

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

РАБОТА ПРОВЕРЕНА

Рецензент

_____ 2020 г.
« ___ » _____

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой ЭВМ

_____ Г.И. Радченко
_____ 2020 г.

Разработка прототипа системы связи Li-Fi для передачи данных между
мобильными устройствами Android (принимающая часть)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ

Руководитель работы,

к.т.н., доцент каф. ЭВМ

_____ И.Л. Кафтанников

« ___ » _____ 2020 г.

Автор работы,

студент группы КЭ-222

_____ А.Д. Хлызов

« ___ » _____ 2020 г.

Нормоконтроллер,

ст. преп. каф. ЭВМ

_____ С.В. Сяськов

« ___ » _____ 2020 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ
Заведующий кафедрой ЭВМ
_____ Г.И. Радченко
« ___ » _____ 2020 г.

ЗАДАНИЕ

на выпускную квалификационную работу магистра
студенту группы КЭ-222

Хлызову Александру Дмитриевичу

обучающемуся по направлению

09.04.01 «Информатика и вычислительная техника»

1. **Тема работы:** «Разработка прототипа системы связи Li-Fi для передачи данных между мобильными устройствами Android (принимающая часть).»
2. **Срок сдачи студентом законченной работы:** 1 июня 2020 г.
3. **Исходные данные к работе:** статьи, книги.
 - Wu, X. Access point selection for hybrid Li-Fi and Wi-Fi networks / Wu, X., Safari, M., Haas, H. – IEEE Transactions on Communications, 2017. – pp. 10;
 - Soltani, M.D. Modeling the random orientation of mobile devices: Measurement, analysis and LiFi Use Case / M.D. Soltani, A.A. Purwita, Z. Zeng, H. Haas, M. Safari. – IEEE Transactions on Communications, 2019. – pp. 15;
 - Haas H. A guide to wireless networking by light / H. Haas, C. Chen, D. O'Brien. – University of Edinburgh, King's Buildings, UK and University of Oxford, Parks Road, Oxford, UK, 2017;

– Петрусь, И.П. Современные методы построение коммуникационных линий в видимой части спектра электромагнитных волн / И.П. Петрусь. – УрГУПС. – С. 5.

4. Перечень подлежащих разработке вопросов:

- изучение принципов передачи данных с помощью света и научных проектов;
- анализ существующих решений приемопередатчиков для систем Li-Fi;
- формирование требований к системе;
- разработка архитектуры решения;
- разработка принципиальной электрической схемы устройства приемопередатчика;
- разработка печатной платы устройства приемопередатчика;
- выбор аппаратных и программных средств;
- разработка мобильного приложения для обмена данными между мобильным устройством и приемопередатчиком системы Li-Fi;
- оценка работоспособности разработанного прототипа программно-аппаратного комплекса для систем Li-Fi для мобильных устройств в различных условиях.

5. Дата выдачи задания: 1 декабря 2019 г.

Руководитель работы _____ /И.Л. Кафтанников /

Студент _____ /А.Д. Хлызов /

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Введение и обзор литературы	01.03.2020	
Формирование требований к системе и разработка архитектуры	01.04.2020	
Реализация основных модулей системы	01.05.2020	
Тестирование, отладка, эксперименты	15.05.2020	
Компоновка текста работы и сдача на нормоконтроль	24.05.2020	
Подготовка презентации и доклада	30.05.2020	

Руководитель работы _____ /И.Л. Кафтанников /

Студент _____ /А.Д. Хлызов /

Аннотация

А.Д. Хлызов, Разработка прототипа системы связи Li-Fi для передачи данных между мобильными устройствами Android (принимающая часть): ЮУрГУ (НИУ), ВШ ЭКН; 2020, 124 с., 43 ил., библиогр. список – 62 наим.

В ходе выпускной квалификационной работы был разработан прототип приемопередатчика системы Li-Fi для мобильных устройств – режим приема данных. Данный прототип состоит из аппаратной части для управления на микропроцессорных средствах, а также программной составляющей для управления приемом данных на мобильных устройствах. Решение основано на микроконтроллере STM32F100, отвечающего современным требованиям эффективности надежности.

Назначение физической составляющей заключается в обеспечении приема передаваемых данных с помощью световых волн и передачи полученной информации на мобильные устройства для дальнейшей работы пользователя с ней.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	8
1. ОБЗОР И АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ	11
1.1 СИСТЕМА «Trulifi» КОМПАНИИ SIGNIFY	14
1.2 СИСТЕМА «LiFiMAX» КОМПАНИИ OLEDCOMM.....	16
1.3 РЕШЕНИЯ КОМПАНИИ PURELIFI.....	17
1.4 СИСТЕМА КОМПАНИИ LUCIBEL.....	19
1.5 СИСТЕМА УНИВЕРСИТЕТА ИТМО.....	20
1.6 ВЫВОД.....	21
2. ФОРМИРОВАНИЕ ТРЕБОВАНИЙ К СИСТЕМЕ	23
2.1 ПОСТАНОВКА ЗАДАЧИ	23
2.2 АНАЛИЗ ОСНОВНЫХ ТЕХНОЛОГИЧЕСКИХ РЕШЕНИЙ.....	23
2.3 ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ	27
2.4 НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ	30
2.5 ВЫВОД	31
3 ПРОЕКТИРОВАНИЕ.....	32
3.1 ДИАГРАММА КЛАССОВ ПРИЛОЖЕНИЯ	32
3.2 СТРУКТУРНАЯ СХЕМА ПРИНИМАЮЩЕГО МОДУЛЯ.....	33
3.3 АРХИТЕКТУРА СИСТЕМЫ.....	35
3.4 ПОДБОР КОМПОНЕНТОВ.....	37
3.5 РАЗРАБОТКА ПРИНЦИПИАЛЬНОЙ СХЕМЫ АППАРАТНОЙ ЧАСТИ СИСТЕМЫ.....	47
3.6 РАЗВОДКА ПЕЧАТНОЙ ПЛАТЫ	48
3.7 ПРОЕКТИРОВАНИЕ КОРПУСА УСТРОЙСТВА	48
3.8 ВЫБОР ИНСТРУМЕНТОВ РАЗРАБОТКИ.....	50
3.9 АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧИ.....	52
3.10 ОПИСАНИЕ ДАННЫХ	56
4. РЕАЛИЗАЦИЯ.....	59
4.1 ПРИЛОЖЕНИЕ ANDROID.....	59

4.2 ПРОГРАММА МИКРОКОНТРОЛЛЕРА	76
5. ТЕСТИРОВАНИЕ	79
6. ЗАКЛЮЧЕНИЕ.....	82
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	83
ПРИЛОЖЕНИЕ А СХЕМА ЭЛЕКТРИЧЕСКАЯ ПРИНЦИПИАЛЬНАЯ	89
ПРИЛОЖЕНИЕ Б ПЕРЕЧЕНЬ ЭЛЕМЕНТОВ К СХЕМЕ ЭЛЕКТРИЧЕСКОЙ ПРИНЦИПИАЛЬНОЙ.....	90
ПРИЛОЖЕНИЕ В РАЗВЕДЕННАЯ ПЛАТА УСТРОЙСТВА.....	92
ПРИЛОЖЕНИЕ Г ФРАГМЕНТ ИСХОДНОГО КОДА ДЕШИФРОВАНИЯ.	93
ПРИЛОЖЕНИЕ Д ФРАГМЕНТ ИСХОДНОГО КОДА ПРИЕМА ДАННЫХ НА МОБИЛЬНОМ УСТРОЙСТВЕ.....	95
ПРИЛОЖЕНИЕ Е ФРАГМЕНТ ИСХОДНОГО КОДА ДЕМОДУЛЯЦИИ НА МИКРОКОНТРОЛЛЕРЕ	121

ВВЕДЕНИЕ

Информационные технологии не стоят на месте, и в настоящее время развиваются быстрыми темпами. Беспроводные технологии являются подклассом информационных технологий, служащих для передачи информации между двумя и более точками на расстоянии, при этом не требуя проводной связи. Передавать информацию возможно с помощью радиоволн, а также инфракрасного, оптического или лазерного излучения.

Существует множество беспроводных технологий, чаще всего известных под маркетинговыми названиями, такими как Wi-Fi, WiMAX, Bluetooth. Каждая из технологий обладает определенными характеристиками, которые определяют ее область применения, в данной части, мы рассмотрим беспроводную технологию Wi-Fi.

На сегодняшний день беспроводная технология доступа к Интернету Wi-Fi является неотъемлемой частью повседневной жизни. Каждый человек уже привык пользоваться ею как в домашних условиях, так и в кафе, торговых центрах, вокзалах, аэропортах, офисах и различных учреждениях [3]. Количество передаваемой информации по беспроводным сетям увеличивается с каждым днем. Соответственно это очень негативно сказывается на пропускной способности данной технологии. Установленные повсеместно Wi-Fi роутеры, работающие в достаточно узком радиочастотном диапазоне, создают многочисленные помехи, что приводит к снижению скорости двухстороннего обмена данными.

Представьте себе, что было бы возможно снизить нагрузку на пропускную способность беспроводной технологии Wi-Fi, а также снизить количество создаваемых помех и значительно увеличить скорость обмена информацией.

И это возможно сделать с помощью беспроводной технологии передачи информации Li-Fi, которая работает как Wi-Fi, но во много раз быстрее и использует не радиоволны, а видимый свет.

Частотный диапазон видимого света значительно превосходит диапазон радиоволн, что позволяет устройствам не конкурировать за пропускную способность и снизить нагрузки на Wi-Fi сети [7].

Эта технология очень хорошо работает в пределах помещения и на расстоянии около 10 метров, но передавать сигнал в другое помещение не сможет, так как свет не может проходить сквозь стены и другие препятствия, но технология Li-Fi не требует работы в пределах прямой видимости, если между источником и приемником поставить препятствие, то доступ в глобальную сеть Интернет не исчезнет: свет отразится от стен и достигнет приемника, но при этом скорость снизится.

Большинство компаний, занимающихся разработкой данной технологии совершенствуют свои устройства и программное обеспечение для использования их с персональными компьютерами, не обращая внимания на тот фактор, что большую часть времени человек проводит за использованием своего мобильного устройства, и не развивают данную технологию для области мобильных устройств.

Изучив данные проблемы, связанные с беспроводной технологией Wi-Fi, а также с развитием беспроводной технологии Li-Fi для мобильных устройств, было принято решение разработать прототип приемопередатчика системы Li-Fi для мобильных устройств, который позволит увеличить скорость обмена информацией в несколько раз и исключит создание помех, мешающих работе другого оборудования, а также снизит нагрузку на пропускную способность беспроводной технологии Wi-Fi.

Целью, представленной выпускной квалификационной работы, является разработка архитектуры и реализация опытного образца приемника системы Li-Fi для мобильных устройств, а также разработка алгоритма работы приемника, написание мобильного приложения для общения с приемником.

Для достижения поставленной цели, необходимо решить следующие поставленные задачи:

1. Изучить проблемы, возникающие в процессе передачи информации с помощью видимого света;
2. Провести обзор аналогов – модулей для систем Li-Fi, выделить недостатки рассмотренных систем и сформировать список требований к новой системе;
3. Выбрать аппаратные и программные средства для реализации системы удовлетворяющей поставленным требованиям;
4. Разработать архитектуру системы, удовлетворяющую полученным требованиям, схему электрическую принципиальную, выполнить трассировку печатной платы приемопередатчика, разработать основные алгоритмы и представить их конечную реализацию;
5. Реализовать основные компоненты системы для тестирования и демонстрации работы полученного решения;

1. ОБЗОР И АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Технология Wi-Fi обеспечивает высокоскоростную двунаправленную мобильную связь, используя радиочастотный диапазон. Но в процессе обмена информацией с помощью технологии Wi-Fi могут возникать задержки, так как на сегодняшний день существует огромное количество устройств использующих подобную технологию, что приводит к большой нагрузке и снижает пропускную способность, тем самым заставляя устройства конкурировать за доступ к сети.

К тому же существуют запреты на технологию Wi-Fi, такие как на использование в самолетах, в медицинских учреждениях, опасной промышленности или в военной промышленности, так как может мешать работе оборудования или является не слишком безопасным для передачи информации.

Проблема заключается в том, что Wi-Fi может создавать помехи в работе различного оборудования там, где это недопустимо, что может привести к человеческим жертвам [8]. Также возможны утраты информации, имеющей ценность для государств, компаний или обычных людей.

Существует несколько решений, с помощью которых возможно решить подобные проблемы. Возможно использование проводных способов связи: использование волоконно-оптического кабеля (оптоволокно), витая пара категории CAT6A и выше. Из беспроводных технологий можно использовать 5G. Основным недостатком оптоволокна является высокая стоимость его прокладки и эксплуатации [3]: необходимо иметь специальное оборудование для преобразования сигнала, а также при разрыве, волокно практически не подлежит ремонту (обычно менять приходится целый участок кабеля). По сравнению с волоконно-оптическим кабелем, кабель витой пары более тонкий, более гибкий и его проще устанавливать. Он также недорог. Однако, витая пара обладает следующими недостатками: сильное воздействие внешних электромагнитных наводок, возможность утечки информации и сильное затухание сигналов (потеря мощности сигналов). Затухание растет с увеличением частоты сигнала и удлинением кабеля передающей среды. Проводники витой пары подвержены

поверхностному эффекту - при высокой частоте тока, электрический ток вытесняется из центра проводника, что приводит к уменьшению полезной площади проводника и дополнительному ослаблению сигнала [4]. Недостатки сетей 5G: необходимость разработок новых стандартов, более мощных микропроцессоров и программного обеспечения, для плотного покрытия понадобится установить большое количество станций сотовой связи [6].

В таблице 1 ниже сведены основные характеристики существующих решений для более детального представления их возможностей.

Таблица 1 – Возможности технологий

Наименование решения	Максимальная частота	Ширина сигнала	Максимальная скорость
Оптоволокно [5]	100 ПГц	От 100 ТГц до 100 ПГц	50000 Гбит/с
Витая пара [4]	700 МГц	От 500 МГц до 700 МГц	100 Гбит/с
5G сети [6]	100 ГГц	От 24 ГГц до 100 ГГц	20 Гбит/с
Li-Fi ([7], [9])	800 ТГц	От 430 ТГц до 790 ТГц	224 Гбит/с

В данной работе было решено реализовать систему с использованием видимого света, которая сможет решить проблемы, описанные выше. Очевидно, что в рамках дипломной работы охватить их все невозможно. Поэтому данная работа будет нацелена на создание прототипа приемопередатчика Li-Fi системы для мобильных устройств, который позволит повысить скорость обмена информацией между устройствами, увеличить безопасность передаваемых данных, так как видимый свет не может проходить сквозь стены, в дополнении позволит уменьшить нагрузку на радиочастотный диапазон.

В настоящее время, мобильные устройства используют для хранения и передачи большого количества информации, поэтому организация передачи

данных с помощью Li-Fi позволит в значительной степени ускорить процесс обмена между мобильными устройствами.

Последние достижения в сфере производства светодиодов сделали их более энергоэффективными, а скорость включения-выключения стала достигать нескольких наносекунд, что позволяет использовать их также и для передачи данных.

Принцип работы системы заключается в следующем, передатчик модулирует данные для приемника с частотой невидимой человеческому глазу и посылает полученную последовательность бит на светодиод, светодиод начинает мерцать, обозначая уровни лог. 1 и лог. 0. В приемном модуле фотодетектор реагирует на мерцание диода и генерирует посылку данных состоящую из 0 и 1, которую приемник затем демодулирует в цифровой сигнал, который пересылает требуемые данные.

Для модуляции и демодуляции данных необходимо иметь заготовленную таблицу с кодами символов в памяти микроконтроллера. Также для взаимодействия смартфонов между собой необходимо написать мобильное приложение для смартфона, поскольку подавляющее большинство мобильных устройств имеют ОС (операционную систему) Android, что отражено на рисунке 1 [10], то приложение будет написано для этой операционной системы.

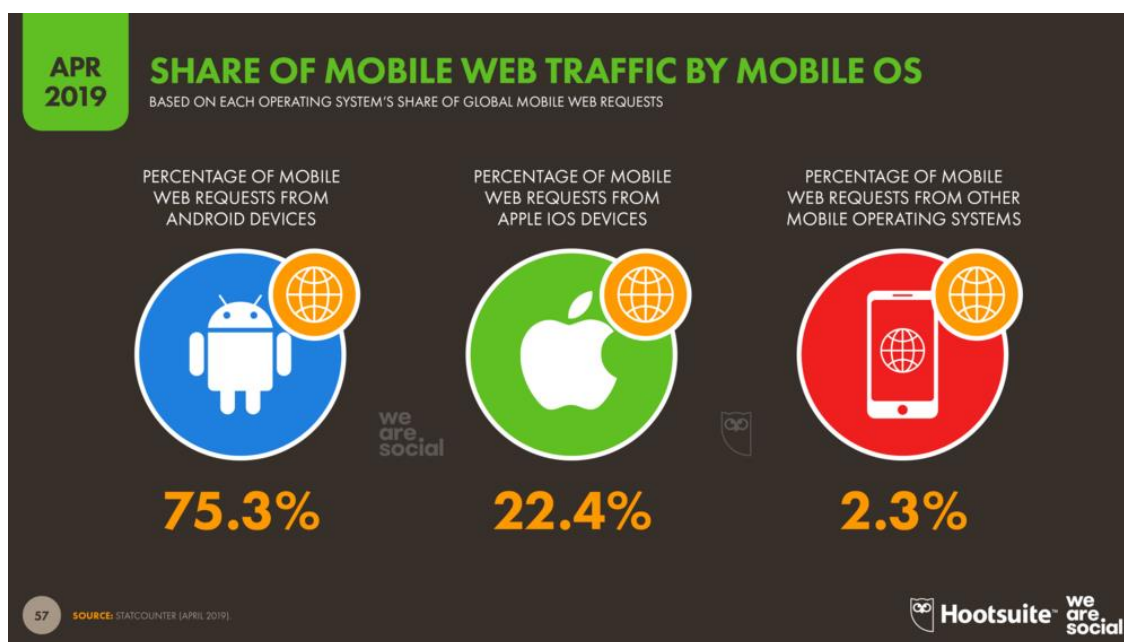


Рисунок 1 – Статистика Web-запросов с различных ОС

При проектировании системы основными компонентами являются:

1. Яркий светодиод, являющийся источником передаваемых данных;
2. Кремниевый фотодиод (фотодетектор), способный реагировать на видимый свет или другую электромагнитную энергию, который преобразует световые фотоны в ток.

Передающий модуль Li-Fi включает в себя четыре компонента:

1. Светодиод;
2. Высокочастотный усилитель цепи питания;
3. Печатная плата;
4. Корпус.

Печатная плата предназначена для соединения входов и выходов светодиода, а встроенный микроконтроллер используется для управления различными функциями светодиода. Радиочастотный сигнал порождается твердотельным высокочастотным усилителем цепи питания и преобразуется в электрическое поле вблизи светодиода. Высокая плотность энергии электрического поля доводит содержимое диода до состояния плазмы. Эта управляемая плазма и является источником света. Все эти компоненты обрамлены алюминиевым корпусом.

В приемном модуле полученный оптический сигнал усиливается, затем проходя через фильтр верхних частот, избавляется от шумов, вследствие чего попадает на компаратор для преобразования в цифровую форму в итоге на выходе принимающего модуля имеется полноценный цифровой сигнал, который передает необходимые данные. Подключение приемного модуля для смартфонов следует реализовать через разъем TRRS jack 3.5, так как он имеется на многих мобильных устройствах, дешев и его достаточно для учебных целей.

1.1 СИСТЕМА «Trulifi» КОМПАНИИ SIGNIFY

В 2019 году компании Philips представила лампы-модемы, которые используют оптическую технологию Li-Fi для беспроводного обмена данными.

Это решение получило название Trulifi, элементы решения представлены на рисунках 2-3 [13].



Рисунок 2 – Приемопередатчик Trulifi компании Signify



Рисунок 3 – точка доступа Trulifi компании Signify

Серия 6001 использует видимый свет и обеспечивает скорость обмена данными до 30 Мбит/сек. Передатчик Li-Fi монтируется в светодиодную лампу Philips и обеспечивает покрытие диаметром в 2 метра при монтаже на высоте в 2 метра и покрытие в 3 метра при монтаже на высоте 2,5 метра [22].

Серия 6002 использует инфракрасный светодиод, обеспечивая общую скорость отдачи в 150 Мбит/сек, максимальная высота потолка 2 метра [22].

Серия 6013 использует видимый и инфракрасный свет для создания соединения со скоростью до 250 Мбит/сек в обоих направлениях, высота потолка 2 метра, а ширина покрытия до 8 метров, благодаря, инфракрасному излучению соединение может быть установлено даже при выключенном свете [22].

Принимающий модуль подключается к персональным компьютерам или ноутбукам через USB интерфейс. Решение поддерживает операционные системы семейства Windows, Linux и MacOS. Стоимость на линейку продукции можно узнать, сделав предзаказ у менеджеров компании.

1.2 СИСТЕМА «LiFiMAX» КОМПАНИИ OLEDCOMM

Компания Oledcomm вдохновленная идеей передачи данных с помощью видимого света разработала свою собственную систему LiFiMAX, которая может быть легко монтирована в качестве основного освещения или же использоваться в качестве местного освещения, в данный момент компания является одним из лидеров в области световых решений [11]. На рисунках 4-5 представлены разработанные решения [15].



