

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»

Кафедра Электроника и робототехника

«ДОПУЩЕН К ЗАЩИТЕ»

Зав. кафедрой _____

(ученая степень, звание, Ф.И.О.)

« _____ »

201__ г.

(подпись)

ДИПЛОМНАЯ РАБОТА

На тему: Разработка системы управления объектами по технологии IoT

Специальность 5В041600 – Приборостроение

Выполнил (а) Искаков Ермекжан Азгарович Группа ИС-15-4
(Ф.И.О.)

Научный руководитель _____

доц. к.т.н. Бегежиков Б.Е.
(ученая степень, звание, Ф.И.О.)

Консультанты:

по экономической части:

к.э.н., доцент Бекмишва А.И.
(ученая степень, звание, Ф.И.О.)

ДЧ « 21 » мая 2019 г.
(подпись)

по безопасности жизнедеятельности:

д.х.н., профессор Бухардыков Н.Г.
(ученая степень, звание, Ф.И.О.)

НГ « 31 » 18 2019 г.
(подпись)

Нормоконтролер: _____

к.т.н., доцент Чылабаев Т.О.
(ученая степень, звание, Ф.И.О.)

ТО « 7 » 06 2019 г.
(подпись)

Рецензент: _____

(ученая степень, звание, Ф.И.О.)

« _____ »

201__ г.

(подпись)

Алматы 2019

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»

Институт космической инженерии и телекоммуникаций

Кафедра Электроника и радиотехника

Специальность 5В071600 – Приборостроение

ЗАДАНИЕ

на выполнение дипломного проекта

Студенту Исмакову Ермебулату Адамеревичу
(Ф.И.О.)

Тема проекта Разработка системы управления объектами по технологии IoT

Утверждена приказом по университету № 124 от «26» октября 2018 г.

Срок сдачи законченного проекта «7» июня 2019 г.

Исходные данные к проекту (требуемые параметры результатов исследования (проектирования) и исходные данные объекта):

1. Одноплатный микрокомпьютер Raspberry Pi 3 B+
2. Запчасти и элементы управления объекта
3. Интернет подключение

Перечень вопросов, подлежащих разработке в дипломном проекте, или краткое содержание дипломного проекта:

1. Ознакомление с прибором
2. Выбор соответствующей платформы
3. Принципиальная схема подключения
4. Выбор и загрузка программ на микрокомпьютер
5. Тестирование системы

Перечень графического материала (с точным указанием обязательных чертежей):

- 1) Титульный лист
- 2) Актуальность и цель
- 3) Структура и схема разработки управления объектов
- 4) Детали и элементы для реализации прибора
- 5) Безопасность жизнедеятельности
- 6) Экономический раздел

Основная рекомендуемая литература:

- 1) Петин В.А. «Arduino и Raspberry Pi в проектах»
Тет 2-е издание, СПб: БХВ-Петербург, 2019, 432 с. Электроника
- 2) Светодиоды и подключение LР01/LР02.
Техническое описание
- 3) Raspberry Pi Cookbook: Software and hardware problems and solutions, 12.2013, Simon Monk

Консультации по проекту с указанием относящихся к ним разделов проекта

Раздел	Консультант	Сроки	Подпись
БЖД	Приходько И.Г.	13.02-31.05.19	
Эконом. часть	Веников Д.М.	01.02-25.03	
Технолог. часть	Вайкент В.В.	01.02-25.03	
Конструкторская	Вайкент В.В.	01.02-25.03	
Програм. часть	Вайкент В.В.	01.02-25.03	

График

ПОДГОТОВКИ ДИПЛОМНОГО ПРОЕКТА

[illegible]

Дата выдачи задания « 1 » февраля 2019 г.

Заведующий кафедрой


 (Подпись)
 
 (Ф.И.О.)

Научный руководитель
проекта

 (подпись) Баднер В. (Ф.И.О.)

Задание принял к
исполнению студент

Искаков (подпись) (Искаков Е.А. (Ф.И.О.))

Андатпа

Бұл дипломдық жобада объектімен бақылау негізінде жүйе құрастырылып, сәйкес платформа таңдалып және ұялы телефонмен бақылау көрсетілген. Басқа аналог платформалар арасында анализ жүргізе отырып, ерекше тандау Raspberry Pi нұсқауында таңдалды. Жоба тәжірибелік үлгі ретінде көрсетілген, сонымен бірге техникалық шешім техникалық-экономикалық негіздеме бойынша расталған.

Аннотация

Дипломный проект посвящен разработке актуальной системы управления объектом с выбором подходящей платформы и управление с помощью сотовой связи, имея свободный доступ к интернету. Среди других аналогичных платформ был сделан анализ и тщательно был выбран вариант на основе Raspberry PI. Данный проект реализован в виде опытного образца, техническое решение которого подтверждено технико-экономическим обоснованием.

Annotation

The diploma project is devoted to the development of an actual object management system with the choice of a suitable platform and management via cellular communication, having free access to the Internet. Among other similar platforms, an analysis was made and a carefully selected option based on the Raspberry PI. This project is realized in the form of a prototype which technical solution is confirmed with the feasibility study.

