
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
#include <ESP8266mDNS.h>
#include <Adafruit_Fingerprint.h>

void setup() {
  _tmain();
}
void loop() {
  _tloop();
}
```

Вкладка “A_SETTING”:

```
#define RST_PIN         D2
#define SS_PIN          D8
#define LOCK_PIN        D2
#define LED_PIN         D3
#define TOUCH_PIN       D0
```

Вкладка “A_STRUCT”:

```
struct sImprint {
  byte imprint;
  int count;
  byte stage;
};

struct Device{
  byte status;
```

```
unsigned long time;
  byte WiFinfo;
  sImprint imprint;
};

Device Device;
```

Вкладка “B_OBJECT”:

```
MFRC522 mfrc522(SS_PIN, RST_PIN);

Adafruit_Fingerprint imprint = Adafruit_Fingerprint(&Serial);

ESP8266WebServer server(80);
```

Вкладка “Device”:

```
void _tmain() {
  pinMode(LOCK_PIN, OUTPUT);
  pinMode(LED_PIN, OUTPUT);
  pinMode(TOUCH_PIN, INPUT);

  Serial1.begin(9600);
  SPI.begin();
  mfrc522.PCD_Init();
  ESP.wdtFeed();

  Device.status = 0;

  digitalWrite(LOCK_PIN, LOW);
  digitalWrite(LED_PIN, LOW);

  WiFiSetup();

  imsetup();

  Serial2.print("Device open");
```

```
void _tloop() {
   switch(Device) {
    case 0: // Обычный статус считывания
     if (RFIDbegin()) {
      switch(RFIDReadUID()) {
       case 0:
        Serial1.print("Read ERROR");
        break;
       case 1:
        Serial1.println("Read UID key");
        OpenLock();
        break;
       case 2:
        Serial1.println("Read UID master");
        Device.status = 1;
        Device.time = millis();
        Device.imprint.stage = 0;
        digitalWrite(LED_PIN, HIGH);
        break;
      };
      RFIDend();
     }
     break;
    case 1: // Запись
     if ( (millis() - Device.time) > (1000 * 30) ) {
      Device.status = 0;
      digitalWrite(LED_PIN);
     }
     if (RFIDbegin()) {
      if (RFIDWrite()) {
       Device.status = 0;
       digitalWrite(LED_PIN);
       Serial1.println("Write UID key");
      }
      RFIDend();
     }
     break;
```

```
 case 2: // Запись мастера
       if ( (millis() - Device.time) > (1000 * 30) ) {
         Device.status = 0;
         digitalWrite(LED_PIN, LOW);
       }
       if (RFIDbegin()) {
         if (RFIDWriteMASTER()) {
           Device.status = 0;
           digitalWrite(LED_PIN, LOW);
           Serial1.print("Write UID key");
         }
         RFIDend();
       }
       break;
     case 3: // Отчистка метки
       if ( (millis() - Device.time) > (1000 * 30) ) {
         Device.status = 0;
         digitalWrite(LED_PIN, LOW);
       }
       if (RFIDbegin()) {
         if (RFIDWriteUIDCLEAR()) {
           Device.status = 0;
           digitalWrite(LED_PIN, LOW);
           Serial1.println("Write UID key");
         }
         RFIDend();
       }
       break;
   };
   imloop();
   WiFiLoop();
   ESP.wdtFeed();
}
```

Вкладка “IMPRINT”:

```
void imsetup() {
```

```
imprint.begin(57600);
  Device.imprint.stage = 0;
  if (imprint.verifyPassword()) {
    Device.imprint.imprint = 1;
    Device.imprint.count = imprint.getTemplateCount();
    Device.imprint.count = imprint.templateCount;
    Serial1.println("Open imprint1");
    Serial1.print("Count1: ");
    Serial1.println(Device.imprint.count1);
  }else{
    Device.imprint.imprint = 0;
    Serial1.print("imprint ERROR1");
  }
}

void imloop() {
  if (Device.imprint. = 1) return;
  if (digitalRead(TOUCH_PIN) == 1) return;
  Serial1.println("imloop()");
  int p = -1;
  while (p = FINGERPRINT_OK) {
    p = imprint.getImage();
    switch (p) {
      case FINGERPRINT_NOFINGER:
        Serial2.println("FINGERPRINT_NOFINGER");
        return;
    }
  }

  switch (Device.status) {
    case 0:
      imdefault();
      return;
    case 1:
      imwrite();
      return;
  }
```