Рисунок 4 – Решение компании Oledcomm система LiFiMAX для местного освещения

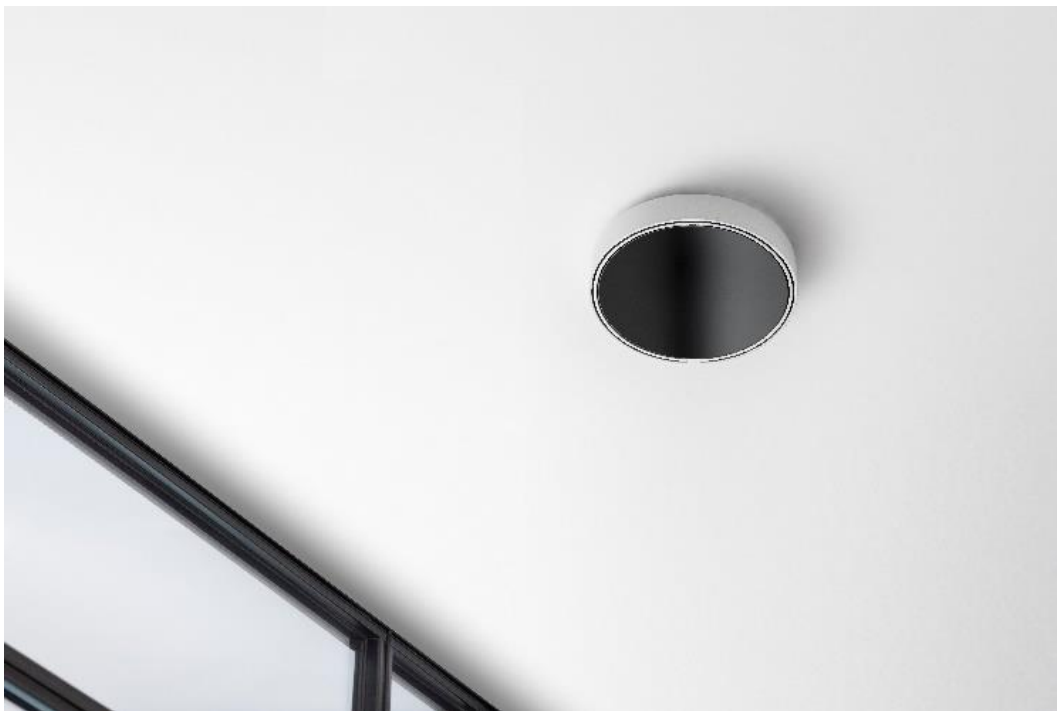


Рисунок 5 – Решение компании Oledcomm система LiFiMAX в качестве основного освещения

LiFiMAX обеспечивает быструю скорость загрузки данных до 100 Мбит/сек и скорость отправки данных до 40 Мбит/сек, что дает плавный доступ к сети между несколькими пользователями. Так же данная система может одновременно поддерживать до 16 пользователей и обеспечивать каждому из них быстрый и надежный доступ к сети Интернет [15]. Приемопередатчик подключается к устройствам посредством USB-ключа (USB-разъема), именно, благодаря этому ключу и происходит обмен данными и подключение такого количества пользователей. Питание точки доступа осуществляется посредством PoE. Система поддерживает такие операционные системы как Windows, Linux и MacOS. Решение LiFiMAX стоит 900 \$, поставки этого решения начались еще в первой половине 2019 года [16].

1.3 РЕШЕНИЯ КОМПАНИИ PURELIFI

Основателем компании является Харальд Хаас, который предложил первым передавать данные с помощью видимого света еще в 2011 году. У компании

имеется несколько решений, которые с каждым годом совершенствуются, повышая свою производительность, некоторые решения представлены на рисунках 6-8 [12].



Рисунок 6 – Встраиваемый светильник системы Li-Fi компании PureLiFi



Рисунок 7 – LiFi-XC компании PureLiFi



Рисунок 8 – Интегрированный ноутбук системой Li-Fi компании PureLiFi

Компания занимается разработкой решений для персональных компьютеров на базе операционных систем Windows, Linux и MacOS, а также мобильных устройств на базе операционной системы Android. Сейчас компания разрабатывает первую гигабитную технологию, поддерживающую Li-Fi, которая может быть встроена в мобильные устройства следующего поколения. Пропускная способность технологии составляет 1 Гбит/сек [12]. Также у компании находится в разработке первая в мире Li-Fi ASIC – это первый специально созданный кремний Li-Fi, совместимый с Wi-Fi, поддерживает полный дуплекс, до 16 пользователей на точку доступа. USB интерфейс используется для подключения к персональным компьютерам.

1.4 СИСТЕМА КОМПАНИИ LUCIBEL

Lucibel разработала самый первый осветительный прибор, оснащенный технологией Li-Fi, которая обеспечивает доступ к глобальной сети Интернет на основе светодиодного света. Li-Fi – это технология, которая позволяет осуществлять связь через модулированный светодиодный свет. На рисунках 9-10 представлены некоторые из решений компании [14].

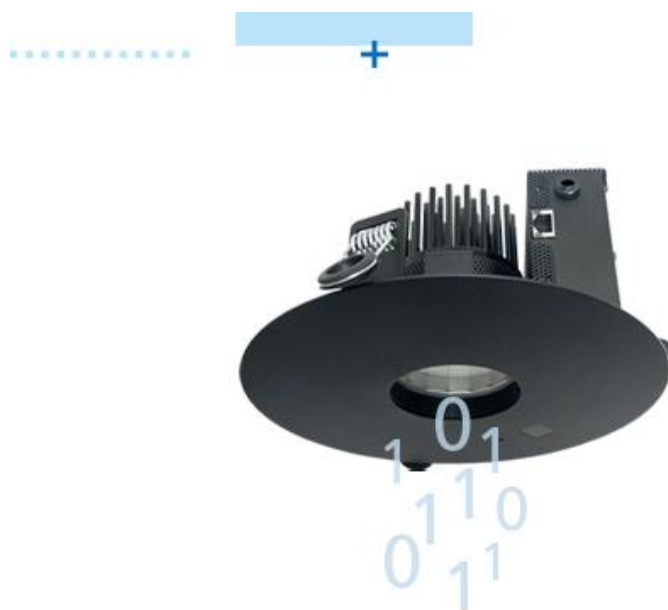


Рисунок 9 – Точка доступа ORES компании Lucibel



Рисунок 10 – Принимающий модуль LIFI USB KEY компании Lucibel

Решение поддерживает такие операционные системы как Windows, Linux и MacOS. В качестве интерфейса для подключения к персональным компьютерам и ноутбукам был выбран USB, а скорость соединения может достигать до 54 Мб/с [14]. Стоимость системы отсутствует в открытом доступе, необходимо делать предзаказ данного решения.

1.5 СИСТЕМА УНИВЕРСИТЕТА ИТМО

Первыми в России кто организовал устойчивую сеть передачи данных с помощью света был университет ИТМО [1]. В университете была запущена локальная сеть передачи данных с помощью света в новом формате известном как Li-Fi.

Так, в лаборатории университета сотрудники смогли добиться скорости в 50 мегабит в секунду, что уже сопоставимо со скоростью передачи данных с помощью технологии Wi-Fi и в некоторых моментах даже превосходит ее.

Сама система состоит из двух приемопередатчиков. Первый, состоит из модуля-модема и светодиодного модуля из 64 белых светодиодов для обычного освещения. С помощью этих светодиодов и происходит передача сигнала на фотоприемник второго модуля. Второй же модуль состоит из фотодетектора и

нескольких инфракрасных диодов, для передачи обратной информации на передающее устройство.

Оба этих модуля подключались к двум ноутбукам и располагались в зоне прямой видимости. По результатам тестирования этих устройств удалось замерить скорость передачи данных в такой сети, она составила 50 Мбит/с на расстоянии свыше трех метров между модулями.

1.6 ВЫВОД

В результате обзора выше перечисленных аналогов были выделены их функциональные составляющие и вынесены в таблицу 2.

Таблица 2 – Преимущества и недостатки имеющихся решений

Наименование решений/компаний	Trulifi [13]	LiFiMAX [11]	PureLiFi [12]	Lucibel [14]	ИТМО [2]
Скорость обмена данными	250 Мбит/с	40 Мбит/с	1 Гбит/с	-	50 Мбит/с
Интерфейс передачи данных с устройствами	USB	USB	USB	USB	USB
Поддерживаемые виды устройств	ПК и ноутбуки	ПК и ноутбуки	ПК, ноутбуки, мобильные устройства	ПК и ноутбуки	ПК и ноутбук и
Поддерживаемые ОС	Windows, Linux, MacOS	Windows, Linux, MacOS	Windows, Linux, MacOS, Android	Windows, Linux, MacOS	Windows

Продолжение таблицы 2

Наименование решений/компаний	Trulifi [13]	LiFiMAX [11]	PureLiFi [12]	Lucibel [14]	ИТМО [2]
Количество пользователей одновременно	-	16	16	-	2
Обмен данными между устройствами	+	-	-	-	+
Стоимость	Продается за территории РФ	Продается за территории РФ	Продается за территории РФ	Продается за территории РФ	-

Учитывая достоинства и недостатки имеющихся решений из таблицы 2, включить в прототип приемопередатчика системы Li-Fi:

1. Разработку приемопередатчика, который смогут использовать существующие мобильные устройства и планшетные компьютеры;
2. Поддержку операционной системы Android, так как она является наиболее распространенной операционной системой для мобильных устройств;
3. Разработку многопоточного мобильного приложения для мобильных устройств, которое позволит обмениваться данными между несколькими смартфонами;
4. Использовать интерфейс обмена данными TRRS jack 3.5;
5. Обеспечить скорость передачи до 1 Гб/с.

2. ФОРМИРОВАНИЕ ТРЕБОВАНИЙ К СИСТЕМЕ

2.1 ПОСТАНОВКА ЗАДАЧИ

Задача состоит в разработке аппаратного модуля, реализованного на микропроцессорных средствах, позволяющего принимать данные, а также из программного модуля, который позволит принимать выбранные данные и следить за процессом изменений. Данное решение позволит расширить функциональные возможности уже существующих аналогов, обеспечит защиту передаваемых данных, увеличит скорость обмена информацией, а также уменьшит нагрузку на радиочастотный диапазон.

Основные группы пользователей проекта:

1. Владельцы мобильных устройств с операционной системой Android, которые хотят повысить скорость обмена данными и их защиту;
2. Работники и посетители медицинских учреждения, пассажиры самолетов, рабочие на опасном производстве, военные и другие люди в подобных местах, где имеется запрет на использование Wi-Fi или Bluetooth;
3. Люди, заинтересованные в развитии концепта IoT.

Общее требование: графический интерфейс клиентского приложения – отображение и настройка имени устройства и ключа безопасности в мобильном приложении, получение сведений об устройстве такие как: имя устройства и ключ, используемый мобильным приложением.

2.2 АНАЛИЗ ОСНОВНЫХ ТЕХНОЛОГИЧЕСКИХ РЕШЕНИЙ

Прием данных на мобильных устройствах Android возможно реализовать 3 способами: используя фоторезистор, фотодиод или фототранзистор. Каждый из вариантов имеет свои достоинства и недостатки. Рассмотрим каждый из них:

1. Фоторезистор – тип резистора, сопротивление которого изменяется при изменении интенсивности светового потока [25]; фоторезисторы имеют

сравнительно небольшие размеры, бюджетные, а также их легко переносить из одного места в другое, основной их недостаток низкая точность;

2. Фотодиод – радиоэлемент, преобразующий свет, падающий на его чувствительную зону, в электрический разряд [26]; фотодиоды обладают большим быстродействием, но имеют большую стоимость по сравнению с фоторезисторами;

3. Фототранзистор – транзистор у которого вместо вывода базы есть в корпусе окошко для попадания света [27]; основными недостатками фототранзистора являются его нестабильность параметров, большой уровень шумов и плохие частотные характеристики.

Поскольку Li-Fi является аналогом Wi-Fi и считается дешевой технологией в производстве, поэтому при выборе основной компоненты для приемника будет учитываться его стоимость, а также быстродействие и влияние шумов. В следствие этого фототранзисторы не будут рассматриваться как один из вариантов, так как они имеют высокий уровень шумов, и нестабильность параметров, хотя у них средняя стоимость по сравнению с другими двумя аналогами. Фотодиод имеет большее быстродействие по сравнению с фоторезистором, но и большую стоимость, но так как решение должно быть бюджетным, то выбор был сделан в пользу фоторезистора. Фоторезисторы имеют быстродействие до 10 кГц, для корректного приема данных необходимо, чтобы он реагировал на мерцания источника света на частоте не менее 4 кГц, а также фоторезисторы обладают низкой стоимостью, что отлично подходит для создания устройства приемника в учебных целях.

Для подключения приемного модуля к мобильному устройству Android выбран разъем TRRS jack 3.5 [46]. Так как он есть и на ПК, и на мобильных устройствах, что при дальнейшем развитии системы будет плюсом при разработке, а также он достаточно дешев и его возможностей достаточно для реализации устройства приемопередатчика в учебных целях. В подавляющем большинстве аналогов используется разъем USB разных типов, но разъема USB A почти нет на мобильных устройствах Android, а USB C нет на ПК. Единственным минусом

использования jack 3.5 является его низкое питание на выходе в мобильных устройствах (около 2 В), его не хватит, чтоб запитать приемник, поэтому необходимы батарея или стабилизатор напряжения.

В качестве питания устройства можно использовать аккумулятор, батарею или стабилизатор напряжения. Аккумулятор имеет более большие габариты, чем батарея или стабилизатор, когда как устройство приемника предполагается быть компактным. Аккумулятор самое дорогое решение из предложенных. Батарейки нужно периодически менять. Оптимальным решением является использование стабилизатора напряжения, он выигрывает и по цене и удобству использования конечным пользователем.

Для приема информации из окружающей среды в системе используется фоторезистор, который плавно изменяет свое сопротивление, меняя тем самым напряжение на выходе. Чтобы мы могли обрабатывать полученный аналоговый сигнал, необходимо использовать АЦП в микроконтроллере. АЦП преобразовывает некоторый аналоговый сигнал, в его цифровое представление, тем самым упрощая понимание, какие данные сейчас поступают на приемное устройство. Пример вида сигнала на выходе АЦП приведен на рисунке 11 (красная ломаная). Время преобразования и точность АЦП зависит от его разрядности – количество дискретных значений, которые аналого-цифровой преобразователь может выдать на выходе [51] и частоты преобразования. Чем выше скорость преобразования, тем ниже точность получаемых дискретных значений, и наоборот, чем выше точность преобразуемых значений, тем ниже скорость обрабатываемого сигнала [52].

Передача данных в системе между принимающим модулем и мобильным устройством основана на использовании аналоговых сигналов. Для формирования аналоговых сигналов на МК используется ЦАП. ЦАП осуществляет преобразование некоторых цифровых данных, определенных в виде двоичного кода, в их аналоговое представление, тем самым позволяя плавно регулировать напряжение [23]. На рисунке 11 приведен пример сигнала на выходе ЦАП (черная кривая линия). Время преобразования ЦАПа зависит также от разрядности и

частоты преобразования, как и у АЦП: чем выше разрядность, тем точнее преобразование, но тем самым оно медленнее, и наоборот, если выше частота преобразования, тем меньше точность. На стороне мобильного устройства данные будут приниматься при помощи специального AudioRecord API.

Параметры ЦАП и АЦП могут варьироваться в зависимости от используемой модели контроллера в устройствах.

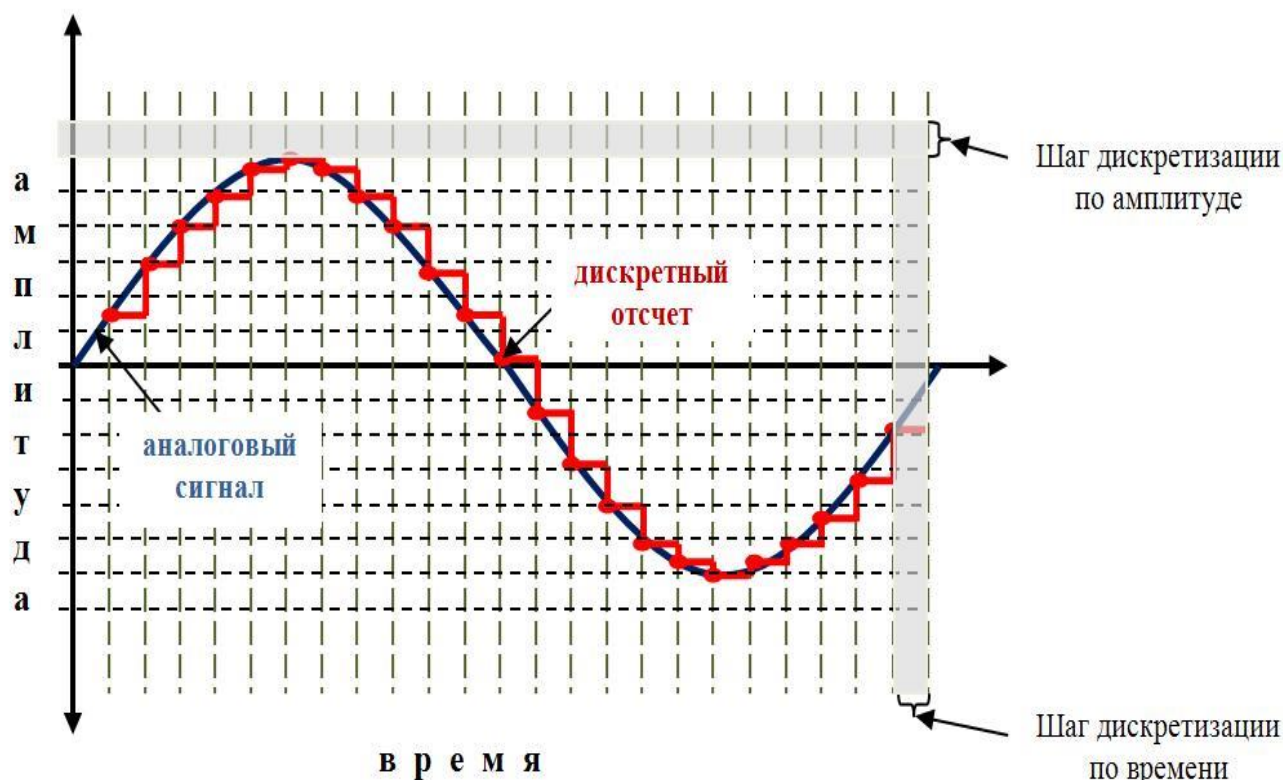


Рисунок 11 – Виды сигналов с ЦАП и АЦП

Мобильное приложение для Android может быть реализовано с поддержкой конкретной версии Android и выше. Если использовать поддержку начиная с версий 2.3.3-2.3.7 приложение будет работать на всех существующих устройствах Android, но при этом инструментарий для разработки будет сильно ограничен, так как данная версия Android имеет API 10. Если же разрабатывать для последней версии Android 9, то это тоже не имеет смысла, так как сейчас число ее пользователей мало, и приложение не будет поддерживаться более ранними версиями. На 2019 год наиболее распространенной версией ОС Android является

версия 6.0, что показано в таблице 3 [24]. Поэтому для разработки мобильного приложения минимальной версией следует выбрать именно 6.0.

Таблица 3 – Распространенность различных версий Android на 2019 год

Версия	Название	API	Распространенность
2.3.3 - 2.3.7	Gingerbread	10	0,3%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0,3%
4.1.x	Jelly Bean	16	1,2%
4.2.x		17	1,5%
4.3		18	0,5%
4.4	KitKat	19	6,9%
5.0	Lollipop	21	3,0%
5.1		22	11,5%
6.0	Marshmallow	23	16,9%
7.0	Nougat	24	11,4%
7.1		25	7,8%
8.0	Oreo	26	12,9%
8.1			15,4%
9	Pie	28	10,4%

Приложение для Android может быть многопоточным или однопоточным. Основные преимущества многопоточности – выполнение долгих операций в отдельном потоке, что позитивно сказывается на производительности, а также позволяет реализовать отзывчивый интерфейс. Ввиду преимуществ многопоточности, ее следует реализовать для мобильного приложения системы.

2.3 ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

В результате обзора аналогов в пункте 1.6 были выявлены недостатки существующих устройств, исходя из этого были учтены характеристики для разрабатываемого решения. Должны обеспечиваться следующие функциональные требования:

1. Прием и передача одного или нескольких файлов с другого устройства Android, через устройство приёмопередатчика на частоте 4 кГц;
2. Выбор режима коммуникации и необходимых файлов;
3. Настройка данных об устройстве в мобильном приложении: ввод имени устройства, задание ключа;
4. Просмотр справки в мобильном приложении;
5. Прерывание передачи/приема данных;
6. Просмотр информации об устройстве.

На рисунке 12 показана диаграмма вариантов использования системы в целом. Варианты использования системы описаны универсальным языком моделирования UML. На диаграмме отражены сценарии работы пользователей с системой, которые приводят к желаемому результату (ПС Li-Fi МУ – краткое название разрабатываемой системы).