СОДЕРЖАНИЕ

Введение	7
1 Анализ существующих управлений систем	
1.1 Обзор системы управления объектов	8
1.2 Выбор актуальной платформы	9
1.3 Использование SSh настроек для управления	10
1.4 Этапы создания систем управления объектов	13
2 Разработка системы управления	
2.1 Постановка задачи	16
2.2 Выбор интернет модуля и настройка	18
2.3 Принципиальная схема устройства	21
2.4 Возможность управления смартфоном	22
3 Программное обеспечение	
3.1 Среда разработки	25
3.2 Листинг программы	27
3.3 Алгоритм работы программы	29
4 Безопасность жизнедеятельности	33
4.1 Анализ опасных и вредных факторов	33
4.2.1 Расчет искусственного освещения	36
4.3 Расчет уровня шума производимого персональным компьютером на работе	40
5 Технико-экономическое обоснование	42
5.1 Расчет капиталовложений на оборудование	42
5.1.1 Расчет капиталовложений на оборудование для первого варианта реализации проекта	43
5.1.2 Расчет капиталовложений на оборудование для второго варианта	43
5.2 Расчет капиталовложений на разработку программного обеспечения	44
5.3 Расчет эксплуатационных затрат	51
5.4 Оценка экономической эффективности реализации проекта	53
Заключение	54
Список литературы	55

Введение

Актуальность работы основана в разработке системы управления различными объектами с возможностью управлением смартфоном. Реализованная система автоматизационного управления создана с современным термином для создания технологии управления систем объектами «IoT – интернет вещей», что соответствует принятым современными вещами разработки электронных систем управления.

Проект основан на управлении объектом в предприятии, который способствует упрощению в работе и уменьшить число работников.

В эпоху развития технологии, данный проект даёт возможность управлять объектами с любой точки мира, одним из важных факторов – интернет, нужно иметь свободный доступ к интернету и с помощью авторизации и адреса основного устройства даёт возможность управлять всеми процессами одним кликом. Основная часть проекта основана на кодировании и управление прибором – это доказывает продвижение приборостроение на новый уровень развития и создавая новые технологии непосредственно алгоритмами кодов. Проект несёт смысловую нагрузку «DIY – Do It Yourself/Сделай Сам»), имея знания по кодированию и основной микрокомпьютер Raspberry Pi, можно создавать проекты, которые не требуют проводов, других либо модулей управления (Bluetooth, ESP), так как можно создавать приборы, которые управляются интернетом.

Система в отличии от других платформ (Banana PI, Orange PI), работает автономно целый рабочий день, только требуется установить кондиционеры охлаждения и не требует особых вложений, это и отличает от других платформ, а также стабильность работы и надежность.

В проекте используется модель Raspberry PI 3 B+, на всех моделях можно создать данный проект, отличие лишь в некоторых вещах: в подключении интернета, число GPIO портов, наличие модулей Wi-Fi, Bluetooth.

Основное управление происходит с помощью IP-управления, можно использовать любой компьютер, смартфон, любое устройство, имеющий свой динамический IP номер и выход в сеть. Систему можно улучшить, если приобрести статический IP адрес, так как он остается у владельца и никогда не подлежит к изменению, но учитывая оплату, динамический IP адрес ничем не хуже и не составит труда за минуту вычислить IP адрес.

1 Анализ существующих управлений систем

1.1 Обзор систем управления объектов

Разработка управления объектами с каждым годом развивается и на данный момент управление является прозрачным, чтобы упростить контроль и управление процессами. На данной стадии представлена управление с помощью IP-адреса объекта, имея доступ к сети с устройств, возможно управлять процессами на разных дистанциях.

В первую очередь, данный проект позволит упростить управление на предприятиях, так и в домашних условиях: возможность дистанционно включить обогрев, электропитание, управлять гаражом, всеми электрическими приборами и другими объектами соответственно.

Основным элементом «мозгом» проекта является микрокомпьютер серии Raspberry Pi, который может автономно работать на долгое время, гарантируя стабильность и отсутствие погрешностей, при правильном программировании для управления объектов. Нюансом является то, чтобы запустить другую иную систему – необязательно иметь лишний микрокомпьютер, достаточно загрузить с помощью карты памяти SD на Raspberry Pi и запустить систему управления иными объектами. Все модели поддерживают стабильность и возможность управления, но последние модели упрощены и немного ускорены, дабы система работала быстрее, да и оснащены беспроводным модулем Wi-Fi, а также современным модулем Bluetooth для соединения с другими устройствами.

Данному проекту можно добавить разные датчики, такие как: датчик температуры и влажности, они будут работать автономно при включении прибора, да и система управления Raspberry Pi настроена на асинхронное обновление, а именно не требует вмешательства пользователя, кликая каждый раз – прибор будет обновляться каждую секунду, показывая все данные на странице прибора настроенным IP-адресом.

Сфера надежности тоже присутствует в современных систем управления, достаточно настроить объект под соответствующим параметром защиты, самый примитивный – иметь логин и пароль для входа и управления объектом.

Любая платформа, контроллер не может работать неделями круглосуточное время, для поддержки системы, можно добавить вентиляторы, чтобы уменьшить температуру нагрева и держать максимальный уровень производительности, чтобы управлять объектами без препятствий. Каждая система имеет свой уникальный IP-адрес, является они динамичными, при подключении на разные объекты не составит труда вычислить данный адрес и приступить к управлению, но имея услугу у провайдера – есть возможность приобрести статический IP-адрес, который присуждается прибору и дальнейшее управление не будет требовать вычисления адреса.

1.2 Актуальная платформа

На данный момент развития технологий существуют множество видов одноплатных компьютеров, все они имеют различные функции, особенности и недостатки. Выбор сложился на одноплатном компьютере Raspberry Pi 3b+ [3]. Основные ключевые особенности заключаются в стабильности работы, модули беспроводного интернета, модуля Bluetooth, удобные контакты подключения приборов. Данная модель поддерживает операционную систему Windows 10 IoT, который способствует созданию новых проектов IoT (Internet of Things).

Raspberry Pi ([ˈrɑːzbəri paɪ], Рáзбери пáй) — одноплатный компьютер размером с банковскую карту, изначально разработанный как бюджетная система для обучения информатике, но позже получивший более широкое применение. Разрабатывается Raspberry Pi Foundation. Всего за пять лет было продано более 12,5 миллионов устройств Raspberry Pi [11].