```
void imdefault() {
  Serial1.println("imdefault()");
  if (imprint.image2Tz(1) != FINGERPRINT_OK) return;
  if (imprint.fingerFastSearch() != FINGERPRINT_OK) return;
  Serial1.print("Found ID #");
  Serial1.println(imprint.fingerID);
  OpenLock();
}

void imwrite() {
  Serial1.println("imwrite()");
  switch (Device.imprint.stage) {
    case 0:
      Serial1.println("stage 0");
      if (imprint.image2Tz(1) == FINGERPRINT_OK) {
        // delay(20);
        while (imprint.getImage() != FINGERPRINT_NOFINGER) { }
        Device.imprint.stage = 1;
        Serial1.print("stage 0 - end");
      }else{
        Serial1.print("WRITE ERROR");
      }
      break;
    case 1:
      Serial1.println("stage 1");
      if (imprint.image2Tz(2) == FINGERPRINT_OK) {
        if (imprint.createModel() == FINGERPRINT_OK) {
          if (imprint.storeModel(Device.imprint.count + 1) == FINGERPRINT_OK) {
            Device.imprint.count = Device.imprint.count + 1;
            if (Device.imprint.count > 127) {
              Device.imprint.count = 1;
            }
            Device.status = 0;
            Serial1.println("WRITE OK");
          }else{
            Serial1.println("storeModel ERROR");
          }
        }else{
```

```
        Serial1.println("createModel ERROR");
       }
      }else{
       Serial1.println("WRITE ERROR");
      }
      break;
  }
}

void imdell() {
  for (int i = 0; i <= 127; i++) {
    imprint.deleteModel(i);
  }
}
```

Вкладка “Lock”:

```
void OpenLock() {
  digitalWrite(LOCK_PIN, HIGH);
  delay(500);
  digitalWrite(LOCK_PIN, LOW);
  delay(500);
}
```

Вкладка “RFID”:

```
#define NR_KNOWN_KEYS   9

byte knownKeys[NR_KNOVN_KEYS][MFRC522::MM_KEY_SIZE] =  {
    {0xff, 0xff, 0xff, 0xff, 0xff, 0xff}, // FF FF FF FF FF FF
    {0xa0, 0xa1, 0xa2, 0xa3, 0xa4, 0xa5}, // A0 A1 A2 A3 A4 A5
    {0xb0, 0xb1, 0xb2, 0xb3, 0xb4, 0xb5}, // B0 B1 B2 B3 B4 B5
    {0x4d, 0x3a, 0x99, 0xc3, 0x51, 0xdd}, // 4D 3A 99 C3 51 DD
    {0x1a, 0x98, 0x2c, 0x7e, 0x45, 0x9a}, // 1A 98 2C 7E 45 9A
    {0xd3, 0xf7, 0xd3, 0xf7, 0xd3, 0xf7}, // D3 F7 D3 F7 D3 F7
    {0xaa, 0xbb, 0xcc, 0xdd, 0xee, 0xff}, // AA BB CC DD EE FF
    {0x00, 0x00, 0x00, 0x00, 0x00, 0x00}  // 00 00 00 00 00 00
    };
```

```
byte RFIDUIDKEY[16]    = {0x08, 0x09, 0x06, 0x07, 0x04, 0x09, 0x00, 0x02, 0x00,
0x07, 0x02, 0xFF, 0xFF, 0xFF, 0xFF, 0x00};
byte RFIDUIDMASTER[16] = {0x08, 0x09, 0x02, 0x07, 0x08, 0x09, 0x00, 0x00,
0x03, 0x01, 0x08, 0xFF, 0xFF, 0xFF, 0xFF, 0x00};
byte RFIDUIDCLEAR[16]  = {0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00};

bool RFIDbegin() {
  if (  mfrc522.PICC_IsNewCardPresent() ) return false;
  if (  mfrc522.PICC_ReadCardSerial() ) return false;
  return true;
}

void RFIDend() {
  mfrc522.PICC_HaltA();
  mfrc522.PCD_StopCrypto1();
}

byte RFIDReadUID() {
  MFRC522::MIFARE_Key key;
  MFRC522::StatusCode status;
  byte _RFIDUIDKEY[18];
  int _cont = 0;
  byte byteCount = sizeof(_RFIDUIDKEY);
  byte _ret;

  for (byte k = 0; k < NR_KNOWN_KEYS; k++) {
    for (byte i = 0; i < MFRC522::MF_KEY_SIZE; i++) {
      key.keyByte[i] = knownKeys[k][i];
    }
    _cont = 0;
    Label:
    _cont ++;
    if (!RFIDbegin() && _cont < 10) {
      ESP.wdtFeed();
      goto Label;
    }
```