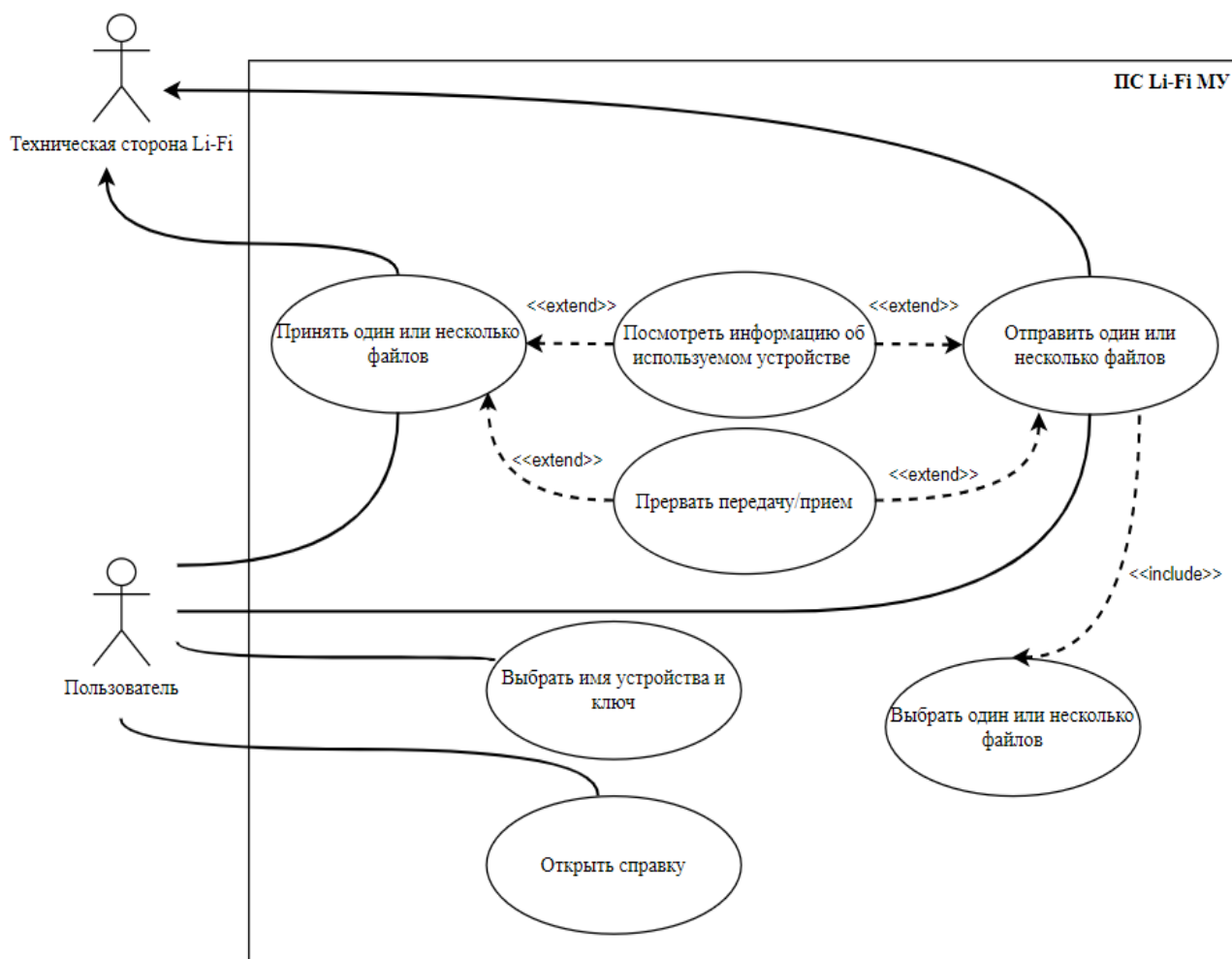


Рисунок 12 – Диаграмма прецедентов системы

В системе есть 2 действующих лица: «Пользователь» – лицо из группы пользователей системы, определенных ранее, взаимодействующих с самой системой и «Техническая сторона Li-Fi» – аппаратная часть системы. Ниже описаны основные потоки событий для каждого варианта использования:

1. Отправить один или несколько файлов: пользователь вставляет приемопередатчик в разъем jack 3.5 -> нажимает кнопку «Режим передачи» -> на следующем экране нажимает кнопку «Включить передачу» -> нажимает кнопку «Выбрать файл(ы)» -> из найденных устройств пользователь выбирает принимающее -> отправляет ему ключ безопасности с передающего устройства, установленный ранее -> если ключ совпал с ключом принимающей стороны, то -> посылки данных отправляются на передающий модуль приемопередатчика-> передающий модуль модулирует источником света посылки данных; также прецедент отправки файла(ов) включает в себя поведение прецедентов прерывания передачи, просмотра информации о подключенном устройстве и выбора файла(ов);

2. Выбрать один или несколько файлов: пользователь, находясь в режиме передачи нажимает кнопку «Выбрать файл(ы)» -> файлы преобразовываются в бинарный формат -> данные зашифровываются -> формируется посылка;

3. Принять файл(ы): пользователь вставляет приемопередатчик в разъем jack 3.5 -> нажимает кнопку «Режим приема» -> на следующем экране нажимает кнопку «Включить прием» -> на принимающее устройство отправляется ключ безопасности с передающего устройства-> если ключ совпал с ключом принимающей стороны, то начинается прием данных принимающим модулем приемопередатчика -> данные декодируются с помощью специальных методов в мобильном приложении-> данные собираются методами мобильного приложения в исходный файл; прецедент приема файла(ов) включает в себя поведение прецедентов прерывания передачи и просмотра информации о подключенном устройстве;

4. Посмотреть информацию об устройстве: находясь в режиме приема или передачи, пользователь нажимает на имя указанного устройства->

отображается имя устройства и ключ, которые были ранее указаны при настройке мобильного приложения;

5. Отменить передачу/прием: находясь в режиме приема или передачи, пользователь снимает флажок в поле switch рядом с надписью используемого режима -> если отмена со стороны передачи – передающему модулю перестает передаваться от мобильного устройства информация и передающий модуль отправляет специальный символ, который на стороне приема будет интерпретирован как конец передачи, если отмена со стороны приема – принимающий модуль отключает устройство, которое служит для приема;

6. Выбрать имя устройства и ключ: пользователь нажимает кнопку «Настройки» -> на следующем экране вводит желаемое имя устройства и ключ безопасности -> нажимает кнопку «Сохранить» -> введенные данные записываются мобильным приложением в файл, чтобы восстанавливать их при каждом запуске приложения;

7. Открыть справку: пользователь нажимает кнопку «Справка» -> на следующем экране отображается справочная информация по системе.

2.4 НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

Для всех определенных групп пользователей возможно выделить аппаратные требования и программные нефункциональные требования к системе.

К аппаратным требованиям относятся:

1. Приемный модуль должен иметь быстродействующий фоторезистор, способный реагировать на мерцание светодиода с частотой не менее 4 кГц;

2. Передатчик образует с приемником единое устройство – приемопередатчик;

3. Используемый интерфейс для подключения к мобильному устройству и обмена информацией между ними TRRS jack 3.5;

4. Приемопередатчик должен быть оснащен высокопроизводительным и энергоэффективным микроконтроллером;

5. Конструктивно устройство должно иметь сравнительно небольшие размеры;

6. Должно обеспечиваться быстрое действие системы и скорость передачи данных до 1 Гб/с;

7. Приемопередатчик должен быть эргономичным.

К программным нефункциональным требованиям относятся:

1. Поддержка операционной системы Android версии 6.0 и выше;

2. При коммуникации данные должны шифроваться;

3. Устройства должны обмениваться данными только после ввода одинакового защитного ключа принимающей и передающей стороной;

4. Пользовательский интерфейс мобильного приложения должен быть эргономичным;

5. Должна быть поддержка вертикальной и горизонтальной ориентации экрана у мобильного приложения.

Со стороны 3 группы пользователей, кроме выше описанных требований, включается требование по грамотной реализации проекта, чтобы была возможность развивать и дополнять систему в будущем.

2.5 ВЫВОД

Для указанных групп пользователей определены общие требования, основные функциональные требования, которые отражают действия выполнимые системой и нефункциональные требования, которые определяют свойства либо ограничения, демонстрируемые системой.

3 ПРОЕКТИРОВАНИЕ

3.1 ДИАГРАММА КЛАССОВ ПРИЛОЖЕНИЯ

Диаграмма классов – это набор, статических, декларативных элементов модели. Диаграммы классов применяются при прямом проектировании, то есть в процессе разработки новой системы или при обратном проектировании для описания существующих и используемых систем. На основе диаграммы использования, представленной на рисунке 12, спроектирована диаграмма классов для мобильного приложения разрабатываемой системы (см. рисунок 13). Диаграмма классов описана языком UML. Взаимосвязи объектов классов на диаграмме:

1. Ассоциация – показывает, что объекты одной сущности (класса) связаны с объектами другой сущности, таким образом, что можно перемещаться от объектов одного класса к другому; классы «Меню» и «Передача», «Меню» и «Прием», «Меню» и «Настройка», «Меню» и «Справка»;

2. Агрегация – особая разновидность ассоциации, представляющая структурную связь целого с его частями; классы «Прием» и «Информация об используемом устройстве», «Передача» и «Информация об используемом устройстве»; в этом случае при подключении некоторого одного устройства к другому они начинают обмен собственной информацией;

3. Композиция – это форма агрегации с четко выраженными отношениями владения и совпадением времени жизни частей и целого. Композиция имеет жесткую зависимость времени существования экземпляров класса контейнера и экземпляров содержащих классов. Если контейнер будет уничтожен, то все его содержимое будет также уничтожено; классы «Настройка» и «Информация об используемом устройстве»; если пользователь мобильного приложения удалит данные о своем устройстве, то

объект класса «Информация об используемом устройстве», хранящий аналогичные данные, также будет уничтожен.

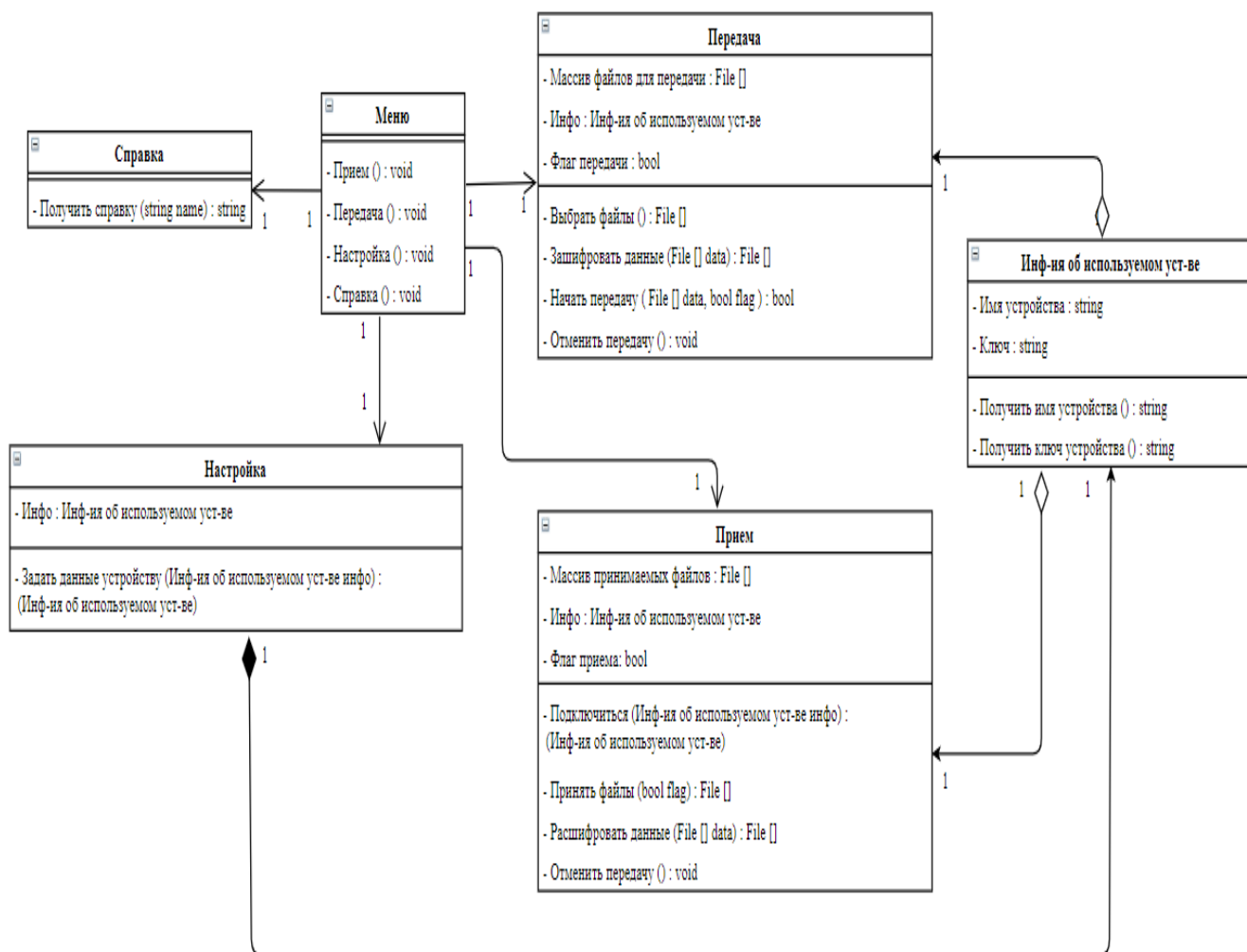


Рисунок 13 – Диаграмма классов мобильного приложения ПС Li-Fi МУ

3.2 СТРУКТУРНАЯ СХЕМА ПРИНИМАЮЩЕГО МОДУЛЯ

Аппаратное обеспечение состоит из различных электронных элементов. Для того чтобы их правильно подобрать необходимо выделить основные функциональные модули аппаратного комплекса. В его состав входят следующие модули:

1. Система управления;
2. Фоторезистор;
3. Питание информационной части;
4. Обмен информацией с системой;
5. Источник светового сигнала.

Каждая из составляющих обладает определенным функционалом, благодаря которому аппаратный комплекс имеет возможность действовать как единый механизм.

Поскольку целью работы является разработка принимающего модуля, то в него входят все перечисленные модули, кроме источника светового сигнала. На рисунке 14 представлена структурная схема, которая наглядно демонстрирует процессы, протекающие в комплексе в целом.



Рисунок 14 – Структурная схема принимающего модуля

3.3 АРХИТЕКТУРА СИСТЕМЫ

Единая аппаратная часть системы состоит из следующих компонентов: 1) источник питания, 2) микроконтроллер, содержащий в себе принимающий и передающий модули устройства, 3) источник светового сигнала, 4) фотодетектор, реагирующий на изменение светового потока, 5) интерфейс для обмена данными с мобильным устройством. Программная часть состоит из таких компонентов, как: 1) ядро мобильного приложения, 2) графическое ядро – UI, 3) API для работы с аудиоразъемом jack 3.5, 4) файл в котором хранятся имя устройства пользователя и ключ безопасности. Архитектура программно-аппаратного комплекса представлена на рисунке 15.

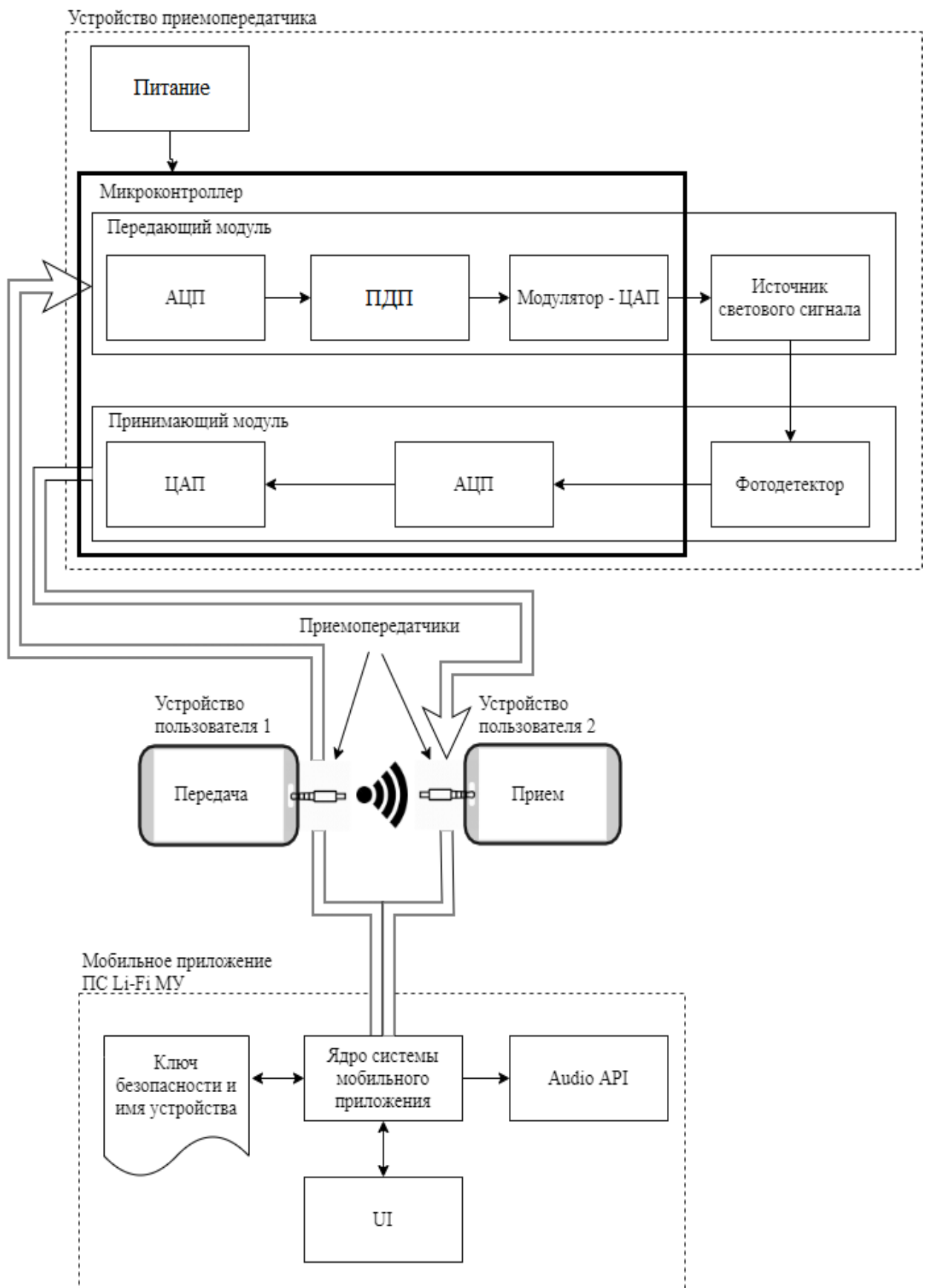


Рисунок 15 – Архитектура системы

3.4 ПОДБОР КОМПОНЕНТОВ

Добавление интеллектуальных функций в современную радиоэлектронную аппаратуру базируется на использовании микропроцессоров (МП), программируемых логических интегральных схем, приборов типа «система на кристалле» и других современных цифровых устройств [36]. Подобная интеграция позволяет автоматизировать процессы измерения, управления, контроля, регулирования и обработки информации, а также обеспечить такие свойства приборных комплексов, как многофункциональность, модифицируемость, адаптивность, обучаемость и ряд других. В отличие от персональных компьютеров, вычислительным ядром которых служат универсальные высокопроизводительные микропроцессоры, для интеграции в различные устройства применяют микроконтроллеры (МК). Благодаря своим большим возможностям микроконтроллер является одним из ключевых элементов разрабатываемой аппаратной части системы.

Li-Fi является высокоскоростной двунаправленной технологией передачи данных, поэтому для корректной работы требуется высокая скорость реагирования приемопередатчика, поэтому необходим высокопроизводительный МК. Кроме того, Li-Fi считается дешевой технологией по сравнению с такими технологиями как Wi-Fi, Bluetooth, поэтому при выборе МК будет учитываться его стоимость, чтобы минимизировать затраты, но сохранить надежность работы устройства и обеспечить достаточную производительность. В ходе поиска в глобальной сети Интернет было найдено множество вариантов высокопроизводительных МК, в рамках работы провести сравнительных анализ для всех не представляется возможным, поэтому для анализа выбраны наиболее распространенные модели:

1. 8-разрядные микроконтроллеры AtTiny13 производства Atmel;
2. 8-разрядные микроконтроллеры ATmega48 производства Atmel;
3. 32-разрядные микроконтроллеры серии ATSAM E70 фирмы Atmel;
4. Высокопроизводительные микроконтроллеры PIC16C7 фирмы Microchip;

5. 32-битные микроконтроллеры Infineon XMC1100 и XMC4100;
6. Семейство 32-разрядных микроконтроллеров STM32F100 производства компании STMicroelectronics;
7. Энергосберегающие 32-разрядные микроконтроллеры EFM32GG компании Silicon Labs.

В таблице 4 приведен их сравнительный анализ по критериям: частота микроконтроллера в МГц, которая является критерием производительности, наборам необходимой периферии – наличие ЦАП и АЦП, ПДП, наличию низкого потребления энергии и цене.

Таблица 4 – Сравнительный анализ микроконтроллеров

МК	Частота, МГц	Число каналов ЦАП	Число каналов АЦП	ПДП	Низкое потребление энергии	Цена, руб.
AtTiny13 [30]	20	-	4	-	+	от 35
ATmega48 [31]	16	-	8	-	+	от 50
ATSAM E70 [32]	до 300	2	24	+	-	от 780
PIC16C7 [33]	24	-	5	-	-	от 211
XMC1100 [34]	32	-	6-12 в зависимости от модели	-	-	от 145
XMC4100 [35]	80	2	суммарно 9	+	-	от 275
STM32F100 [36]	24	суммарно 4	10-16 в зависимости от модели	+	+	от 160

Продолжение таблицы 4

МК	Частота, Мгц	Число каналов ЦАП	Число каналов АЦП	ПДП	Низкое потребление энергии	Цена, руб.
EFM32GG [37]	48	1	7	+	+	от 423

Как видно из таблицы 4, для данного проекта по всем показателям подходят только микроконтроллеры из семейства STM32F100. Так как STM32F100 имеют низкую стоимость и весьма быстродействующие, то для реализации приемника выбран STM32F100RBT6 (см. рисунок 16) по цене 160 рублей. У данного микроконтроллера ЦАП имеет следующие характеристики, влияющие на быстродействие [36]:

1. Время пробуждения из выключенного состояния – 6,5-10 мкс;
2. Время установления (между самым высоким и самым низким значением) – 3-4 мкс.

На работу АЦП у данного типа микроконтроллера влияют характеристики, представленные ниже [36]:

1. Время пробуждения из выключенного состояния – 1 мкс;
2. Время калибровки АЦП – 83 мкс;
3. Время преобразования по инжекторным каналам – 3 мкс;
4. Время выборки – 239,5 мкс.



Рисунок 16 – Микроконтроллер STM32F100RBT6

Для тактирования микроконтроллеров могут применяться внешний высокочастотный генератор (HSE) и внутренний высокочастотный генератор (HSI) [43]. HSI является RC цепочкой с частотой 8 МГц. Точность HSI значительно ниже в отличие от внешнего генератора HSE, так как они являются температурозависимыми. Источником тактирования для HSE служит внешний тактовый сигнал, который может располагаться в диапазоне от 4 до 32 МГц. Это может быть кварцевый резонатор, тактовый генератор и т.д. HSE отличается от HSI высокой температурной стабильностью, точностью частоты и долговечностью. При выборе HSI это может крайне негативно сказаться на обработке сигналов, получаемых от фотодетектора, и исходя из этого, для тактирования МК был выбран HSE на 8 МГц, так как он указан в даташите устройства. Была куплена модель HSE – 8,000 МГц (усечен.) HC-49S (US), Кварцевый резонатор с параметрами [38]:

1. Резонансная частота – 8 МГц;
2. Нагрузочная емкость – 32 пФ;
3. Рабочая температура – $-20\dots70$ °С;
4. Длина корпуса – 11,05 мм;
5. Диаметр (ширина) корпуса – 4,65 мм;
6. Вес – 1 г;
7. Цена – 15 р.