Аппаратная платформа является ARM11, в момент создания прибора была выбрана операционная система Raspbian, который удовлетворяет все функции, которые необходимы для создания прибора. Характеристики в оперативной памяти в зависимости от модели являются разными: 256 (Model A), 512 (Model B/B+/Zero), 1024 (Model 2B/3B/3B+) МБ интегрирована в CPU (часть резервируется видеокартой). Удобством выступает то, что память устанавливается пользователем и наличие несколько карт-памяти упрощает работу пользователя и способствует управления различными операционными системами для управления объектами, достаточно одного одноплатного компьютера Raspberry Pi. Данная модель также имеет аудиовыход, которая интегрирована в CPU [1]. Электропитание от microUSB на 5 В, от 700 мА, что означает низкого потребления электроэнергии. При приобретении комплекта отсутствуют периферии, но проблема решаема: достаточно использовать другие устройства от персонального компьютера, они подходят для данного одноплатного компьютера. В дальнейшем развитии могут выступить различные интерфейсы, которые нет у других одноплатных компьютеров: HDMI, USB, видео RCA, Stereo Jack 3.5 мм, Ethernet, UART, GPIO, JTAG, SPI, I2C, DSI, CSI, Wi-Fi. Вес одноплатного компьютера Raspberry Pi составляет всего 45 г, разработчик CPU является Broadcom, в данной модели для реализации проекта тактовая частота имеет 1,4 ГГц, что достаточно производительно для управления различными процессами при ходе создания прибора [4].

Вывод видеосигнала возможен через композитный разъём RCA или через цифровой HDMI-интерфейс. Отличительной чертой является наличие GPIO портов (General purpose input/output), в данной модели их количество составляет 40-пиновые разъемы, а в предыдущих моделях их количество – 26.

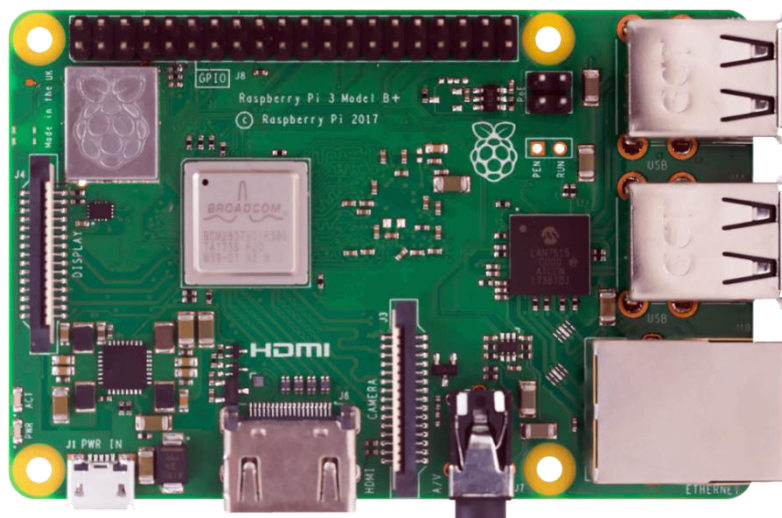


Рисунок 1.1 – Одноплатный компьютер Raspberry Pi 3 b+

1.3 Использование SSh настроек для управления

Raspberry Pi – одноплатный компьютер, с которым многие пользователи взаимодействуют удаленным доступом. Это очень удобно, так как не имеет нужды каждый раз, когда требуется что-то включить или изменить какую-то конфигурацию, подсоединять микрокомпьютер к монитору или телевизору. Есть несколько способов, посредством которых является возможностью соединиться с Raspberry Pi. Один из действенных – это использование Secure Shell.



Рисунок 1.2 – Представление о SSh

SSH – это основной протокол, который обеспечивает и поддерживает соединение по сети (интернет или локальной) между двумя удаленными устройствами. По нему можно передавать почти любую информацию, такие как: файлы, аудио и видео в потоке, команды и различные действия. У данного протокола есть два важных преимущества. Во-первых, он умеет на стадии передачи сжимать данные. Во-вторых, он создает зашифрованный туннель, что обеспечивает безопасность. Если говорить об удаленном доступе, то при подключении к Raspberry Pi через SSH на одноплатном компьютере запускается терминал (командная строка), основная работа с которым осуществляется на другом персональном компьютере. Подключаться через Secure Shell возможно, находясь всего в 1-м метре от Raspberry Pi, а также и на другом конце Земного шара.

Настройка SSH зависит от разных дистрибутивов, на данном приборе установлен Raspbian – популярная адаптированная версия Debian и его настройка не составит труда. Прописав команду в терминале «sudo raspi-config», то перед пользователем откроется полная конфигурация:

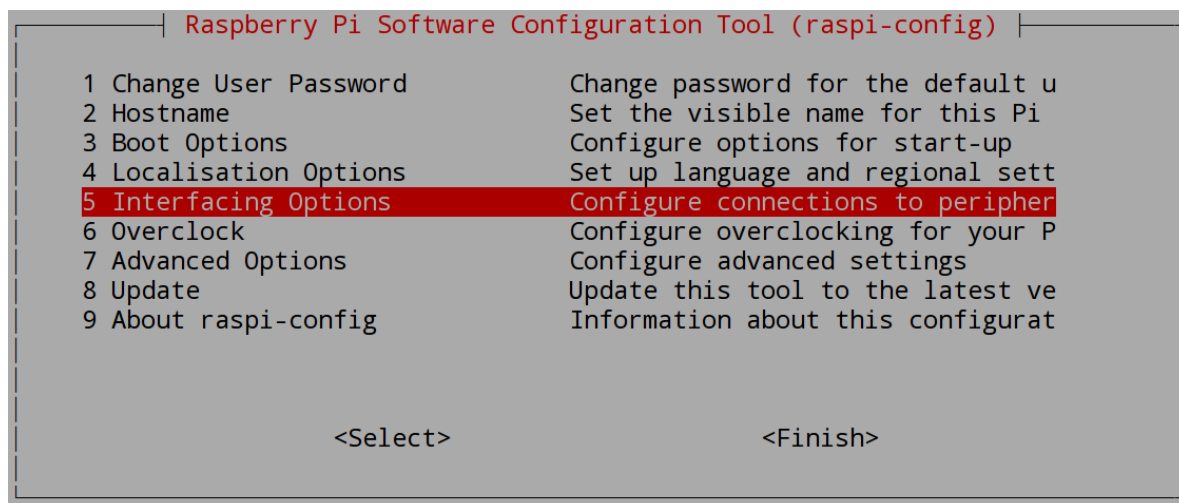


Рисунок 1.3 – Основная конфигурация Raspberry Pi

Перед пользователем стоят различные параметры, выбрать следует пункт Interfacing Option (он идет пятым), а в следующем пункте выбрать Enable SSH, затем нажать ввод и система будет настроена под данный протокол, после перезагрузки система уже имеет потенциал к подключению удаленным доступом с различных устройств.

Теперь необходимо рассмотреть вопрос относительно того, как осуществляется подключение по SSH к Raspberry Pi. Так как многие владельцы Raspberry пользуют операционную систему Windows, то и рассматриваться будет вариант соединения с Raspberry Pi на данной платформе. Линукс-пользователям не составит труда подключения.

В первую очередь, что понадобится — специальное созданное приложение, так как в Windows нет своей программы для работы по данному протоколу Secure Shell. Самым эффективным и популярным из них является приложение PuTTY — это полностью функционирующий SSH-клиент. Следует скачивать данный продукт с официальных сайтов, так как есть риск заражения компьютеров, а то и два сразу.

Подключение достаточно простое, нужно указать адрес Raspberry Pi в поле «Имя хоста», ввести номер порта одноплатного компьютера по стандартному стоит порт 22, в типе подключения следует выбрать SSH подключение.

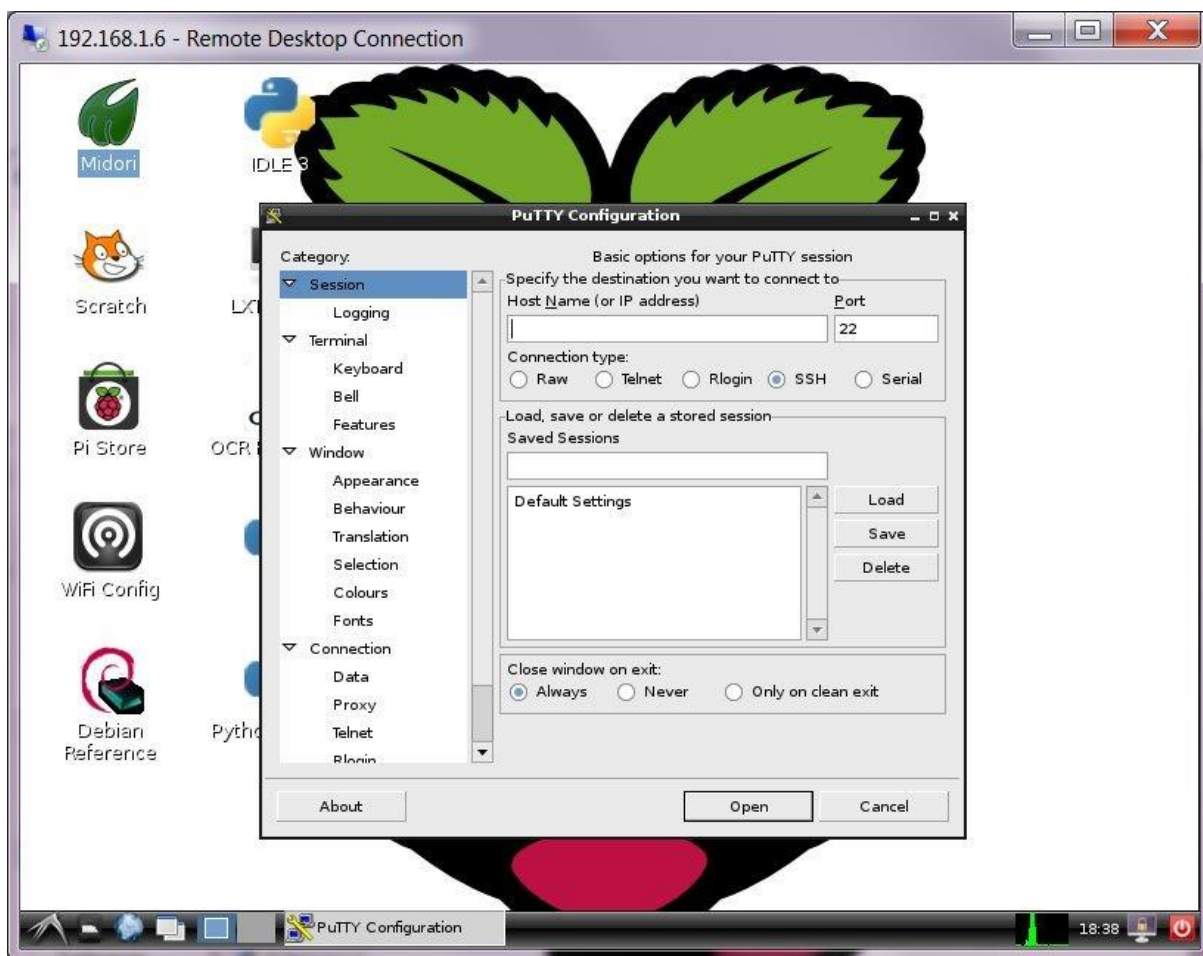


Рисунок 1.4 – Конфигурация подключения

Если всё заполнено верно, то следует нажать на кнопку «соединиться» и после нажатия отобразится командная строка Windows и через неё можно будет взаимодействовать с Raspberry Pi, а также будут доступны все функции и полноценные команды, которые не поддерживают CMD. Это еще не всё, для детальной проработки создании прибора — можно использовать графическое управление через VNC, который предоставляет все возможности, что и при физическом подключений с Raspberry Pi [13].