```
 status = mfrc522.PCD_Authenticate(MFRC522::PICC_CMD_MF_AUTH_KEY_B,
0, &key, &(mfrc522.uid));
    if (status == MFRC522::STATUS_OK) {
      status = mfrc522.MIFARE_Read(1, _RFIDUIDKEY,
 &byteCount);
      if (status == MFRC522::STATUS_OK) {
        _ret = 0;
        for (int e = 0; e < 16 ; e++) {
          if (_RFIDUIDKEY[e] == RFIDUIDKEY[e]) {
            ESP.wdtFeed();
            _ret++;
          }
        }
        if (_ret >= 15) {
          ESP.wdtFeed();
          return 1;
        }
        _ret = 0;
        for (int e = 0; e < 16 ; e+) {
          if (_RFIDUIDKEY[e] == RFIDUIDMASTER[e]) {
            ESP.wdtFeed();
            _ret++;
          }
        }
        if (_ret >= 15) {
          ESP.wdtFeed();
          return 2;
        }
        ESP.wdtFeed();
        return 0;
      }
      RFIDend();
      ESP.wdtFeed();
      return 0;
    }
  }
  return 0;
}
```

```
bool RFIDWriteUID() {
  MFRC522::MIFARE_Key key;
  MFRC522::StatusCode status;
  byte _RFIDUIDKEY[18];
  byte byteCount = sizeof(_RFIDUIDKEY);
  int _cont = 0;
  for (byte k = 0; k < NR_KNOWN_KEYS; k+) {
    for (byte i = 0; i < MFRC522::MF_KEY_SIZE; i+) {
      key.keyByte[i] = knownKeys[k][i];
    }
    status =
mfrc522.PCD_Authenticate(MFRC522::PICC_CMD_MF_AUTH_KEY_B, 0, &key,
&(mfrc522.uid));
    if (status == MFRC522::STATUS_OK) {
      status = mfrc522.MIFARE_Read(1, _RFIDUIDKEY, &byteCount);
      if (status == MFRC522::STATUS_OK) {
        if (memcmp(&_RFIDUIDKEY[0], &RFIDUIDMASTER[0], 16) == 0) {
          ESP.wdtFeed();
          return false;
        }
        status = mfrc522.MIFARE_Write(1, RFIDUIDKEY, 16);
        if (status == MFRC522::STATUS_OK) {
          ESP.wdtFeed();
          return true;
        }
      }
      RFIDend();
      ESP.wdtFeed();
      return false;
    }
    RFIDend();
    ESP.wdtFeed();
  }
  return false;
}

bool RFIDWriteUIDMASTER() {
  MFRC522::MIFARE_Key key;
```

```
 MFRC522::StatusCode status;
  byte _RFIDUIDKEY[18];
  byte byteCount = sizeof(_RFIDUIDKEY);
  int _cont = 0;
  for (byte k = 0; k < NR_KNOWN_KEYS; k++) {
    for (byte i = 0; i < MFRC522::MF_KEY_SIZE; i++) {
      key.keyByte[i] = knownKeys[k][i];
    }
    _cont = 0;
    Label:
    _cont ++;
    if (!RFIDbegin() && _cont < 10) {
      ESP.wdtFeed();
      goto Label;
    }
    status =
mfrc522.PCD_Authenticate(MFRC522::PICC_CMD_MF_AUTH_KEY_B, 0, &key,
&(mfrc522.uid));
    if (status == MFRC522::STATUS_OK) {
      status = mfrc522.MIFARE_Read(1, _RFIDUIDKEY, &byteCount);
      if (status == MFRC522::STATUS_OK) {
        status = mfrc522.MIFARE_Write(1, RFIDUIDMASTER, 16);
        if (status == MFRC522::STATUS_OK) {
          ESP.wdtFeyd();
          return true;
        }
      }
      RFIDend();
      ESP.wdtFeed();
      return false;
    }
    RFIDend();
    ESP.wdtFeed();
  }
  return false;
}

bool RFIDWriteUIDCLEAR() {
```

```
 MFRC522::MIFARE_Key key;
  MFRC522::StatusCode status;
  byte _RFIDUIDKEY[18];
  byte byteCount = sizeof(_RFIDUIDKEY);
  int _cont = 0;
  for (byte k = 0; k < NR_KNOWN_KEYS; k++) {
    for (byte i = 0; i < MFRC522::MF_KEY_SIZE; i++) {
      keyByte[i] = knownKeys[k][i];
    }
    _cont = 0;
    Label:
    _cont ++;
    if (!RFIDbegin() && _cont < 10) {
      ESP.wdtFeed();
      goto Label;
    }
    status =
mfr522.PCD_Authenticate(MFRC522::PICC_CMD_MF_AUTH_KEY_B, 0, &key,
&(mfrc522.uid));
    if (status == MFRC522::STATUS_OK) {
      status = mfrc522.MIFARE_Read(1, _RFIDUIDKEY, &byteCount);
      if (status == MFRC522::STATUS_OK) {
        status = mfrc522.MIFARE_Write(1, RFIDUIDCLEAR, 16);
        if (status == MFRC522::STATUS_OK) {
          ESP.wdtFeed();
          return true;
        }
      }
      RFIDend();
      ESP.wdtFeed();
      return false;
    }
    RFIDend();
    ESP.wdtFeed();
  }
  return false;
}
```

Вкладка “Wi-Fi”:

```
void WiFiRecvIndex() {
  String message  = "<html>";
        message += "<head>";
        message += "<meta http-equiv='Content-Type' content='text/html; charset=utf-
8'>";
        message += "<title>System Control Access</title>";
        message += "<style type='text/css'>";
        message += "a {";
        message += "padding: 5px 15px;";
        message += "font-size: 12pt;";
        message += "text-align: center;";
        message += "text-decoration: none;";
        message += "color: #000;";
        message += "}";
        message += "div {";
        message += "padding: 10px;";
        message += "height: 20px;";
        message += "}";
        message += ".t {";
        message += "border: 1px solid #333;";
        message += "padding: 5px 15px;";
        message += "font-size: 12pt;";
        message += "text-align: center;";
        message += "text-decoration: none;";
        message += "color: #000;";
        message += "width: 100%;";
        message += "}";
        message += "</style>";
        message += "</head>";
        message += "<body>";

        if (Device.WiFinfo ==1) {
          message += "<div class='t'>Открыть замок - Выполненно</div><div
class='t'></div>";
        }
        if (Device.WiFinfo == 2) {
```

```
message += "<div class='t'>Приложите метку для записи мастера</div><div
class='t'></div>";
      }
      if (Device.WiFinfo == 3) {
        message += "<div class='t'>Приложите метку для записи
пользователя</div><div class='t'></div>";
      }
      if (Device.WiFinfo == 4) {
        message += "<div class='t'>Стереть отпечатки - Выполненно</div><div
class='t'></div>";
      }
      if (Device.WiFinfo == 5) {
        message += "<div class='t'>Приложите метку для удаления
данных</div><div class='t'></div>";
      }

      message += "<div class='t'><a href='/write'>Запись мастера</a></div>";
      message += "<div class='t'><a href='/writekey'>Запись ключа</a></div>";
      message += "<div class='t'></div>";
      message += "<div class='t'><a href='/open'>Открыть замок</a></div>";
      message += "<div class='t'></div>";
      message += "<div class='t'><a href='/clear'>Стереть отпечатки</a></div>";
      message += "<div class='t'><a href='/keyclear'>Стереть метку</a></div>";

      message += "</body>";
      message += "</html>";

      server.send(200, "text/html", message);

      Device.WiFinfo = 0;
}


void WiFiRecvvOpen() {
  Device.WieFinfo = 1;
  WiFieRecvIndex();
  OpenLock();
}
```

```
void WiFiRecvWrite() {
  Device.WiFinfo = 2;
  WiFiRacvIndex();
  Device.status = 2;
  Device.time = millis();
  Device.imprint.stage = 0;
  digitalWrite(LED_PIN, HIGH);
}

void WiFiRecvWriteKey() {
  Device.WieFinfo = 3;
  WiFiRecvIndex();
  Device.status = 1;
  Device.time = millis();
  Device.imprint.stage = 0;
  digitalWrite(LED_PIN, HIGH);
}

void WiFieRecvWriteClear() {
  Device.WiFinfo = 4;
  WiFiRecvIndex();
  imdell();
}

void WiFiRecvWriteKeyClear() {
  Device.WiFinfo = 5;
  WiFiRecvIndex();
  Device.status = 3;
  Device.time = millis();
  Device.imprint.stage = 0;
  digitalWrite(LED_PIN, HIGH);
}

void WiFiSetup() {
  WiFi.mode(WIFI_AP);

  WiFi.softAP("System Control Access", "89274902072");
```

```
 Serial1.println(WiFi.softAPIP()); // 192.168.4.1

  server.begin();

  server.on("/", WiFiRecvIndex);
  server.on("/write", WiFiRecvWrite);
  server.on("/writekey", WiFiRecvWriteKey);
  server.on("/open", WiFiRecvOpen);
  server.on("/clear", WiFiRecvWriteClear);
  server.on("/keyclear", WiFiRecvWriteKeyClear);
}

void WiFiLoop() {
  server.handleClient();
  MDNS.update();
}
```