Для приема светового потока и дальнейшей работы с полученными данными необходим фотодетектор, который является еще одним основным элементом в устройстве приемника. В главе 2 для достижения цели работы было определено использование фоторезистора. Для качественного приема данных и дальнейшей работы с ними был выбран фоторезистор GL5537. Его стоимость – 4 рубля, что очень бюджетно, по сравнению с остальными фотоэлементами. GL5537 имеет следующие параметры [39]:

1. Максимальное напряжение $U = 150$ В;
2. Максимальная потребляемая мощность $P = 100$ мВт;
3. Температура окружающей среды: $-30\dots+70$ °С;
4. Спектральный пик $P_k = 540$ нм;

5. Сопротивление при освещении $R_{осв} = 20-45$ кОм;
6. Сопротивление в темноте $R_{темн} = 2$ Мом;
7. Время отклика: повышение – 20 мс, понижение – 30 мс;
8. Вес – 1 г.

Изображение фоторезистора представлено на рисунке 17.



Рисунок 17 – Фоторезистор GL5537

Источником питания устройства приемника в главе 2 был выбран стабилизатор напряжения. Чтобы запитать микроконтроллер STM32F100RBT6 требуется от 3,3 В, с разъема TRRS jack 3.5 может выходить от ~ 1,8 до 5 В, в зависимости от того к чему подключено устройство к смартфону или персональному компьютеру, следовательно, необходим стабилизатор напряжения с выходным напряжением 3,3 В и питанием от 1,8 В. Кроме того, для работы фоторезистора требуется 5 В, поэтому необходим второй стабилизатор с выходным напряжением 5 В и питанием от 3,3 В – для формирования делителя напряжения. Из-за малых входных напряжений стабилизаторы будут иметь более высокую стоимость, чем остальные. Выбор стабилизаторов был остановлен на типе LDO – линейных стабилизаторах с малым падением напряжения. Данные стабилизаторы имеют небольшие размеры, низкий уровень шумов на выходе и высокую точность выходного напряжения [40]. Для выбора стабилизаторов использовалась

информация из рисунка 18 [41]. Были выбраны следующие фиксированные стабилизаторы с наименьшими габаритными размерами и стоимостью:

1. LD39015M33R с основными параметрами [38]:

$$U_{\text{inputMin}} = 1,5 \text{ В};$$

$$U_{\text{inputMax}} = 5,5 \text{ В};$$

$$U_{\text{output}} = 3,3 \text{ В};$$

габариты – 3x1,75x1,3 мм;

вес – 0,05 г;

цена – 43 рубля.

2. STLQ50C50R с основными параметрами [41]:

$$U_{\text{inputMin}} = 2,3 \text{ В};$$

$$U_{\text{inputMax}} = 12 \text{ В};$$

$$U_{\text{output}} = 5 \text{ В};$$

габариты – 2,2x1,35x1 мм;

вес – 0,013 г;

цена – 76 рублей.

Наименование	Входное напряжение, В	Выходное напряжение, В	Вьск. ток, мА	Падение напряжения ¹ , мВ	Потреб. ток (min), мкА	SVR ² , дБ	Шум на выходе ³ , мкBRMS/В	Enable /Power Good	Рекомендуемые характеристики вых. конденсатора		Корпус
									Емкость, мкФ	ESR, Ом	
LD3985	2,5...6	1,22; 1,8; 2,5; 2,6; 2,7; 2,8; 2,9; 3,0; 3,3; 4,7	150	0,4...60	85	50	30	+/-	1...22	0,005...5	SOT23-5L, TSOT23-5L, CSP (1,57×1,22 мм)
LDS3985	2,5...6	1,5; 1,8; 2,5; 2,8; 3,0; 3,3; 5,0	300	0,4...150	85	50	30	+/-	2,2...22	0,005...5	SOT23-5L, DFN6 (3×3 мм)
LD39015	1,5...5,5	0,8; 1,0; 1,2; 1,25; 1,5; 1,8; 2,5; 3,3	150	до 80	18	62	29	+/-	0,33...22	0,15...2	SOT23-5L, SOT666, CSP (1,1×1,1 мм)
LDS9015	2,4...5,5	0,8; 1,2; 1,5; 1,8; 2,5; 3,0; 3,3	150	до 150	31	76	20	+/-	0,33...22	0,05...8	SOT323-5L
LD39020	1,5...5,5	0,8...5,0	200	до 200	20	65	45	+/-	0,22...22	0,05...0,9	DFN4 (1×1 мм)
LD39115J	1,5...5,5	1,0; 1,2; 1,4; 1,5; 1,8; 2,5; 2,8; 3,0; 3,3	150	80 (100 мА)	20	67	30	+/-	1...22	0,1...1,8	CSP4 (0,8×0,8 мм)
LD3913054	1,5...5,5	1,0; 1,2; 1,8; 2,5; 2,9; 3,0; 3,3; 4,1; Adj	300	до 300	55 (1)	65 (48)	38 (100)	+/-	0,33...22	0,1...4	CSP4 (0,69×0,69 мм)/DFN6 (1,2×1,3 мм)
LD39050	1,5...5,5	2,5; 3,3; Adj	500	до 200	20	62	30	+/+	1...22	0,05...0,8	DFN6 (3×3 мм)
LD39100	1,5...5,5	1,2; 2,5; 3,3; Adj	1000	до 200	20	65	85	+/+	1...22	0,05...0,15	DFN6 (3×3 мм)
LD39200	1,25...6,0	3,3; Adj	2000	до 135	100	50	24	+/+	1...22	0,05...1,2	DFN6 (3×3 мм), DFN8 (4×4 мм)
LDK120	1,9...5,5	0,8; 1,0; 1,1; 1,2; 1,5; 1,8; 2,5; 2,8; 2,9; 3,0; 3,1; 3,2; 3,3; 3,5; Adj	200	до 150	30	55	51	+/-	1...22	0...10	SOT23-5L, SOT323-5L, DFN6 (1,2×1,3 мм)
LDK130	1,9...5,5	0,8; 1,1; 1,2; 1,5; 1,8; 2,5; 2,9; 3,0; 3,2; 3,3; Adj	300	до 200	30	55	51	+/-	1...22	0...10	SOT23-5L, SOT323-5L, DFN6 (1,2×1,3 мм)
LDK220	2,5...13,2	1,2...1,8; 2,5...3,3; 3,6; 4,0; 4,2; 5,0; 6,0; 8,5; 9,0; Adj	200	до 200	40	45	20	+/-	1...22	0,05...0,9	SOT23-5L, SOT323-5L, DFN6 (1,2×1,3 мм)
LDLN015	2,1...5,5	1,0; 1,2; 1,5; 1,8; 2,5; 2,8; 3,0; 3,3	150	до 86	17	89	6,3...9,9	+/-	0,33...10	0,05...0,6	DFN6 (2×2 мм)
LDCL015	1,8...5,5	3,3; Adj	150	до 70	120	51	40	+/-	Любая	Любой	SOT23-5L
STLQ50	2,3...12	1,8; 2,5; 3,3; 5,0; Adj	50	до 350	3	30	560	-/-	0,22...4,7	0...10	SOT323-5L
STLQ015	1,5...5,5	1,2; 1,5; 1,8; 2,5; 2,8; 3,0; 3,1; 3,3	150	до 112	1	30	75	+/-	0,47...10	0,056...6	SOT666
ST715	2,5...24	2,5; 3,3; Adj	85	до 500	4,15	45	95	-/-	0,47...1	0...1,5	SOT23-5L, SOT323-5L, DFN8 (3×3 мм)

Рисунок 18 – Основные электрические характеристики LDO-стабилизаторов ST

Также важно правильно подобрать конденсаторы для стабилизаторов напряжения и кварцевого резонатора, которые будут выступать в роли фильтров от помех. Для данной системы наиболее подходящими конденсаторами будут являться керамические конденсаторы. Они отличаются низким последовательным сопротивлением (ESR) и минимальной погрешностью номинала по сравнению с электролитическими или танталовыми конденсаторами, являются неполярными – можно применять в схемах с переменным током, имеют низкую цену по сравнению со слюдяными конденсаторами, обладают миниатюрным размером [42]. Был выбран кер. чип. Конденсатор 0402 X5R 2,2мкФ 2,5В 20%, GRM155C80E225ME15D в количестве по 2 штуке для каждого стабилизатора на каждый вывод, с параметрами [38]:

1. Номинальная емкость – 2,2 мкФ;
2. Рабочее напряжение – 2,5 В;
3. Габариты – 1,0x0,5x0,55 мм;
4. Цена – 1,20 руб. за 1 шт;
5. Вес – 0,65 мг.

Для кварцевого резонатора выбран кер. чип. конденсатор 1206 X6S 22мкФ 4В 10%, GRM31CC80G226KE19L в количестве 2 штук для каждого вывода, с параметрами [38]:

1. Номинальная емкость – 22 мкФ;
2. Рабочее напряжение – 4 В;
3. Габариты – 3,2x1,6x1,5 мм;
4. Цена – 4 руб. за 1 шт;
5. Вес – 0,95 мг.

Для того чтобы ограничивать ток в цепи и формировать нужное напряжение в схеме с делителем понадобится резистор [28]. Был выбран маломощный SMD резистор с мощностью до 1 Вт и с номинальным сопротивлением 470 Ом, так как он значительно меньше по размеру, чем их традиционный аналог, а для сборки компактного приемопередатчика необходимы небольшие размеры:

1. 0,062Вт 0402 470 Ом, 1%, Чип резистор (SMD)[38].

Выбранный резистор обладает следующими характеристиками:

1. Длина корпуса – 1 мм;
2. Ширина (диаметр) корпуса – 0,5 мм;
3. Вес – 0,01 г;
4. Цена – 0,9 рубля.

Устройство приемника подключается к смартфону пользователя через аудио разъем TRRS jack 3.5 [46]. Необходим штекер TRRS jack 3.5 для приемника. Данный штекер может быть моно или стерео. Для разрабатываемой системы подойдет лишь стерео с 4 контактами [46]: по левому каналу будут приходить данные из смартфона на АЦП МК, через правый – смартфон будет принимать полученные данные с ЦАП МК, а через линию микрофона напряжение питания будет подаваться на стабилизатор, который запитает устройство, и общий вывод будет заземлен. Изображение распиновки штекера TRRS jack 3.5 представлено на рисунке 19 [45]. Был выбран штекер с наименьшей стоимостью из представленных [44]:

1. Штекер TRRS jack 3.5 стерео, пластик 4 контакта, СТИА;
2. 4 контакта;
3. Цена – 40 рублей.

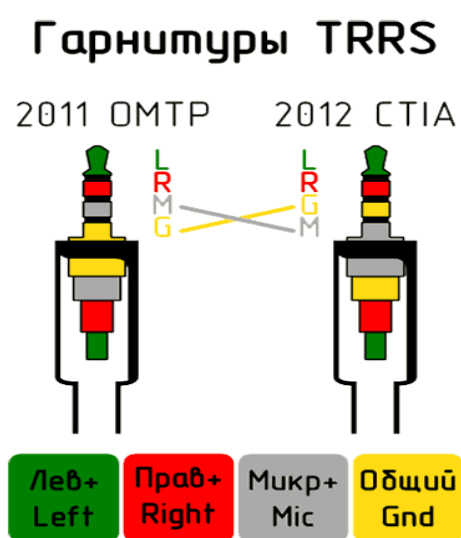


Рисунок 19 – Распиновка TRRS jack 3.5

Финансовые затраты на разработку аппаратной части устройства приведены в таблице 5, в список затрат не входит стоимость изготовления корпуса устройства и печатной платы.

Таблица 5 – Затраты на аппаратное обеспечение

Обозначение на схеме	Устройство	Номинал	Цена 1 товара, руб.	Кол-во	Итоговая цена
U1	МК STM32	STM32F100RBT6 В	160,00	1	160,00
LDR1	Фоторезистор	GL5537	4,00	1	4,00
R3	Резистор SMD	0,062 Вт 0402 22 Ом, 1%	0,9	1	0,9
U3	Стабилизатор напряжения	LD39015M33R, $U_{input} = 1,5 \dots 5,5$ В, $U_{output} = 3,3$ В	43,00	1	43,00
U2	Стабилизатор напряжения	STLQ50C50R, $U_{input} = 2,3 \dots 12$ В, $U_{output} = 5$ В,	76,00	1	76,00
C3 – C6	Керамический SMD конденсатор	0402 X5R 2,2мкФ 2,5В 20%, GRM155C80E225 ME15D	1,20	4	4,80
C1 - C2	Керамический SMD конденсатор	1206 X6S 22мкФ 4В 10%, GRM31CC80G226 KE19L	4,00	2	8,00
X1	Кварцевый резонатор	НС-49S, 8 МГц	15,00	1	15,00

Продолжение таблицы 5

Обозначение на схеме	Устройство	Номинал	Цена 1 товара, руб.	Кол-во	Итоговая цена
JACK1	Аудио штекер	- 3.5мм, стерео, 4 контакта, СТИА	40,00	1	40,00
Итоговая стоимость без учета корпуса, руб.					351,7

3.5 РАЗРАБОТКА ПРИНЦИПИАЛЬНОЙ СХЕМЫ АППАРАТНОЙ ЧАСТИ СИСТЕМЫ

После выбора компонентов для принимающего модуля была спроектирована принципиальная электрическая схема устройства для всей аппаратной части в целом. Она представлена на рисунке А.1 приложения А. Спецификация всех компонентов в схеме представлена в приложении Б.

Для разработанной принципиальной схемы необходимо рассчитать максимальную потребляемую мощность, для этого необходимо рассчитать максимальную потребляемую мощность каждого элемента и просуммировать их все. Потребляемые мощности элементов можно узнать из их даташита, либо используя формулу (1) для расчета мощностей всех элементов [55], кроме конденсаторов и формулу (2) для расчета мощностей конденсаторов [56]:

$$P = I * U, \quad (1)$$

$$P = \frac{W}{t} = \frac{C * U^2}{2 * t}, \quad (2)$$

где P – максимальная потребляемая мощность элемента, I – сила тока, U – напряжение питания, C – емкость конденсатора, W – энергия заряженного конденсатора, t – единица времени.

В результате расчетов ниже представлены значения мощностей для всех элементов схемы:

1. Транзистор КТ645Б – 0,5 Вт;
2. Фоторезистор GL5537 – 0,1 Вт;

3. Светодиод TDS-P001L4U15 STAR – 1 Вт;
4. Резисторы SMD 0402 – 0,062 Вт;
5. Стабилизатор LD39015M33R – 0,099 мВт;
6. Стабилизатор STLQ50C50R – 0,0069 мВт;
7. Кварцевый резонатор HC-49S – 0,001 Вт;
8. Микроконтроллер STM32F100RBT6B – 540 мВт;
9. Конденсатор GRM155C80E225ME15D – 6,875 мкВт;
10. Конденсатор GRM31CC80G226KE19L – 176 мкВт.

Соответственно максимальная потребляемая мощность платы составляет 2,33 Вт.

3.6 РАЗВОДКА ПЕЧАТНОЙ ПЛАТЫ

Для того чтобы сделать устройство более компактным по размерам, была разработана двухслойная плата, кроме того двухслойные платы отличаются более высокой плотностью по сравнению с односторонними, что придаст устройству дополнительную надежность. Компоненты расположены таким образом, чтобы плата имела минимальные размеры и исключало неудобства в использовании мобильного устройства совместно с устройством приемопередатчика.

Разводка печатной платы выполнена в САПР PROTEUS ARES и представлена на рисунке В.1 приложения В. Трассировка печатной платы представлена в приложении В. Плата имеет размеры 47x50 мм, минимальная ширина проводника 0,25 мм, что соответствует 3 классу точности печатной платы [47]. Для изготовления платы был выбран стеклотекстолит двухсторонний Rexant СФ1.5-2-0.35 09-4021, толщиной 1,5 мм, размерами 10x10 мм и стоимостью 87 рублей [48].

3.7 ПРОЕКТИРОВАНИЕ КОРПУСА УСТРОЙСТВА

Устройство должно иметь корпус, отвечающий функционалу и конструктивным особенностям. Поскольку приемопередатчик должен

подключаться к мобильному устройству через интерфейс TRRS jack 3.5, то в корпусе должно быть отверстие, позволяющее надежно закрепить данный разъем и обеспечить надежное соединение со входным отверстием мобильного телефона.

Также устройство должно иметь на противоположной стороне от отверстия для интерфейса TRRS jack 3.5 два отверстия для расположения источника светового сигнала и фотодетектора, которые отвечают за передачу и прием информации. Эти отверстия должны быть защищены от попадания влаги или других механических повреждений прозрачным стеклом, которое не будет влиять на качество передаваемых и принимаемых данных.

Сам корпус должен быть выполнен из черного непрозрачного материала для минимизации помех из внешнего окружения и не должен создавать проблем в использовании мобильного устройства или самого устройства приемопередатчика. Также материал должен быть достаточно крепким, чтобы выдерживать, например, падение устройства с небольшой высоты. Материалы используемые для изготовления корпуса должны быть эргономичны. Корпус имеет габариты 50x53x17,8 мм. На рисунках 20-21 представлена модель полученного корпуса в САПР SolidWorks.

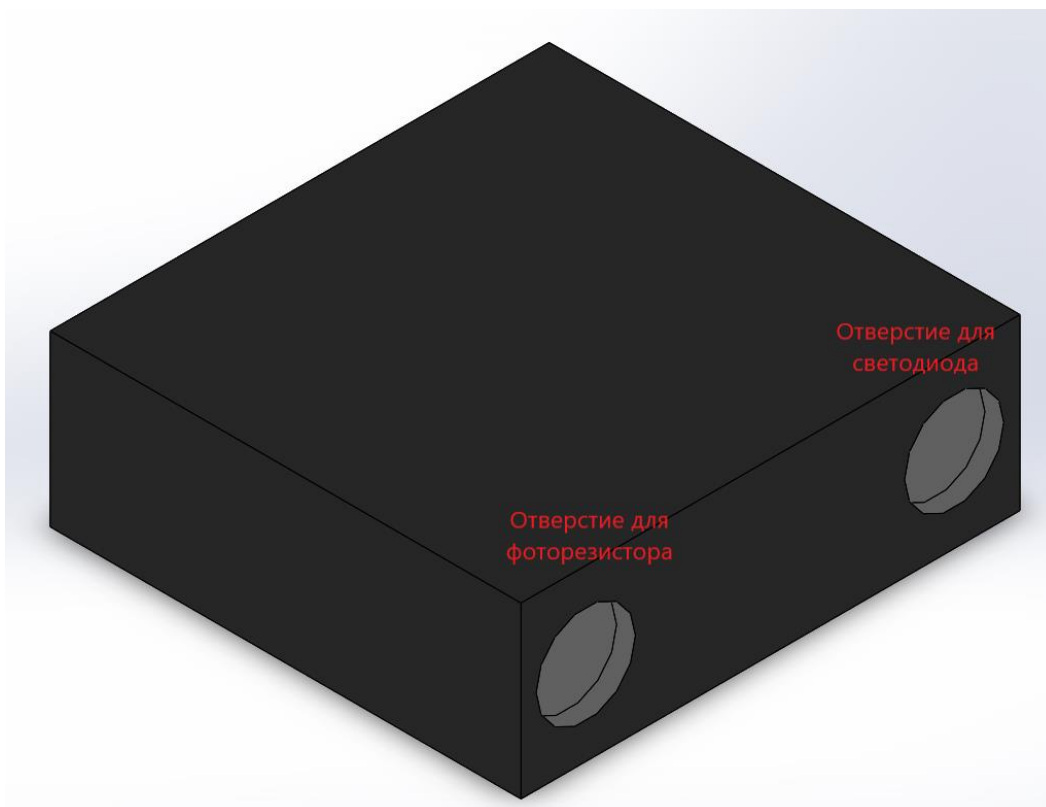


Рисунок 20 – Внешний вид корпуса с внешней стороны устройства

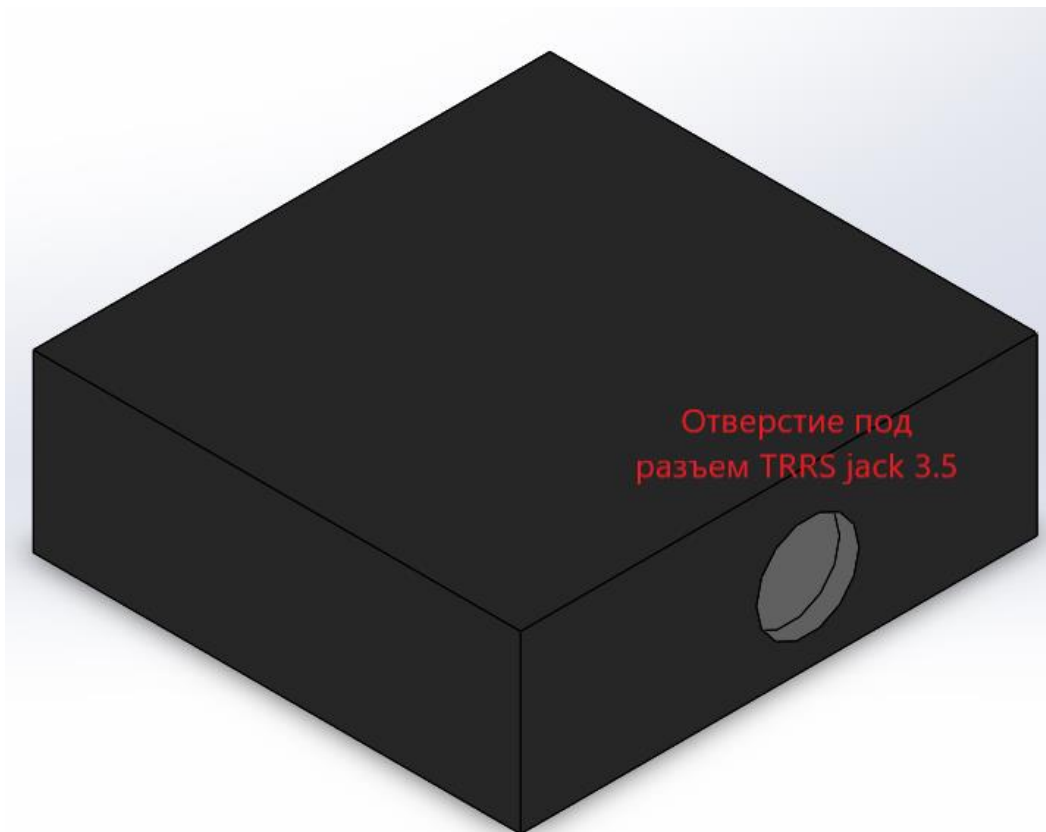


Рисунок 21 – Внешний вид корпуса со стороны разъема TRRS jack

3.8 ВЫБОР ИНСТРУМЕНТОВ РАЗРАБОТКИ

Для реализации аппаратной части системы необходимо определиться с инструментами для программирования МК STM32. Для программирования данного микроконтроллера в настоящее время существует множество IDE как платных (IAR, CrossWorks, SOMNIUM DRT Cortex-M и т.д), так и бесплатных (Atollic TrueStudio, CoCoX, STM32Cube). В связи с ограничением средств коммерческие IDE не рассматриваются для использования в данной работе. Бесплатные имеют практически одинаковый функционал, за исключением, например, STM32Cube, который имеет встроенный кодогенератор, поэтому выбор той или иной IDE зависит от предпочтений разработчика. Для программирования устройства передатчика системы Li-Fi была выбрана CoCoX IDE (CoIDE) совместно с компилятором C. Основные преимущества CoIDE это [49]:

1. Наличие отладчика;

2. Возможно программирование на более высоком уровне без прямого указания регистров;
3. Поддерживает все устройства STM32, в основе которых лежат ядра ARM Cotex-M0, M0+, M3 и M4;
4. Не имеет ограничений на размеры кода;
5. Поддерживаются все необходимые для STM32 конфигурационные файлы и прошивки также, как и интуитивное аппаратное обеспечение, такое как платы STM32 Nucleo или комплекты Discovery Kit, используемые для отладки и проектирования;
6. IDE основана на Eclipse, поэтому имеет все ее плюсы.