1.4 Этапы создания систем управления объектов

Основные настройки базовых параметров Raspberry Pi выполнены, следует обзавестись элементами для создания наглядности управления объектом, а именно достаточно приобрести обычный светодиод, которому достаточно дать питание в 3,3 В и существует как раз такой PIN для подключения диода в одноплатный компьютер с GPIO контактами. Для реализации подключения к контактам GPIO следует изучить предназначение всех PIN-подключений:

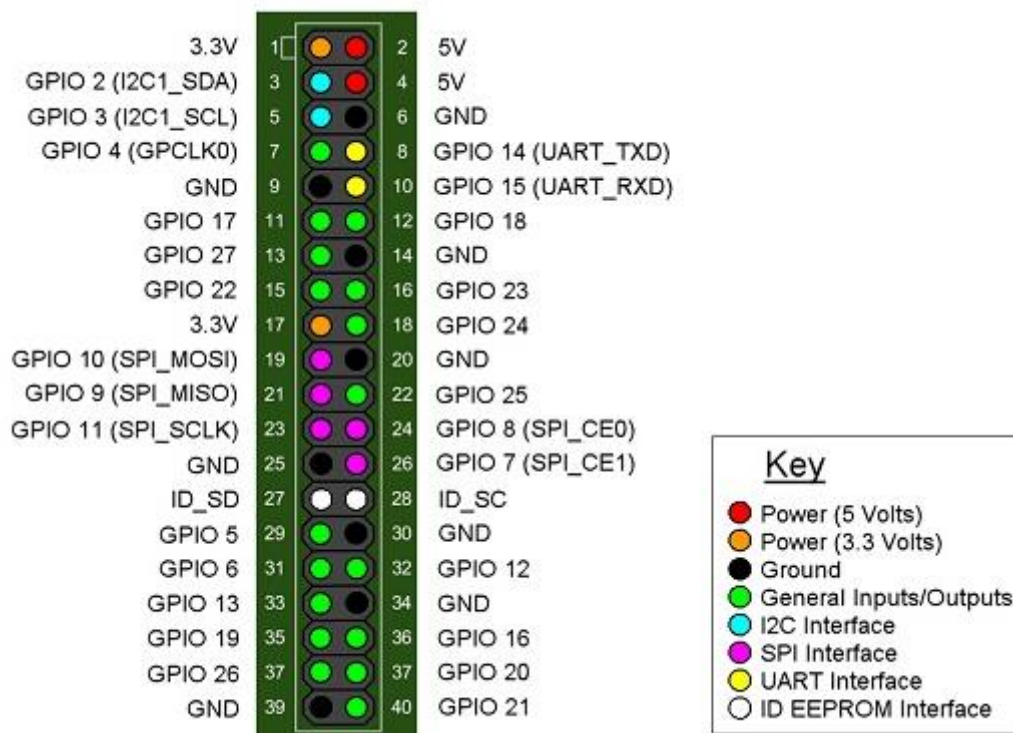


Рисунок 1.5 – Схема назначений GPIO портов.

На приборе для подключения диода используется PIN 32 – назначение которого можно подключить любой прибор, и для диода был выбран PIN 39, что способствует подключения заземлению для диода.

Для удобства полноценного управления были подключены дополнительные периферии – миниатюрный экран для Raspberry Pi 3.5 LCD, который поддерживает сенсорное управление стилусом:



Рисунок 1.6 – LCD экран для Raspberry Pi

Удобство LCD-экрана состоит в том, что подключается напрямую к GPIO контактам, но своим объемом преграждает другие контакты GPIO, но данная модель имеет 40 контактов, а для реализации системы управления объектами достаточно использовать 2 из них.

Следующая периферия Mini Keyboard сразу же заменяет основную полноценную мышь и компьютерную клавиатуру, которая работает от аккумулятора и способствует управлению по радиоканалу, имеет особенности: имеется touchpad, полная клавиатура, полноценный контроль иными приборами, как смарт-телевизор, смартфон, приборы на основе Android, имеют различные функции управления для упрощения.



Рисунок 1.7 – Мини-клавиатура для Raspberry Pi

После дополнительных подключений периферии к Raspberry Pi, одноплатный компьютер стал более компактным и не требует подключений больших мониторов и значительных размеров периферии, а это значит, что мобильность прибора улучшена и стала компактнее.

В данной работе рассматривается система автоматического управления-объектом с помощью IP-управлением.

Для начала рассмотрим управление на основе: GPIO портов, на самом деле виды подключения не так малы, можно использовать на основе ESP-86 (Wi-Fi управление), через UART, SPI и тому подобное. Мы выбрали порты GPIO на Raspberry PI. Общие назначения портов представлены снизу:

2 Разработка системы управления

2.1 Постановка задачи

При полной подготовке одноплатного компьютера и настроек основных подключений следует понять, что реализация интернет-управления наполовину выполнена.

В мире микроэлектроники, соответствующей задачей можно считать управление светодиодом. А именно, включение и выключение — мигание. В этой работе как раз выполняется подключение светодиода к Raspberry Pi и составлением особой программы на языке python, заставляющей этот светодиод работать.

Одно из главных преимуществ как и было написано ранее у Raspberry Pi — это наличие выводов общего назначения (General Purpose Input/Outputs). GPIO — это группа контактов, которыми возможно управлять с помощью программы. Причем управление это может быть совсем уж простым и легким, например, включение/выключение светодиода. Либо весьма сложным — обмен данными с периферийными устройствами по специальным созданным протоколам.

В современных терминах цифровой электроники IoT, управлять — значит менять на выводе уровень напряжения. Иными словами выражаясь, все что можно сделать с помощью программы — это соединить желаемый контактный вывод либо с контактом питания (+3.3 В), либо с заземлением (Gnd). Для полноценного прибора изображена принципиальная схема.

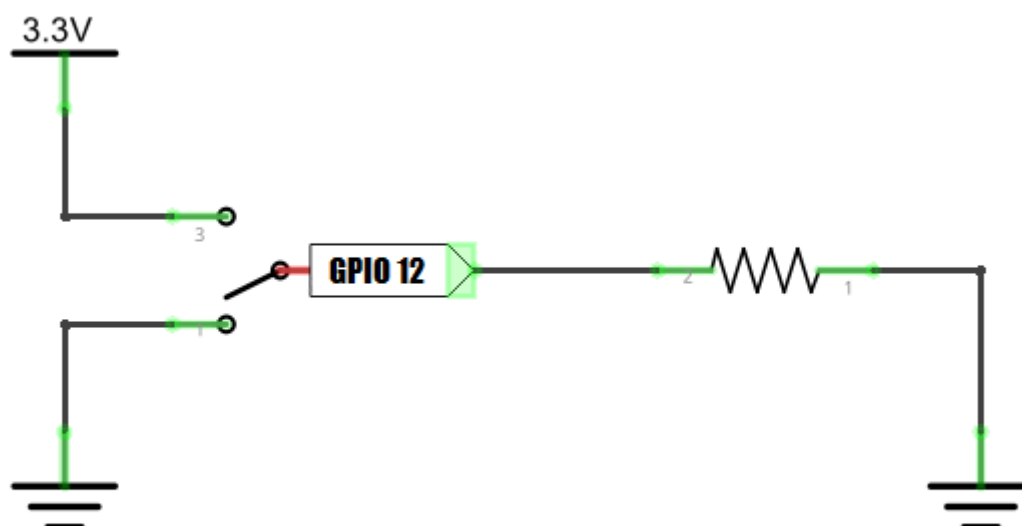


Рисунок 2.1 – Принципиальная схема соединения

На схеме подключен резистор, соединенный справа с заземлением — это основная нагрузка. Вместо резистора может быть светодиод, реле и т.п. Вывод

GPIO12 и переключатель прямо за ним символизируют внутреннее устройство каждого вывода общего назначения контактов. Реализуется это следующим образом.

Если в программе подать на вывод №12 истину True, то Raspberry Pi соединит этот вывод с питанием +3.3В, и через резистор будет проходить ток. Если же мы передадим ложь — False, то одноплатный контроллер соединит вывод №12 с заземлением, и с обоих концов от резистора окажутся равные потенциалы — ток никуда не уйдёт. То есть, с помощью программы как бы осуществляется щёлканием невидимым переключателем, то вверх, то вниз.

Когда косается составления программы для мигания светодиодом, то можно узнать, как передавать истину и ложь на выводы общего назначения контактов Raspberry Pi.

Светодиод — это устройство, которое непосредственно представляет собой полупроводниковый прибор, способный излучать свет при пропускании через него электрического тока в прямом направлении (от анода к катоду). Ниже показана схема типичного светодиода с линзой.

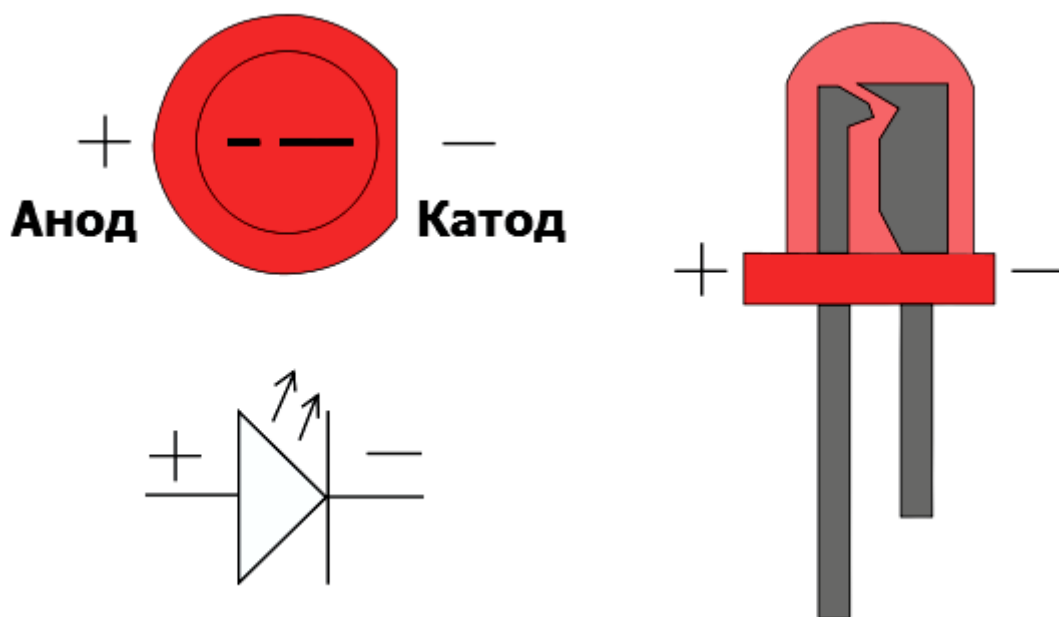


Рисунок 2.1 – Свойства светодиода

Для того чтобы правильно внедрить и включить светодиод в электрическую цепь, нужно отличать катод от анода. Сделать это можно по двум признакам:

- анод светодиода имеет более длинный проводник;
- со стороны катода, корпус светодиода немного срезан.