В качестве основных инструментов разработки программной части у Android устройств являются IDE и язык программирования. Существует большое количество языков программирования для разработки Android-приложений [50]. В настоящий момент Kotlin признан официальным языком для разработки Android приложений. В Kotlin можно использовать все существующие фреймворки и библиотеки Java. В отличие от Java (предыдущий официальный язык программирования для Android) он [50]:

1. null-безопасен;
2. Имеет функции-расширения, которые можно использовать вместо наследования;
3. Имеет специально предназначенные классы для работы с данными;
4. Во множестве случаев не требует явно приводить типы;
5. Позволяет использовать лямбда выражения и пр.

В качестве языка программирования для приложения Android был выбран Kotlin по вышеизложенным причинам. Kotlin поддерживается такими средами разработки как Android Studio и IntelliJ IDEA. Android Studio является официальной IDE от Google для разработки Android-приложений и базируется на IntelliJ IDEA, к тому же она направлена только на разработку для Android, в то время как IntelliJ IDEA поддерживает разработку и на других языках программирования, таких, как PHP, Scala, Ruby и пр. (набор языков зависит от версии IDE), в следствии этого,

IntelliJ IDEA более медленная чем Android Studio. В результате выше описанного была выбрана IDE Android Studio для разработки мобильного приложения.

3.9 АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧИ

Перед отправкой данные зашифровываются, соответственно для дальнейшей работы с ними, их требуется дешифровать. В рамках работы был проведен поиск и анализ методов шифрования/дешифрования данных, которые чаще всего применяются для систем связи Li-Fi. Ниже в таблице 6 представлены достоинства и недостатки обнаруженных методов шифрования/дешифрования.

Таблица 6 – Преимущества и недостатки методов шифрования для Li-Fi систем

Название	Достоинства	Недостатки
Семейство Secure Hash Algorithm (SHA) [57]	<ul style="list-style-type: none"> - надежность тем выше, чем выше версия SHA; - сложность реализации повышается при увеличении версии; 	<ul style="list-style-type: none"> - быстродействие понижается с версией SHA; - на первых версиях SHA хэш-функции имеют коллизии, - удлинение передаваемых сообщений;
Алгоритм Диффи–Хеллмана и его производные (ECDH) [57]	<ul style="list-style-type: none"> - надежный, - сложности в реализации; 	<ul style="list-style-type: none"> - медленный, так как применяются операции деления;
Шифр Цезаря [58]	<ul style="list-style-type: none"> - быстрый; - простой в реализации; 	<ul style="list-style-type: none"> - ненадежный;
RC4–алгоритм [53]	<ul style="list-style-type: none"> - достаточно быстрый; - простой в реализации. 	<ul style="list-style-type: none"> - средняя степень надежности;
XOR–шифрование [59]	<ul style="list-style-type: none"> - очень быстрый; - простой в реализации. 	<ul style="list-style-type: none"> - ненадежный.

Так как в рамках передающего модуля был выбран алгоритм шифрования RC4 [53], поскольку он обеспечивает компромисс между защитой данных и

скоростью передачи в системе, то дешифровка должна производиться согласно этому алгоритму. Ниже на рисунке 22 представлена блок-схема [61] алгоритма дешифрования RC4 для дальнейшей работы с данными.

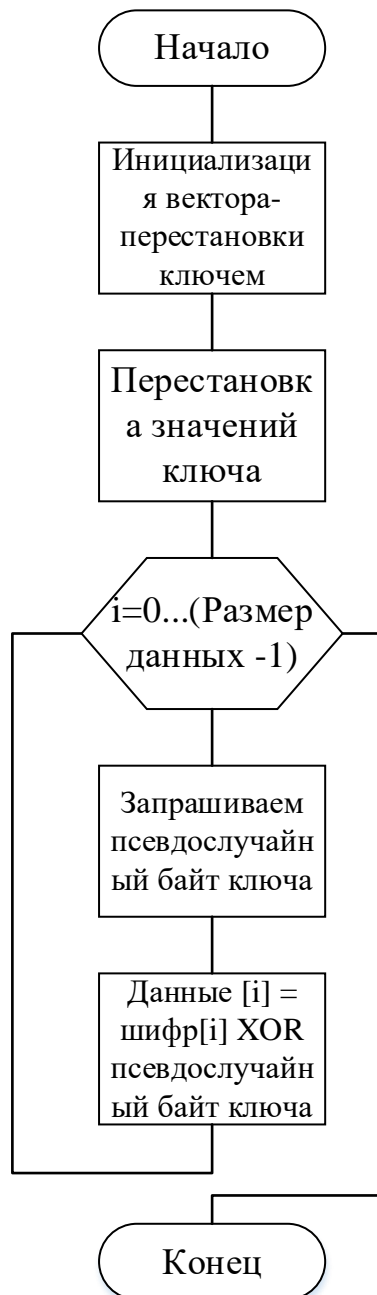


Рисунок 22 – Блок-схема алгоритма дешифрования данных RC4

Для контроля приходящей информации и выявления ошибок, было принято решение использовать циклически избыточный код (CRC). Так как другие методы, такие как посимвольный контроль четности и поблочный контроль четности являются слишком простыми в реализации, и менее надежные. Суть их в том, что в конце посылки, либо блока посылки добавляется дополнительный бит четности,

если проверяется четность, либо нечетности, если наоборот. В посылке при передаче может измениться один бит с 0 на 1, а другой с 1 на 0 и методы контроля четности не смогут обнаружить ошибку. CRC – простейшая контрольная сумма, образуемая суммированием количества единичных бит в посылке, данный вариант имеет те же недостатки, что и методы описанные выше. Поэтому вместо обычного использования контрольной суммы, будет применен вариант с делением на полином, от которого получают остаток от деления посылки данных, тем самым, он является контрольной суммой, это более надежный способ выявления ошибок в пакетах. Данные методы считаются наиболее надежными с возрастанием типа CRC. Типе CRC8 используется как простая контрольная сумма, в других же типах CRC16, 32, 64 используют только остаток от деления на специальный полином, степень которого соответствует типа CRC. Особо важным считается правильный выбор полинома, от которого зависит качество использования данной CRC, существуют стандартные испытанные значения [54]. Для конкретной работы в учебных целях достаточно использовать CRC16, который отвечает требованиям: посылка данных имеет длину 256 бит, из которых 2 старт-бита и 2 стоп-бита, 32 бита – отведено под контрольную сумму и остальное 220 бит данных. Посылка состоит из удвоенных значений из-за требований ООК ZR модуляции. Также был выбран стандартный полином для CRC16: $x^{16} + x^{15} + x^2 + 1$ [54]. Алгоритм вычисления контрольной суммы и применения ее к посылке данных после получения в данном случае прост:

1. Шаг первый – исключить дополнительные биты из посылки из-за требований модуляции;
2. Далее шаг второй – отделить контрольную сумму от посылки данных;
3. Шаг третий – получить остаток от деления посылки данных на 11000000000000101;
4. Шаг четвертый – сравнить полученные контрольные суммы на наличие ошибок в посылке.

На рисунке 23 представлено описание алгоритма приема данных на микроконтроллере в виде блок-схемы [61]. Детальное описание алгоритма см. в пункте 4.2 и в приложении к ВКР.

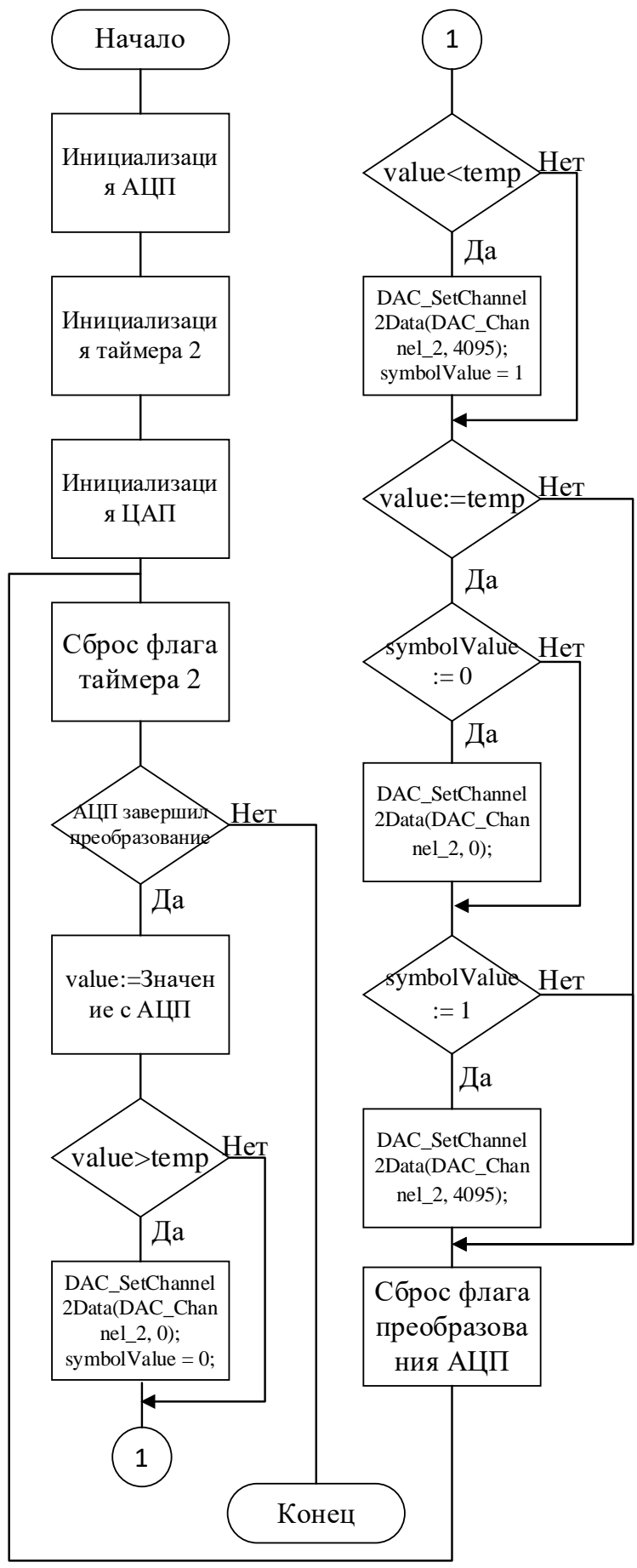


Рисунок 23 – Блок-схема алгоритма приема данных на микроконтроллере

3.10 ОПИСАНИЕ ДАННЫХ

Разрабатываемый прототип нацелен на прием данных любого формата с другого устройства на текущее устройство Android. После попадания данных на фоторезистор, приемное устройство получает аналоговый сигнал, который АЦП преобразует в цифровую форму согласно выбранному типу модуляции. Далее данные поступают на мобильное устройство в виде аналогового сигнала и преобразуются в цифровой код. Входные данные представляют из себя последовательность байт, после получения которых выполняется следующее:

- 1) Полученные данные собираются воедино, затем выполняется проверка на целостность с помощью контрольной суммы, происходит отсеивание дополнительных бит, которые были добавлены во время модуляции и выполняется декодирование полученного массива в соответствии с ключом XOR – шифрования, после этого из массива выбирается информация, относящаяся к файлу и преобразовывается в непрерывную;
- 2) Формируется имя файла согласно шагу 1;
- 3) Определяется формат файла (в будущем система должна принимать любые типы файлов) согласно шагу 1 и создается сам файл в указанной директории;
- 4) Шаг 1 повторяется для всех данных этого файла.

В результате завершения всех шагов полученные данные записываются в ранее созданный файл. Выходными данными являются файлы полученные с передающего устройства. На рисунках 24-25 представлена блок-схема [61] алгоритма обработки полученных данных с микроконтроллера на мобильном устройстве.

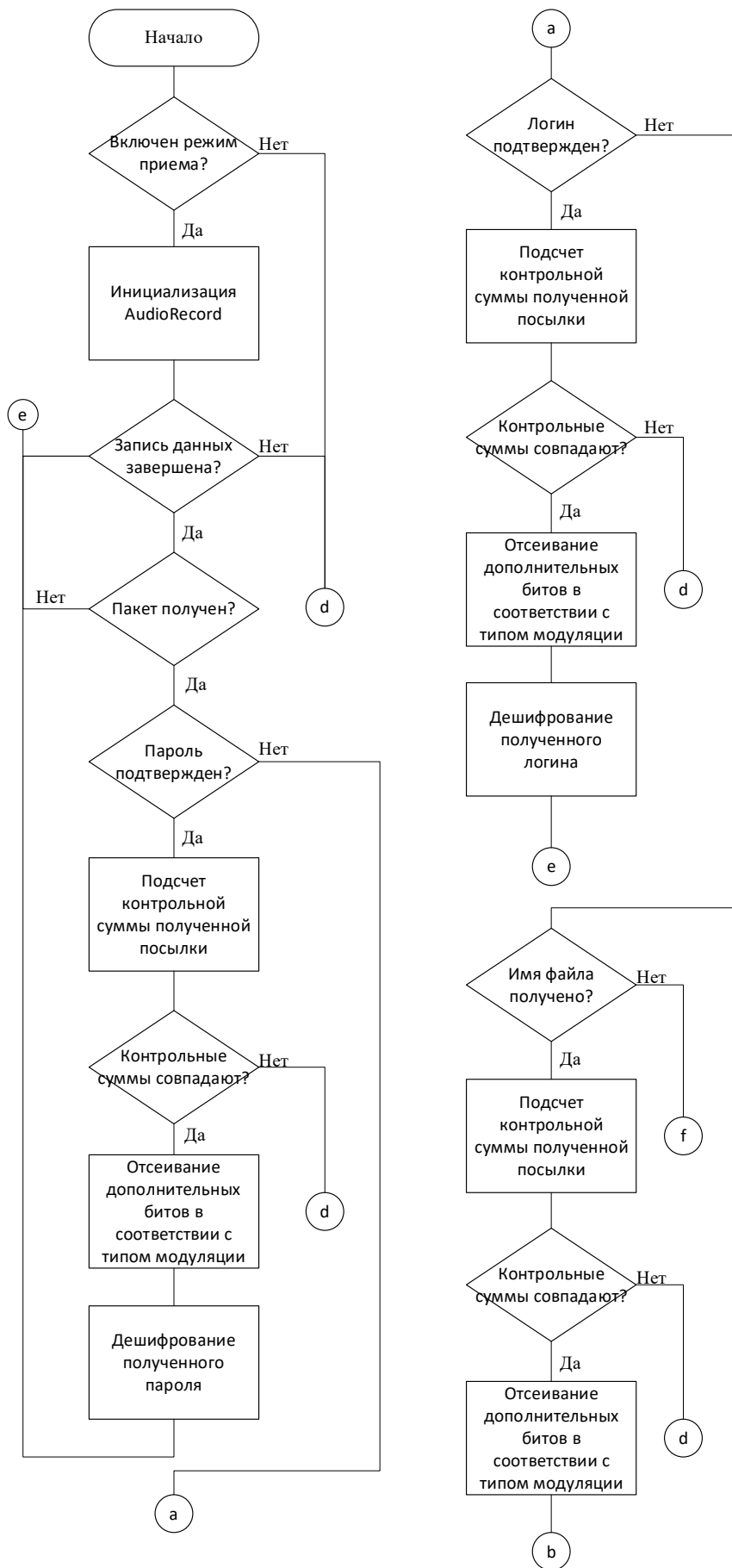


Рисунок 24 – Блок-схема алгоритма обработки данных на мобильном устройстве

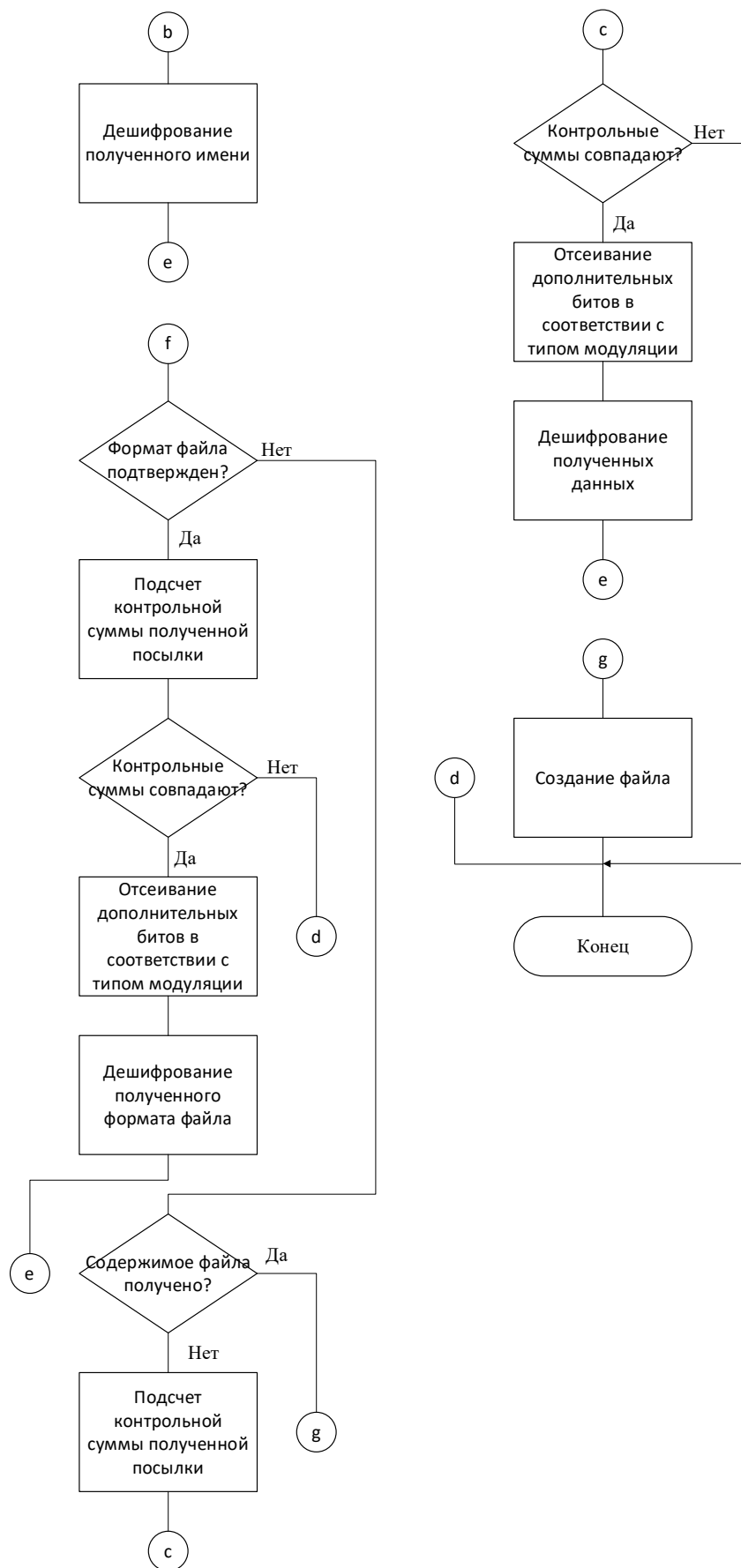


Рисунок 25 – Блок-схема алгоритма обработки данных на мобильном устройстве
(окончание)

4. РЕАЛИЗАЦИЯ

В рамках выполнения ВКР, были реализованы: мобильно приложение для устройств на операционной системе Android, программное обеспечение для микроконтроллера STM32F100RBT6, собрано устройство приемопередатчика на макетной плате. Далее представлены результаты работ в виде скриншотов и фотографий.