В современной микроэлектронике используются миниатюрные светодиоды для поверхностного монтажа. Такие индикаторы, например, имеются на Raspberry Pi для информирования пользователя о состоянии основной системы.

2.2 Выбор интернет модуля и настройка

Данная модель платформы Raspberry Pi оснащена беспроводным модулем Wi-Fi и Bluetooth, что упрощает подключение и добавляет новый метод управления процессами. Работу процессов можно реализовать через Ethernet подключение или же дистанционно через Wi-Fi, пользователь может выбрать один из двух вариантов при наличии одной из них.

Дальнейшее управление зависит от соединения Raspberry Pi к провайдеру с помощью TCP протоколов. Для начала, необходимо узнать адрес и зайти на личную страницу, это «192.168.1.2» и зайти на личную страницу личного провайдера и следует добавить устройство Raspberry Pi в список доступных для управления интернетом.

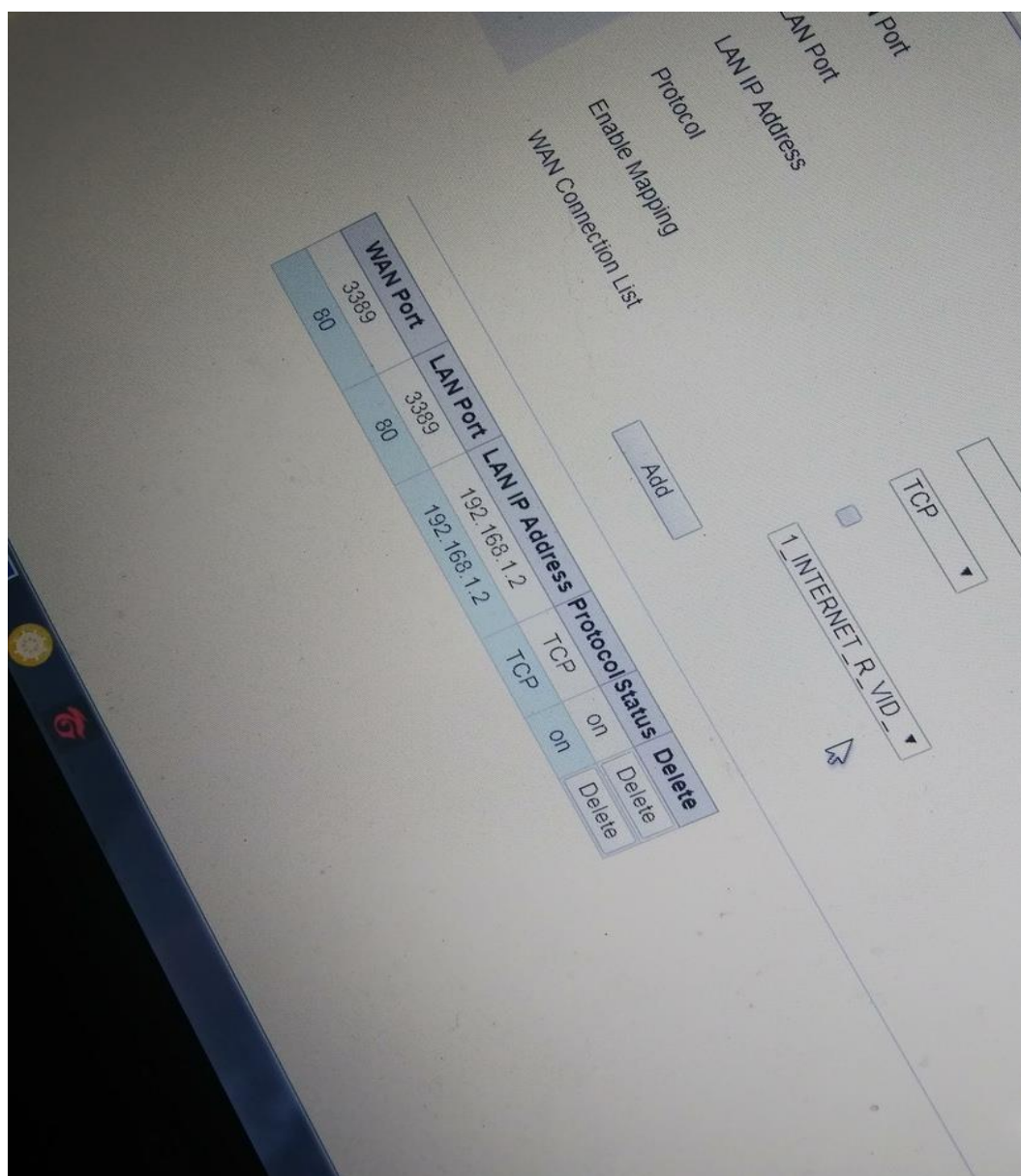


Рисунок 2.2 – Добавление Raspberry Pi в список доступных приборов

На данном этапе добавлено устройство под двумя портами TCP, для добавления нужно установить порт по умолчанию, их 2: 3389 и 80. Они позволяют соединить сети и при доступе IP-адресу возможно было подключаться и выполнять работы по управлению объектами.

Далее, если проверить подключенные устройства, то будет видно, что данное устройство успешно подключено и можно дальше работать.