4.1 ПРИЛОЖЕНИЕ ANDROID

Для устройства приёмопередатчика было разработано мобильное приложение на операционной система Android, состоящее из 4 активити: главное меню приложения – MainActivity, режим приема данных – MainActivityReceive, режим настроек мобильного приложения – MainActivitySetting и справка – MainActivitySpravka.

На рисунках 26-27 представлен интерфейс главного окна мобильного приложения для взаимодействия с пользователем. С его помощью пользователь может перейти в необходимый раздел: передать какие-то файлы, осуществить прием этих файлов, задать первоначальные данные для безопасной передачи или приема, а также посмотреть информацию о самом приложении.

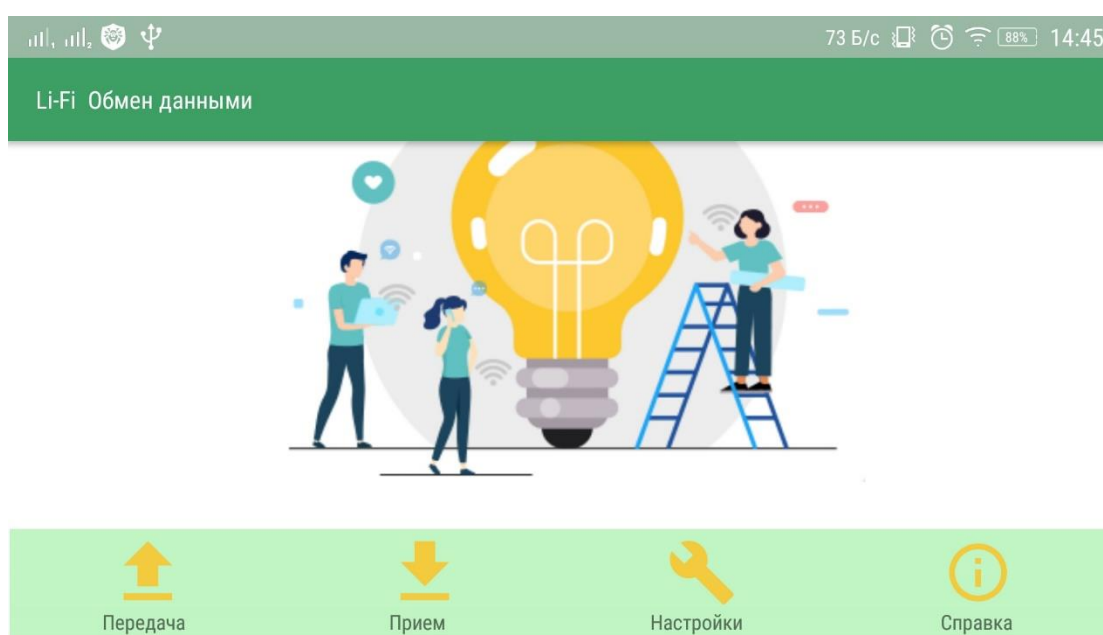


Рисунок 26 – Главное окно мобильного приложения горизонтальный формат

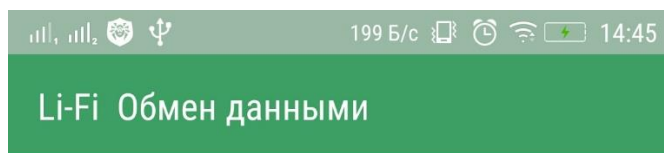


Рисунок 27 – Основное окно мобильного приложения вертикальный формат

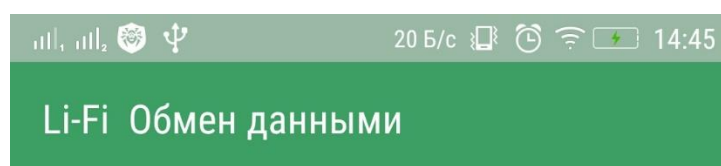
В данном окне были реализованы следующие методы для перехода между окнами:

1. `override fun onCreate (savedInstanceState: Bundle?)` – метод отвечает за создание окна меню приложения;
2. `fun startActivity(view: View)` – метод перенаправляющий пользователя в окно передачи данных;
3. `fun receiveActivity(view: View)` – метод перенаправляющий пользователя в окно приема данных;

4. `fun settingActivity(view: View)` – метод перенаправляющий пользователя в окно настройки приложения для безопасной работы с неизвестными устройствами;

5. `fun spravkaActivity(view: View)` – метод перенаправляющий пользователя в окно передачи данных.

На следующих рисунках 28-29 представлено окно со справочной информацией для пользователя о назначении используемого им приложения, а также рекомендации по первоначальной настройке приложения.



Справочная информация

Приложение предназначено для обмена данными с помощью света.

Перед началом работы установите имя своего устройства и пароль.

При обмене данными пароли на передающей и принимающей сторонах должны быть установлены одинаковые.



Рисунок 28 – Окно справки мобильного приложения вертикальный формат

Справочная информация

Приложение предназначено для обмена данными с помощью света.

Перед началом работы установите имя своего устройства и пароль.

При обмене данными пароли на передающей и принимающей сторонах должны быть установлены одинаковые.



Рисунок 29 – Окно справки мобильного приложения горизонтальный формат

В окне справки мобильного приложения реализована следующая функция:

1. `override fun onCreate (savedInstanceState: Bundle?)` – метод отвечающий за создание окна со справочной информацией;
2. `fun sendMain(view: View)` – метод осуществляющий перенаправление пользователя на главное окно мобильного приложения.

На рисунках 30-31 изображено окно настроек мобильного приложения, где присутствуют поля, в которых отображаются имя и пароль устройства, установленные пользователем ранее и которые он может изменить, либо пустые поля, которые необходимо заполнить, если приложение используется впервые.

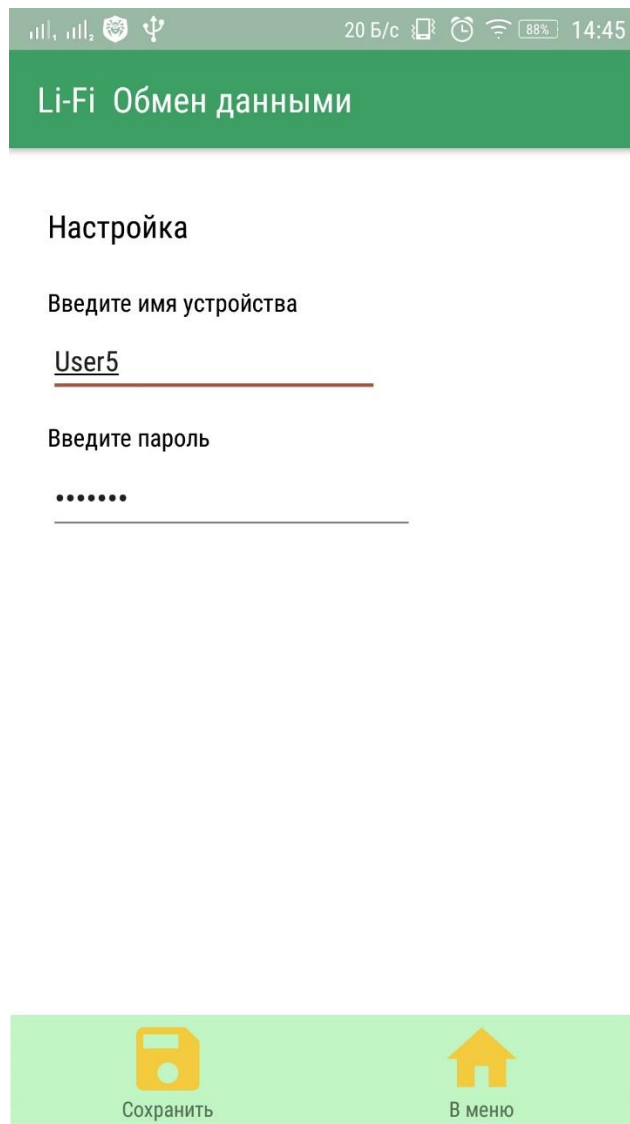


Рисунок 30 – Окно настроек мобильного приложения вертикальный формат

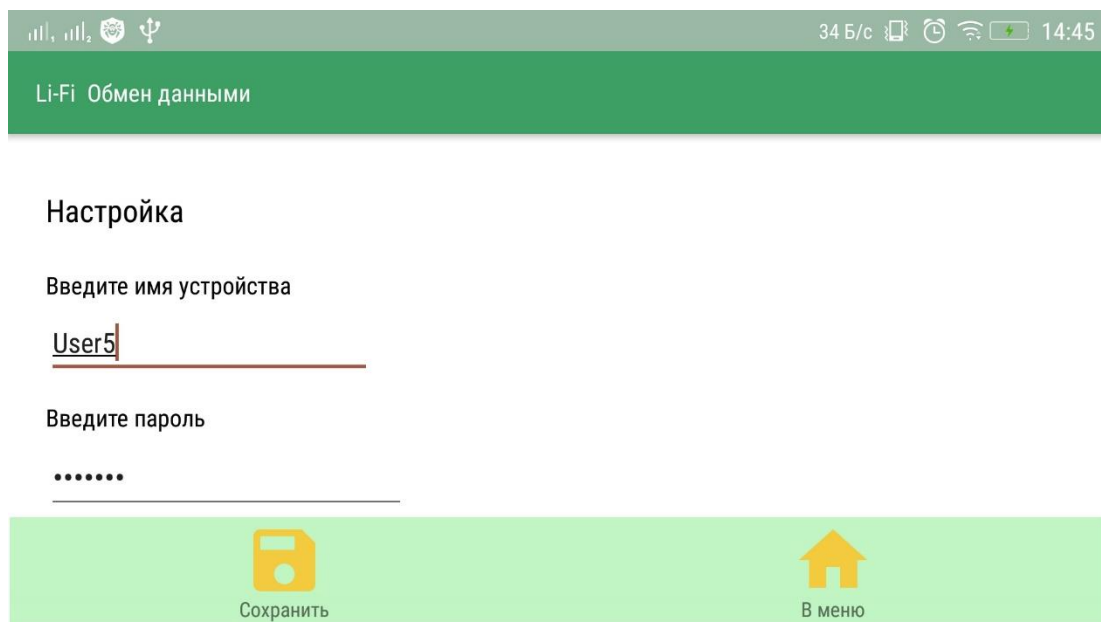


Рисунок 31 – Окно настроек мобильного приложения горизонтальный формат

После того как пользователь заполнит поля и нажимает на кнопку сохранить, указанные данные сохраняются в файл на мобильном устройстве, и в дальнейшем будут использоваться для передачи и/или приема данных.

Для обработки полученных данных и дальнейшей работы с приложением были реализованы методы:

1. `fun onCreate(savedInstanceState: Bundle?)` – данный метод создает экранную форму для пользователя и отображает ранее введенные им имя и пароль для устройства, если такие имеются;
2. `fun mainWindowActivity(view: View)` – метод перенаправляющий пользователя на главное окно мобильного приложения, чтобы тот мог продолжить работу с ним;
3. `fun saveDeviceSettings(view: View)` – данный метод обрабатывает введенную пользователем информацию, если данная информация содержит недопустимые данные, то он предупреждает его об этом, иначе создает файл, в который записываются сведения об устройстве.

На изображениях 32-33 представлено окно, отвечающее за прием информации с другого устройства посредством технологии Li-Fi. В данном окне пользователь имеет возможность начать прием данных с другого мобильного устройства на котором установлено данное приложение и имеется приёмопередатчик. Перед началом приема пользователю необходимо произвести первоначальную настройку приложения, если он этого не делал, то система оповестит его об этом и после он сможет продолжить работу.

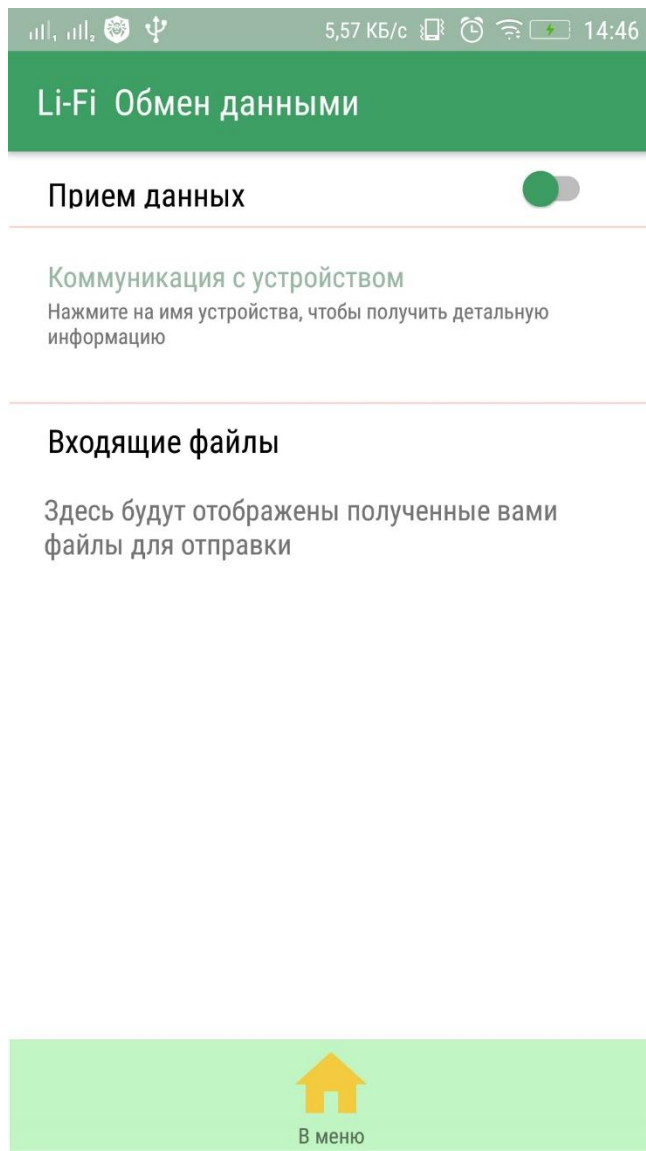


Рисунок 32 – Окно для работы в режиме приема данных вертикальный формат

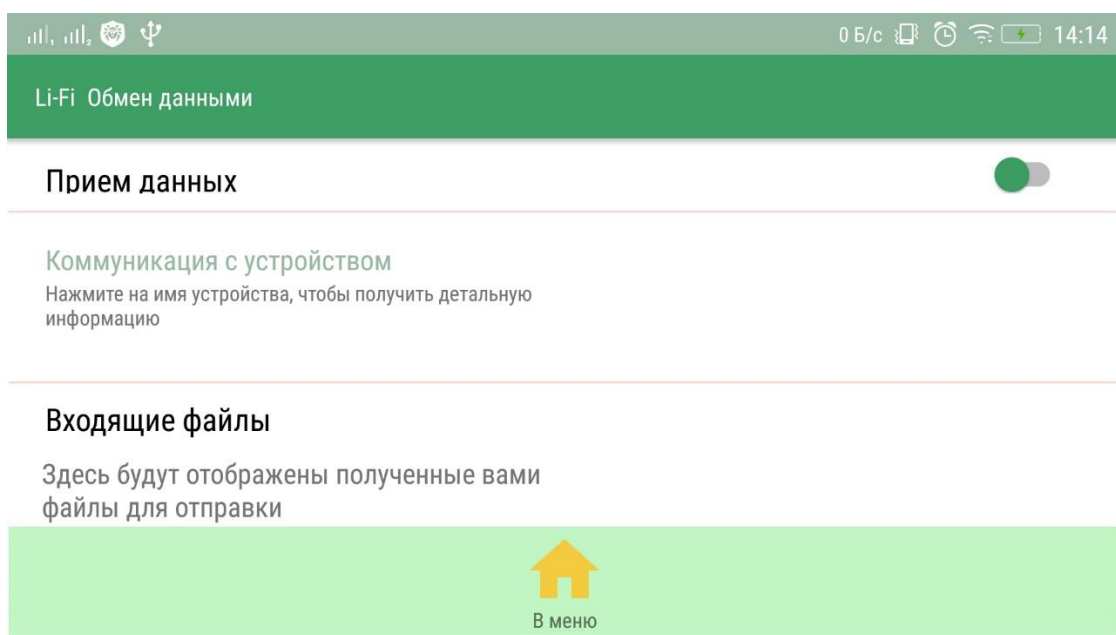


Рисунок 33 – Окно для работы в режиме приема данных горизонтальный формат

Для реализации функций приема данных на мобильном устройстве было принято решение использовать AudioRecord API [60], этот класс отвечает за получение необработанных данных с микрофона или другого источника сигнала. Для приема данных из этого класса были использованы следующие методы:

1. `AudioRecord()` – представляет собой конструктор для создания объекта класса, который получает нужные ресурсы системы для получения необработанной информации, результатом выполнения является объект данного класса;

2. `getMinBufferSize()` – метод, возвращающий минимальный размер внутреннего буфера, при котором объект класса `AudioRecord` сможет функционировать корректно;

3. `read()` – для получения данных из внутреннего буфера применяется данный метод, который возвращает количество прочитанных байт;

4. `stop()` – данный метод объявляет о завершении записи во внутренний буфер;

5. `release()` – этот метод высвобождает все системные ресурсы, которые ранее были заняты объектом класса `AudioRecord`;

6. `startRecording()` – метод обозначающий начало записи данных во внутренний буфер мобильного устройства.

Также для обработки полученных данных были реализованы методы:

1. `fun onCreate(savedInstanceState: Bundle?)` – данный метод отвечает за создание экрана приема, а также за включение и отключение режима приема данных, также в нем реализована проверка на проведение первоначальной настройки мобильного приложения и отображении всплывающей информации о передающем и принимающем устройствах;

2. `fun userConfirm(dataUsers: ByteArray)` – данный метод применяется для подтверждения полученного логина и пароля с передающего устройства, в результате которого осуществляется решение о приеме данных с неизвестного мобильного устройства или об отказе;

3. `fun fileCreate(dataUsers: ByteArray)` – метод осуществляющий прием самих файлов, их дешифровку, демодуляцию и создание, если была пройдена проверка на авторизацию неизвестного мобильного устройства;

4. `fun crcPack(packData: BitSet) : BitSet` – метод подсчитывающий контрольную сумму полученного пакета, которая в дальнейшем служит для определения целостности получаемых данных;

5. `fun receiveData()` – метод отвечающий за инициализацию объектов и ресурсов, необходимых для обеспечения приема информации с неизвестного мобильного устройства и в котором начинается прием данных;

6. `fun mainWindowOpen(view: View)` – метод перенаправляющий пользователя на главное окно мобильного приложения, чтобы тот мог продолжить работу с ним;

7. `fun showWarning(textWarning: String)` – метод отвечающий за уведомление пользователя о различного рода событиях, на вход функции подается строка, которую необходимо отобразить на экране.

В мобильном приложении на операционной системе Android также было реализовано RC4 дешифрование получаемых данных в соответствии с повышением безопасности определенной в передающей части ВКР. Для этого был создан класс RC4, с полями итераторов $x = 0$ и $y = 0$, и полем S ключевого потока, он имеет конструктор, который при создании объекта, проводит инициализацию ключевого потока S предназначенным для этого методом. В классе реализованы методы:

1. `fun initParam(key: ByteArray)` – метод, который вызывается в конструкторе, данный метод сначала инициализирует ключевое поле значениями от 0 до 255, а затем, используя входной ключ `ByteArray` (в данном случае “Light”), выполняет перестановки в ключевом поле;

2. `fun keyRandom(): Int` – метод, который возвращает псевдослучайное значение из ключевого поля S для дальнейшего применения операции XOR к нему и значению из массива пользовательских данных;

3. `fun encode(dataBinaryUsers: ByteArray, size: Int): ByteArray` – метод для шифрования данных, применяет к каждому байту данных пользователя и псевдослучайному значению ключа *S*, преобразованного в байты, операцию XOR, возвращает массив зашифрованных данных;

4. `fun decode(dataBinaryUsers: ByteArray, size: Int): ByteArray` – метод, для дешифрования данных, применяется к каждому байту зашифрованных данных пользователя и псевдослучайному значению ключа *S*, преобразованного в байты, операцию XOR и возвращает массив расшифрованных данных.

Работа алгоритма дешифровки данных представлена на рисунке 34, для наглядности мобильное приложение было запущено в режиме отладки, чтобы отразить данные после расшифровки.

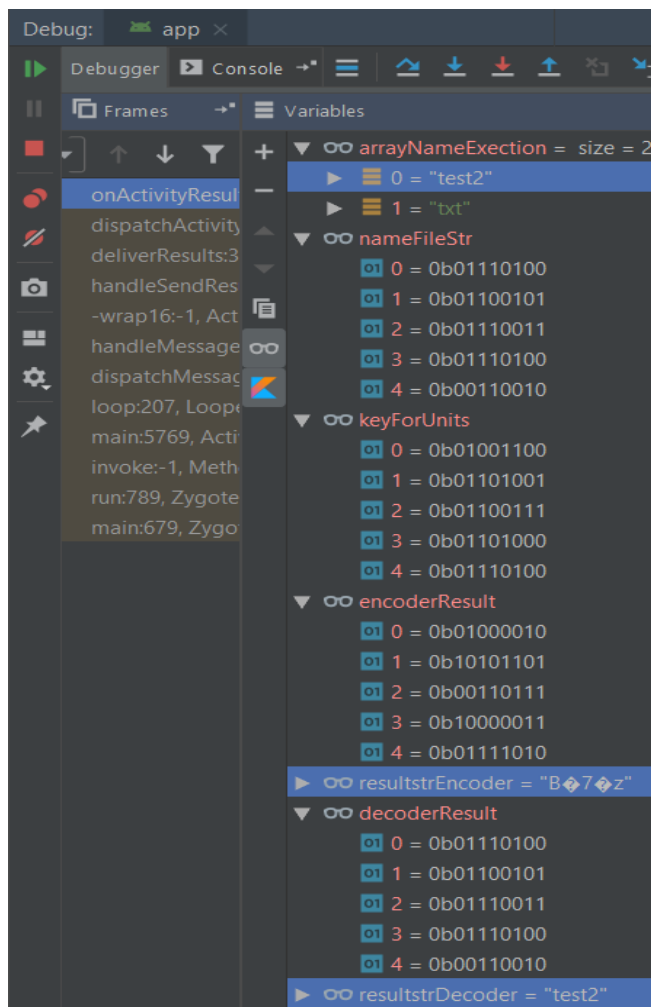


Рисунок 34 – Результат работы алгоритма дешифровки

Реализация класса дешифрования и его методов, приведена в листинге Г.1 приложения Г.

Ниже на рисунках 35 и 36 представлены результаты работы в окне настроек мобильного приложения. Настройки после сохранения хранятся в файле dataUser.txt, для файла установлен режим MODE_PRIVATE, который гарантирует сохранность данного файла, ограничивая к нему доступ из других приложений. По умолчанию установлены имя устройства «User1» и пароль «User123». Также были установлены ограничения по длине для имени устройства, которое не должно быть меньше 3 символов и больше 11 и для ключа безопасности, длина которого должна быть в пределах от 5 и до 11 символов. При несоответствии вводимых данных установленным ограничениям, пользователь увидит соответствующее уведомление на экране мобильного телефона. Если файл с конфигурацией был удален по какой-то причине, то при вводе новых данных об устройстве файл будет создан заново.

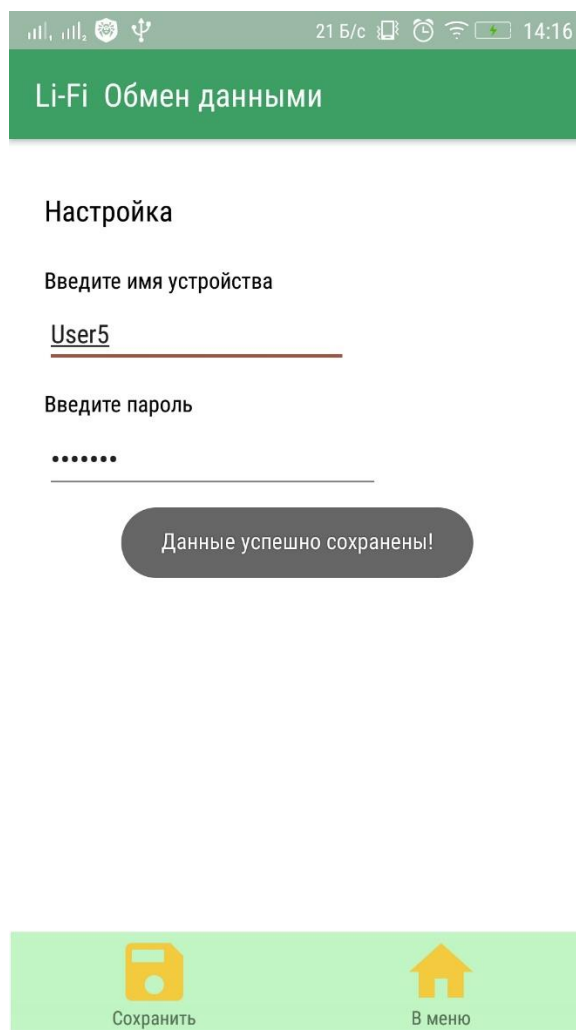


Рисунок 35 – Результат сохранения данных об устройстве

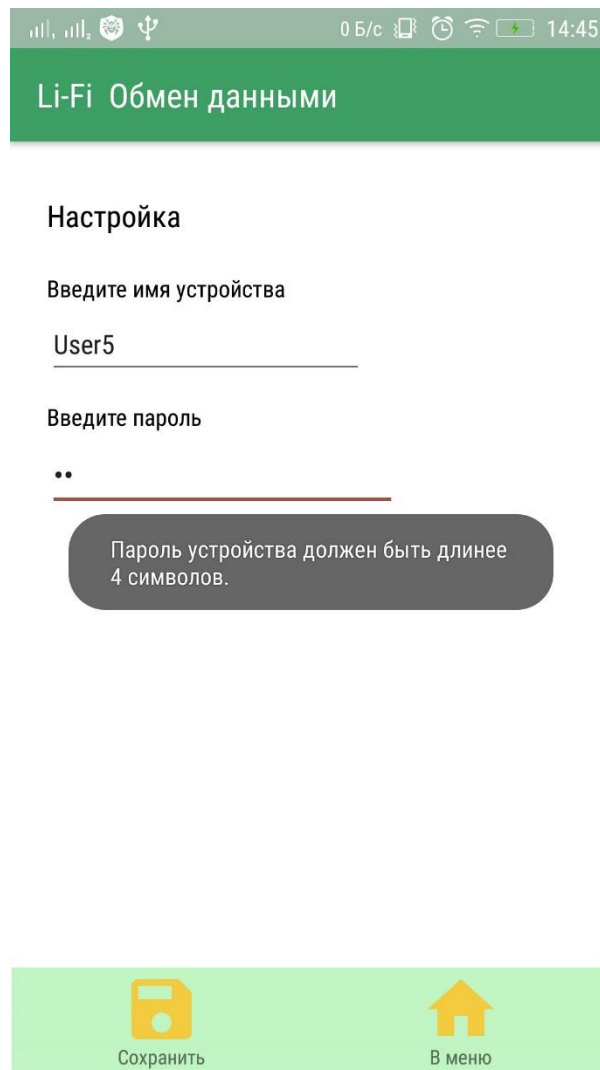


Рисунок 36 – Уведомление пользователя о нарушении установленного ограничения

Фрагменты работы мобильного приложения приведены на скриншотах 37-40. На рисунке 37 представлено получение информации об устройстве по нажатию на имя устройства «User1».

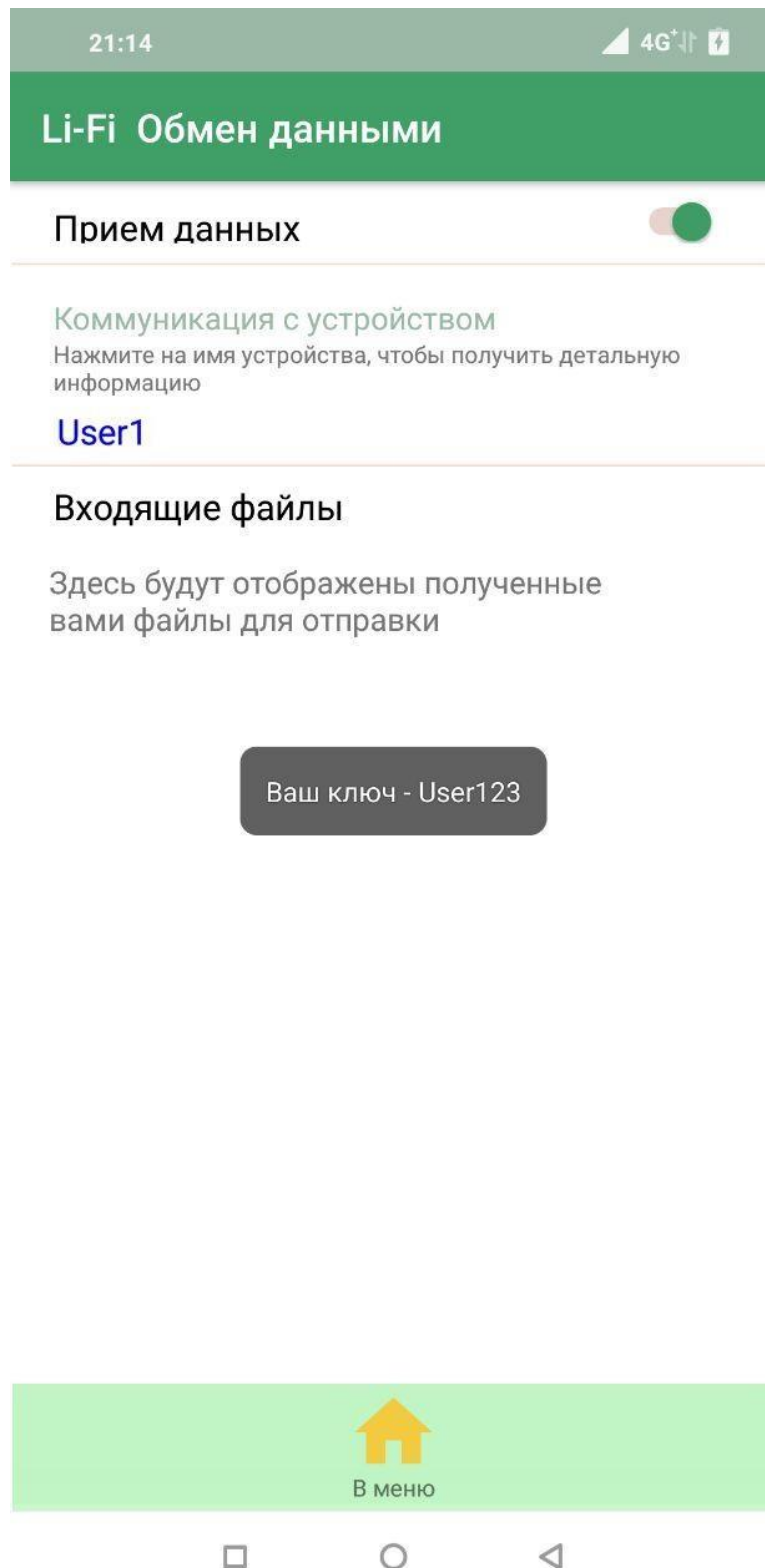


Рисунок 37 – Получение информации об устройстве

Изображение 38 иллюстрирует оповещение пользователя, о том, что файл был полностью загружен.

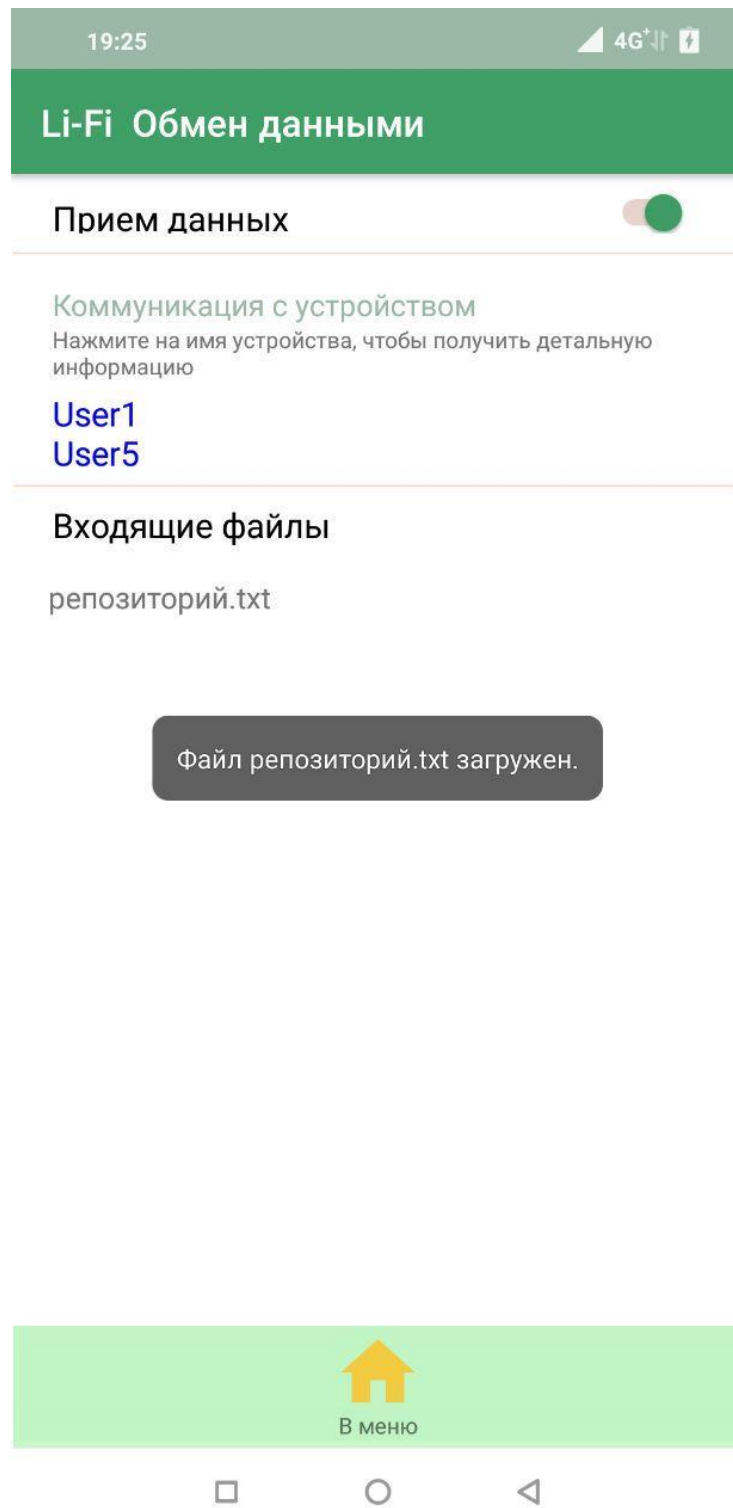


Рисунок 38 – Оповещение о загрузке файла

На рисунке 39 представлено оповещение пользователя, если прием файлов был прерван непосредственно самим пользователем.

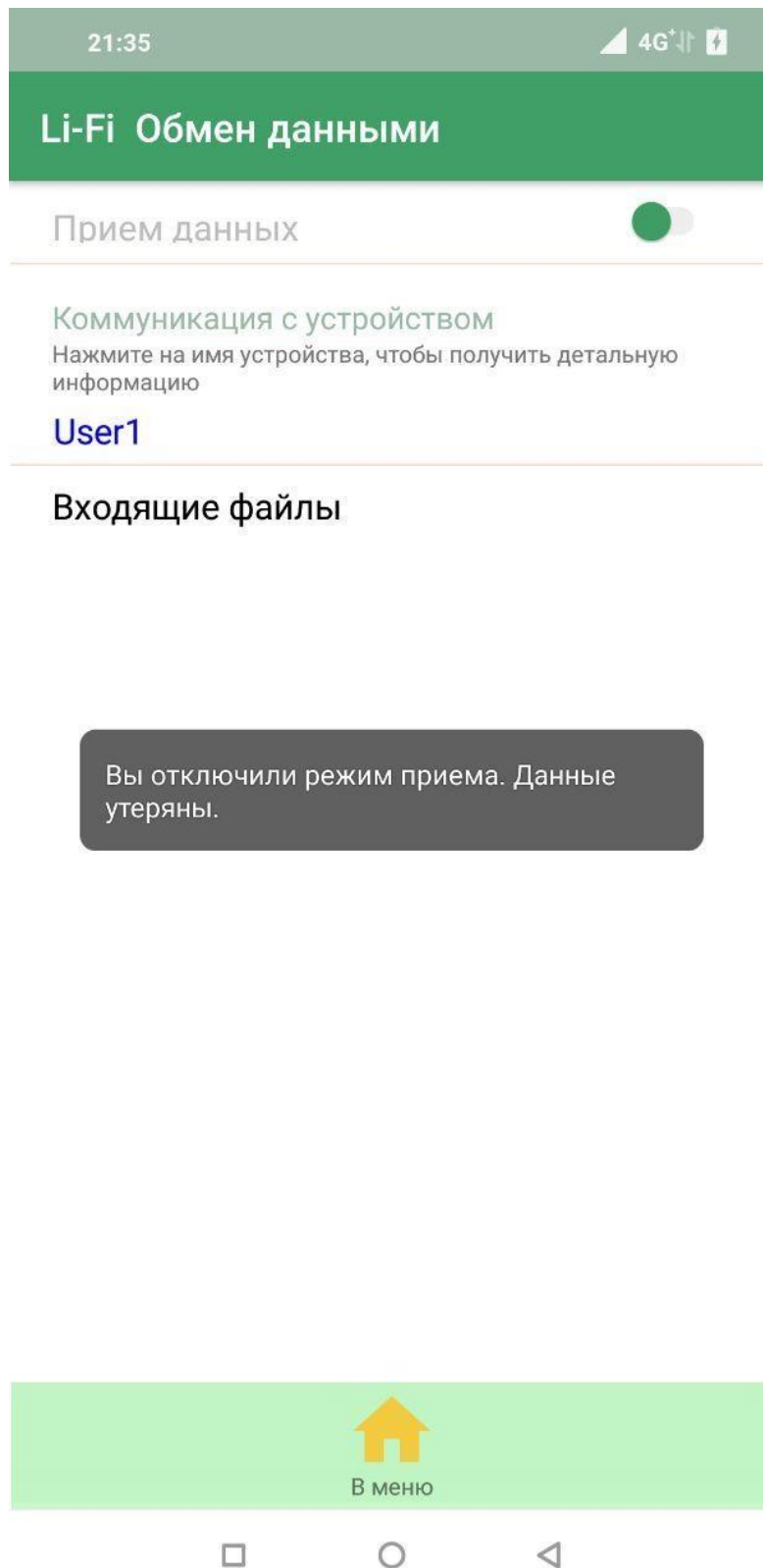


Рисунок 39 – Оповещение об отмене загрузки

На рисунке 40 изображено, оповещение пользователя, информирующее его о том, что все файлы были успешно приняты.

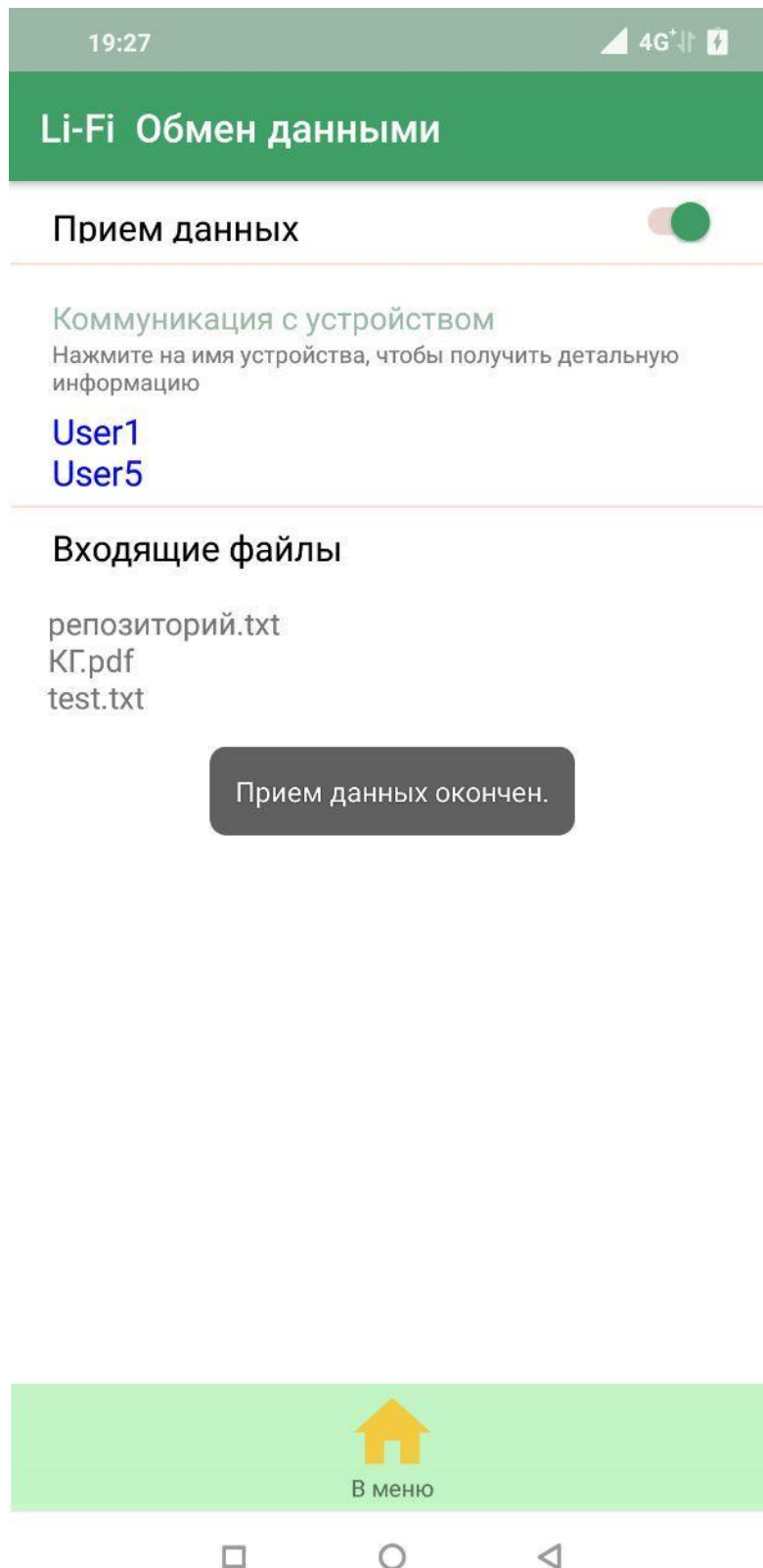


Рисунок 40 – Окончание приема информации

Ниже в таблице 7 представлены для удобства классы программы и реализованные в них методы.

Таблица 7 – Реализованные методы приложения

Классы программы	Методы	Назначение
MainActivity	override fun onCreate (savedInstanceState: Bundle?)	Создание экрана меню
	fun receiveActivity(view: View)	Открытие экрана режима приема
	fun settingActivity(view: View)	Открытие настроек
	fun spravkaActivity(view: View)	Открытие справки
	fun sendActivity (view: View)	Открытие экрана режима передачи
MainActivitySpravka	fun sendMain(view: View)	Возвращение к экрану меню
	override fun onCreate (savedInstanceState: Bundle?)	Создание экрана справки
MainActivitySetting	override fun onCreate (savedInstanceState: Bundle?)	Создание экрана настроек
	fun mainWindowActivity(view: View)	Возвращение к экрану меню
	fun saveDeviceSettings(view: View)	Сохранение настроек
MainActivityReceive	override fun onCreate (savedInstanceState: Bundle?)	Создание экрана приема
	fun userConfirm(dataUsers: ByteArray)	Подтверждение передающего устройства
	fun fileCreate(dataUsers: ByteArray)	Преобразование получаемой информации
	fun receiveData()	Прием данных

Продолжение таблицы 7

Классы программы	Методы	Назначение
MainActivityReceive	fun showWarning (textWarning: String)	Создание сообщений для пользователя
	fun mainWindowOpen(view: View)	Возвращение к экрану меню
RC4	private fun initParam(key: ByteArray)	Генерация ключа шифрования
	private fun keyRandom(): Int	Получение случайного байта ключа шифрования
	fun encode(dataBinaryUsers: ByteArray, size: Int): ByteArray	Шифрование данных
	fun decode(dataBinaryUsers: ByteArray, size: Int): ByteArray	Дешифрование данных
–	fun crcPack (packData:BitSet): BitSet	Расчет CRC16

Реализация приема данных в мобильном приложении приведена в листинге Д.1 приложения Д.

4.2 ПРОГРАММА МИКРОКОНТРОЛЛЕРА

Для программы управления приемопередатчиком был написан код для приема данных. Он находится в функциях void TIM2_IRQHandler(void), void ADC1_IRQHandler(void), void ADCInit(void), void DACMobileInit(void) и void main(void). В функции void ADCInit(void) происходит инициализация и настройка необходимой периферии для реализации задачи приема. Основная периферия – АЦП (1-ый канал), ЦАП (2-й канал) и таймер 2. АЦП на микроконтроллере настроен в сторожевом режиме, то есть при выходе аналогового сигнала с фоторезистора за установленные границы 1570 и 1185 единиц происходит

прерывание и АЦП начинает прием данных, выполняется функция void ADC1_IRQHandler(void). При наступлении прерывания таймер с установленной частотой, для учебных целей у таймера настроена частота 8 кГц, опрашивает АЦП и в соответствии с алгоритмом приема записывает данные в переменную. Для проверки работоспособности приема был принят символ «Л» (эль), его бинарный код: 10100111. Затем после приема данных осуществляется передача полученного бита ЦАП, который был ранее проинициализирован в функции void DACMobileInit(void) для того, чтобы осуществлять обмен данным между мобильным устройством и устройством приемопередатчика, который также работает на частоте 8 кГц. Данные передаются через правый канал разъема TRRS jack 3.5 Реализация приема данных и передачи их на мобильное устройство на микроконтроллере приведена в листинге E.1 приложения E.

Ниже в таблице 8 представлены для удобства функции с их описанием, реализованные в прошивке микроконтроллера приёмопередатчика.

Таблица 8 – Функции прошивки микроконтроллера

Функции	Описание
void TIM2_IRQHandler(void)	Функция содержит алгоритм преобразования информации полученной с помощью фоторезистора и дальнейшую ее обработку и передачу с помощью ЦАП микроконтроллера.
void ADC1_IRQHandler(void)	В данной функции осуществляется обработка прерывания АЦП в сторожевом режиме.
void ADCInit(void)	Данная функция содержит конфигурацию АЦП, который отвечает за преобразование информации полученной с фоторезистора.

Продолжение таблицы 8

Функции	Описание
void DACMobileInit(void)	В функции определяется конфигурация ЦАПа микроконтроллера для корректного обмена информацией между устройством приемопередатчика и мобильным устройством.
void main (void)	В данной функции осуществляется первоначальный вызов функций инициализации необходимой периферии для работы устройства приемопередатчика.

На рисунке 41 представлен прототип приемопередатчика системы связи Li-Fi для передачи данных между мобильными устройствами Android.

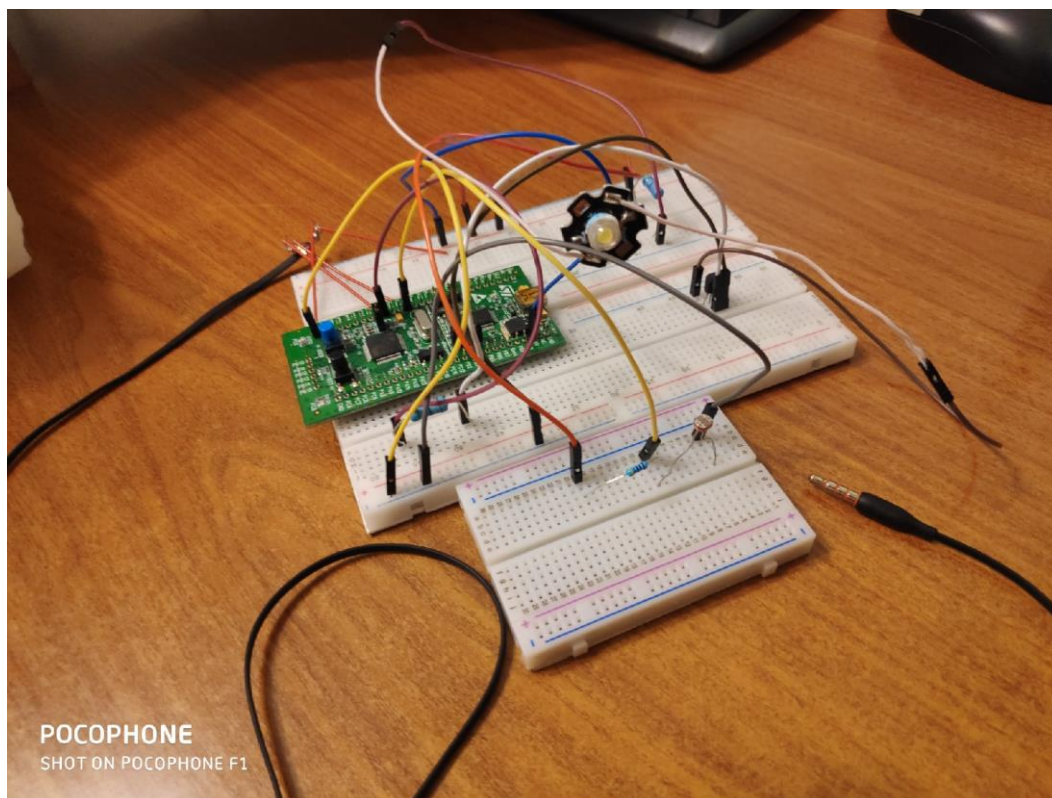


Рисунок 41 – Прототип устройства приемопередатчика

5. ТЕСТИРОВАНИЕ

Для проверки работоспособности прототипа принимающего модуля программно-аппаратного комплекса системы связи Li-Fi для мобильных устройств было проведено функциональное тестирование. Тестирование – это процесс исследования разрабатываемого программного продукта, целью которого является проверка соответствия между реальным поведением программы и ее ожидаемым поведением на конечном наборе тестов. Функциональное тестирование позволяет проверить реализуемость функциональных требований, то есть способности программы в определенные моменты времени решать задачи, необходимые пользователям (функциональные требования) [62]. Результаты тестирования принимающего модуля мобильного приложения и прошивки для STM32 сведены в таблицу 8.

Таблица 8 – Результаты функционального тестирования

Название теста	Входные данные	Ожидаемый результат	Полученный результат	Результат теста
Прием файлов	Пользователь перевел переключатель в состояние On при приеме в экране приема, не указав при этом имя и ключ устройства	Вывод сообщения, что не заданы имя устройства и ключ безопасности, прием не начат	Всплывающее сообщение, не заданы имя устройства и ключ безопасности, прием не начат	Тест пройден

Продолжение таблицы 8

Название теста	Входные данные	Ожидаемый результат	Полученный результат	Результат теста
	Пользователь перевел переключатель в состояние On при приеме в экране приема, указав при этом имя и ключ устройства несовпадающий с передающим устройством	Прием начат, проверка ключей безопасности, вывод сообщения, что ключ безопасности не совпадает, прием прерван	Прием начат, проверка ключей безопасности, всплывающее сообщение, ключ безопасности не совпадает, прием прерван	Тест пройден
	Пользователь перевел переключатель в состояние On при приеме в экране приема, указав при этом имя и ключ устройства совпадающий с передающим устройством	Прием начат, вывод сообщения, что файлы получены и прием завершен, появление имени и формата файла на экране режима приема	Прием файлов, всплывающее сообщение о получении файлов и завершении приема, появление имени и формата файла на экране режима приема	Тест пройден
Получение данных об устройстве	Пользователь нажал на имя устройства в экране приема	Получить информацию о ключе безопасности устройства	Всплывающее сообщение, показывающее ключ безопасности	Тест пройден

Продолжение таблицы 8

Название теста	Входные данные	Ожидаемый результат	Полученный результат	Результат теста
Получение справки	Пользователь нажал кнопку «Справка» в меню	Открытие экрана со справкой	Открытие экрана со справкой	Тест пройден
Прервать прием	Пользователь перевел переключатель в состояние Off при приеме в экране приема	Отмена приема и информирование об этом пользователя	Отмена приема, сообщение пользователю об отмене операции	Тест пройден
Задание имени и ключа	Пользователь в экране настроек вводит имя устройства и пароль и нажимает «Сохранить»	Введенные данные сохраняются	Введенные данные сохраняются с помощи записи в файл, всплывающее сообщение о сохранении	Тест пройден
	Пользователь в экране настроек вводит некорректное имя устройства или пароль и нажимает «Сохранить»	Вывод сообщения о неправильном вводе, новые данные не сохранены	Всплывающее сообщение о неправильном вводе без сохранения новых данных	Тест пройден

Как видно из таблицы 8 и скриншотов, приведенных в предыдущем разделе, принимающий модуль прототипа системы связи Li-Fi для приема данных между мобильными устройствами Android работает корректно.

6. ЗАКЛЮЧЕНИЕ

При разработке выпускной квалификационной работы был проведен анализ существующих решений для Li-Fi систем, способных выполнять схожие задачи. Были выявлены недостатки этих решений и принято решение о разработке нового устройства. Проанализировав полученные данные, был составлен полный список требований к системе.

Исходя из списка требований, была разработана архитектура будущего решения и изготовлен опытный образец устройства. По итогам выполнения дипломной работы было сделано следующее:

- 1 Поставлена задача.
- 2 Проведен анализ существующих решений.
- 3 Сделан вывод о необходимости создания нового устройства.
- 4 Разработаны требования к системе.
- 5 Осуществлено проектирование архитектуры.
- 6 Произведен подбор компонентов.
- 7 Спроектированы принципиальная электрическая схема и плата устройства.
- 8 Выбраны средства программной разработки.
- 9 Осуществлено написание программной части.
- 10 Произведена проверка на соответствие предъявленным требованиям.

В пояснительной записке к выпускной квалификационной работе отображены все основные этапы проектирования системы, разработка ее модулей и настройка для дальнейшего использования решения.

В будущем система может быть доработана до двунаправленной системы Li-Fi и использования выхода в Интернет, что понизит расходы по сравнению с другими технологиями организации сети, а также обеспечит высокую скорость обмена данными.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Новости Университета ИТМО. – <http://news.ifmo.ru/ru/science/photonics/news/6805/>. Дата обращения: 15.12.2019.
2. Elec.ru – Электротехнический интернет-портал. – <https://www.elec.ru/articles/li-fi-svetodiodnye-kommunikacii-v-pogone-za-potreb/>. Дата обращения: 15.12.2019.
3. ЭМИЛИНК – проектирование, производство и дистрибуция телекоммуникационных и электротехнических продуктов и решений для ИТ- и инженерной инфраструктуры зданий. – https://emilink.ru/preimuschestva_i_nedostatki_volokonno_opticheskiy_kabelya/. Дата обращения: 16.12.2019.
4. О витой паре. – http://netcab.narod.ru/vitpar_cab.html. Дата обращения: 16.12.2019.
5. Оптоволоконные линии связи. – <http://www.on-lan.ru/sety/C/2.1.2.htm>. Дата обращения: 20.02.2020.
6. Robohunter – сообщество робототехников. – <https://robohunter.com/news/perspektivnaya-slojnaya-i-dorogaya-vse-o-5g-samoi-progressivnoi-mobilnoi-svyazi16133/>. Дата обращения: 16.12.2019.
7. Khan, L. U. Digital Communication and Networks: Visible light communication: Applications, architecture, standartization and research challenges. Article / L. U. Khan. – Pakistan: Department of Electrical Engineering, University of Engineering & Technology, Peshawar, 2016. – 78-88 p.
8. Шнайдер, Г. Е. Использование технологии Li-Fi для передачи данных: материалы Международной научно-технической конференции / Г. Е. Шнайдер. – М.: МИРЭА, 2015. – 4 с.
9. Li-Fi: Будущее интернета – <https://habr.com/ru/post/435262/>. Дата обращения: 20.02.2020.

10. Социальный интернет-сервис Pinterest. – <https://www.pinterest.ru/pin/598204762984707260/>. Дата обращения: 17.12.2019.
11. Сайт компании OledComm. – <https://www.oledcomm.net/>. Дата обращения: 18.12.2019.
12. Сайт компании PureLiFi. – <https://purelifi.com/>. Дата обращения: 19.12.2019.
13. Сайт компании Signify. – <https://www.signify.com/ru-ru/our-company/news/press-release-archive/2018/20180430-philips-lighting-introduces-lifi>. Дата обращения: 19.12.2019.
14. Сайт компании Lucibel. – <http://www.lucibel.io/web/guest/communication>. Дата обращения: 19.12.2019.
15. Решение LiFiMAX от OledCom. – <https://www.oledcomm.net/lifi-max/>. Дата обращения: 13.01.2020.
16. Сайт о LiFi. – <https://lifi.co/news/oledcomm-launches-lifimax/>. Дата обращения: 13.01.2020.
17. Wu, X. Access point selection for hybrid Li-Fi and Wi-Fi networks / Wu, X., Safari, M., Haas, H. – IEEE Transactions on Communications, 2017. – pp. 10;
18. Soltani, M.D. Modeling the random orientation of mobile devices: Measurement, analysis and LiFi Use Case / M.D. Soltani, A.A. Purwita, Z. Zeng, H. Haas, M. Safari. – IEEE Transactions on Communications, 2019. – pp. 15;
19. Haas H. A guide to wireless networking by light / H. Haas, C. Chen, D. O'Brien. – University of Edinburgh, King's Buildings, UK and University of Oxford, Parks Road, Oxford, UK, 2017;
20. Петрусь, И.П. Современные методы построение коммуникационных линий в видимой части спектра электромагнитных волн / И.П. Петрусь – Изд-во УрГУПС. – С. 5.
21. Университет ИТМО. – факультет лазерной фотоники и оптоэлектроники. – <http://lfo.itmo.ru/li-fi>. Дата обращения: 13.01.2020.
22. Новости о технике. – <https://ww.9to5mac.com/2019/06/20/truelifi/#>. Дата обращения: 14.01.2020.

23. Рынок Микроэлектроники. –
<http://www.gaw.ru/html.cgi/txt/doc/dac/dacsindex.htm>. Дата обращения:
09.02.2020.
24. Developers – для разработчиков Android. –
<https://developer.android.com/about/dashboards/index.html>. Дата обращения:
10.02.2020.
25. MEANDERS.RU. – <https://meanders.ru/fotorezistor-opredelenie-i-vidy-kak-rabotajut-preimushhestva-i-nedostatki.shtml>. Дата обращения: 15.02.2020.
26. ЭЛЕКТРОИНФО.NET. – Электрика и радиодетали –
<https://electroinfo.net/poluprovodniki/chto-takoe-fotodiod.html>. Дата
обращения: 21.02.2020.
27. Студопедия – Фототранзистор. –
https://studopedia.su/7_39133_fototranzistor.html. Дата обращения: 21.02.2020.
28. ARDUINOMASTER. – <https://arduinomaster.ru/datchiki-arduino/photorezistor-arduino-datchik-sveta/#i-5>. Дата обращения: 15.01.2020.
29. Кочегаров, И. И. Микроконтроллеры AVR. Лабораторный практикум: уч. пособие. / И. И. Кочегаров, В. А. Трусов. – П.: Изд-во ПГУ, 2012. – 122 с.
30. 8-bit AVR Microcontroller with 1K Bytes In-System Programmable Flash: ATtiny13. (Datasheet). – <https://pdf1.alldatasheet.com/datasheet-pdf/view/80274/ATMEL/ATTINY13.html>. Дата обращения: 15.02.2020.
31. 8-bit Atmel Microcontroller with 4/8/16K Bytes In-System Programmable Flash: ATmega48/V, ATmega88/V, ATmega168/V. (Datasheet). – http://radio-point.narod.ru/datasheet/a1/atmega48_88_168.pdf. Дата обращения: 15.02.2020.
32. SAM E70 Atmel. (Datasheet). –
http://catalog.gaw.ru/index.php?page=component_detail&id=78844. Дата
обращения: 15.02.2020.
33. Microchip PIC16C71X 8-Bit CMOS Microcontrollers with A/D. (Datasheet). –
<http://www.microchip.ru/cdrom/ww1.microchip.com/downloads/en/DeviceDoc/30272a.pdf>. Дата обращения: 15.02.2020.

34. Infineon XMC1100 AB-Step Microcontroller Series for Industrial Application. (Datasheet). – https://www.infineon.com/dgdl/Infineon-xmc1100_AB-DS-v01_08_EN.pdf?fileId=5546d4624a0bf290014a4bdaff9325bd. Дата обращения: 15.02.2020.
35. Infineon XMC4100/4200 Series for Industrial Application. (Datasheet). – <https://www.compel.ru/item-pdf/d1fb9679fc8205a37b24c35b6afe3363/ps/infin~xmc4100.pdf>. Дата обращения: 15.02.2020.
36. STM32F100x4, STM32F100x6, STM32F100x8, STM32F100xB. (Datasheet). – <https://static.chipdip.ru/lib/820/DOC000820500.pdf>. Дата обращения: 15.02.2020.
37. EFM32 Giant Gecko Family EFM32GG. (Datasheet). – <https://www.silabs.com/documents/public/data-sheets/efm32gg-datasheet.pdf>. Дата обращения: 15.02.2020.
38. ЧИП и ДИП – магазин электронных компонентов. - <https://www.chipdip.ru>. Дата обращения: 16.02.2020.
39. AVRrobot.RU Умный дом. – http://avrobot.ru/product_info.php?products_id=1865. Дата обращения: 20.02.2020.
40. Бэйкер, Б. Что нужно знать цифровому инженеру об аналоговой электронике. / Б. Бэйкер, – М.: Додэка-XXI, 2010. – 360 с.
41. Mouser Electronics – дистрибьютором полупроводников и электронных компонентов. – <https://ru.mouser.com/ProductDetail/STMicroelectronics/STLQ50C50R?qs=CUsY4352MFhоWHVppG9%2F8w%3D%3D>. Дата обращения: 17.02.2020.
42. Терра электроника – web-портал для разработчиков электроники. – <https://www.terraelectronica.ru/news/5188>. Дата обращения: 18.02.2020.
43. КОМПЭЛ – поставка компонентов и модулей для производителей электронной аппаратуры. – <https://www.compel.ru/lib/53954>. Дата обращения: 18.02.2020.

44. Radio-sale – Интернет-магазин радиодеталей и комплектующих. – <https://radio-sale.ru>. Дата обращения: 18.02.2020.
45. CHIPINFO – справочник по микросхемам и ежедневная лента новостей электроники. – https://chipinfo.pro/elements/acoustics/jack_pinout.shtml. Дата обращения: 19.02.2020.
46. Ронэс: дельные страницы. – <http://rones.su/techno/jack.html/comment-page-5>. Дата обращения: 16.01.2020.
47. Чернышев А. Печатный монтаж – классы точности печатных плат и современные технологии. Журнал. / А. Чернышев. – Изд-во «ТЕХНОСФЕРА», 06.2009. – 21 с.
48. Rexant – поставщик кабельной продукции и расходных материалов и для монтажа слаботочных систем. – <https://rexant-shop.ru/>. Дата обращения: 19.02.2020.
49. Блог, посвященный робототехнике. – <http://robotosha.ru/stm32/free-ide-stm32.html>. Дата обращения: 20.02.2020.
50. Tproger — издание о разработке и обо всём, что с ней связано. – <https://tproger.ru/translations/kotlin-vs-java-android/>. Дата обращения: 20.02.2020.
51. Аналого-цифровой преобразователь. – https://ru.wikipedia.org/wiki/Аналого-цифровой_преобразователь. Дата обращения: 24.02.2020.
52. Аналого-цифровое преобразование для начинающих. – <https://habr.com/ru/post/125029/>. Дата обращения: 24.02.2020.
53. Klein, A. Attacks on the RC4 stream cipher. / A. Klein. – Designs, codes and cryptography. — 2008. — Т. 48, № 3. — P. 269—286.
54. CRC-алгоритмы обнаружения ошибок. – http://embedded.ifmo.ru/embedded_old/ETC/REFERAT/crc/crc.htm. Дата обращения: 23.02.2020.
55. Мощность. – <https://ru.wikipedia.org/wiki/Мощность>. Дата обращения: 12.03.2020.

56. Электрический конденсатор. –
https://ru.wikipedia.org/wiki/Электрический_конденсатор. Дата обращения:
12.03.2020.
57. Фергюсон, Н. Практическая криптография. / Н. Фергюсон, Б. Шнайер. – М.:
Изд-во «Вильямс», 2005. – 432 с.
58. Anusuya, S. J. Data Transmission by Ceaser Cipher Wheel encryption using Lifi.
/ S. J. Anusuya. – International Journal of Advance Research, Ideas and
Innovations in Technology. — 2018. – V. 4, № 2. – P. 6.
59. Сайт о криптографии. – <http://kriptografea.narod.ru/XOR.html>. Дата
обращения: 21.02.2020.
60. AudioRecord. –
<https://developer.android.com/reference/kotlin/android/media/AudioRecord/>.
Дата обращения: 10.03.2020.
61. Основные блоки для составления схем алгоритмов. – [https://prog-
cpp.ru/block-schema/](https://prog-cpp.ru/block-schema/). Дата обращения: 10.02.2020.
62. Функциональное тестирование. – <http://getbug.ru/funktsionalnoe-testirovanie/>.
Дата обращения: 28.04.2020.