Wireless Settings		Never_Give_Up	
Network Name		d0-54-2d-01-d4-e0	
Access Point			

Local Devices				
Connection Type	Device Name	IP Address	Hardware Address	IP Address Allocation
Wireless	raspberrypi	192.168.1.2	b8:27:eb:a9:c2:77	DHCP(Private)
Wireless	iPhone-Era	192.168.1.4	68:d9:3c:30:65:c0	DHCP(Private)
Wireless	android-72094d69ce38b058	192.168.1.6	b4:f1:da:bc:86:91	DHCP(Private)
Wireless	SHADY-??	192.168.1.3	60:d8:19:27:d5:5e	Manual(Private)

Refresh

Рисунок 2.3 – Добавленное устройство и её состояние

Дальнейшее действие укажет про подключение к основной сети и требуется проверить порт, что является стабильным и действие в будущем не составит никакой сложности, поэтому, на терминале ввод пишем «netstat -an», данная команда отобразит нам наш порт (рисунок 2.4).

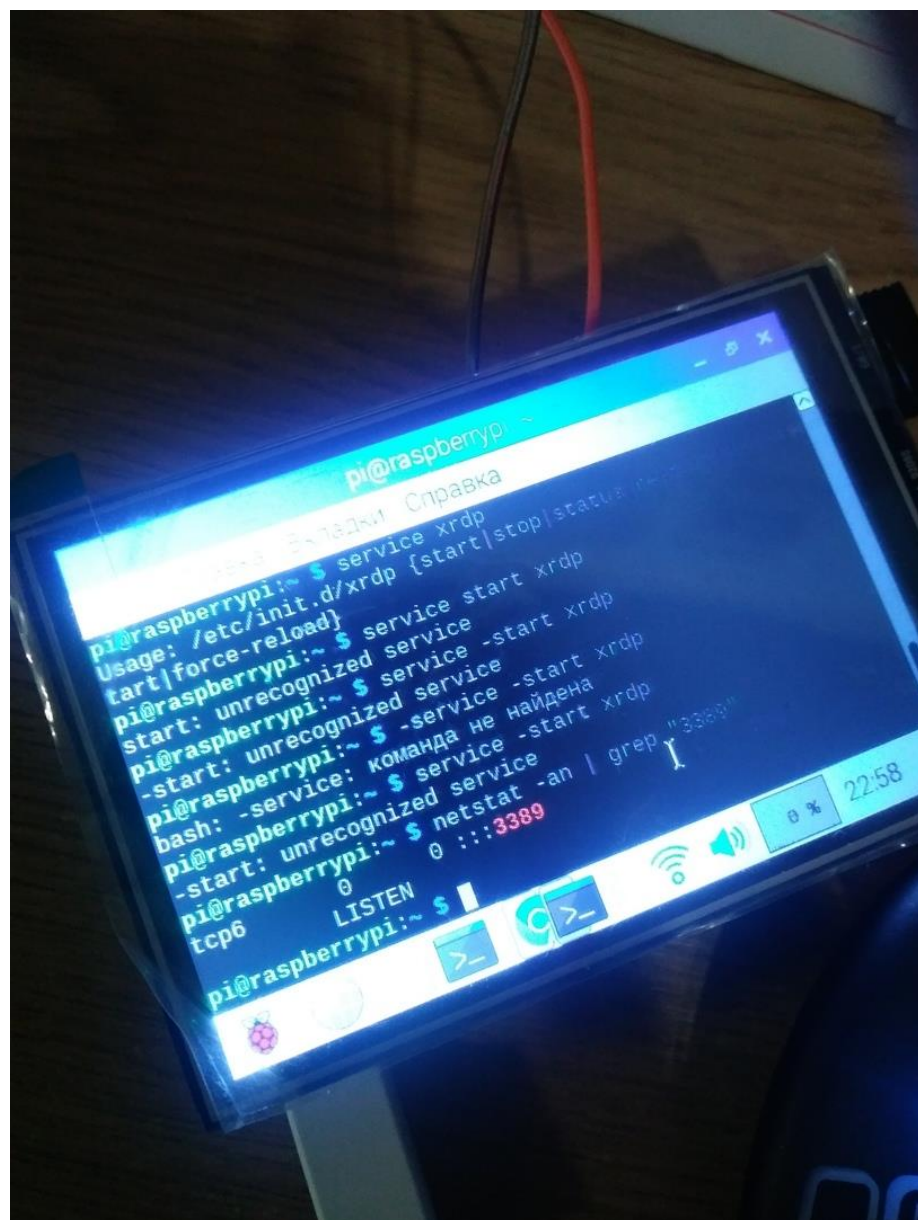


Рисунок 2.4 – Проверка подключения

Далее, видно, что порт работает по принципу «3389» и управление с помощью IP-подключения вполне реальна.

2.3 Принципиальная схема устройства

Чтобы включить светодиод, потребуется подключить его к одному из цифровых выводов Raspberry Pi. Пусть это будет вывод №12.

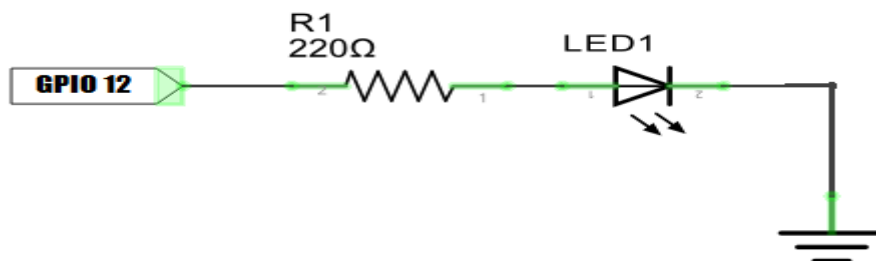


Рисунок 2.5 – Принципиальная схема

А теперь следует рассмотреть, как будет выглядеть внешний вид макета и общее соединение:

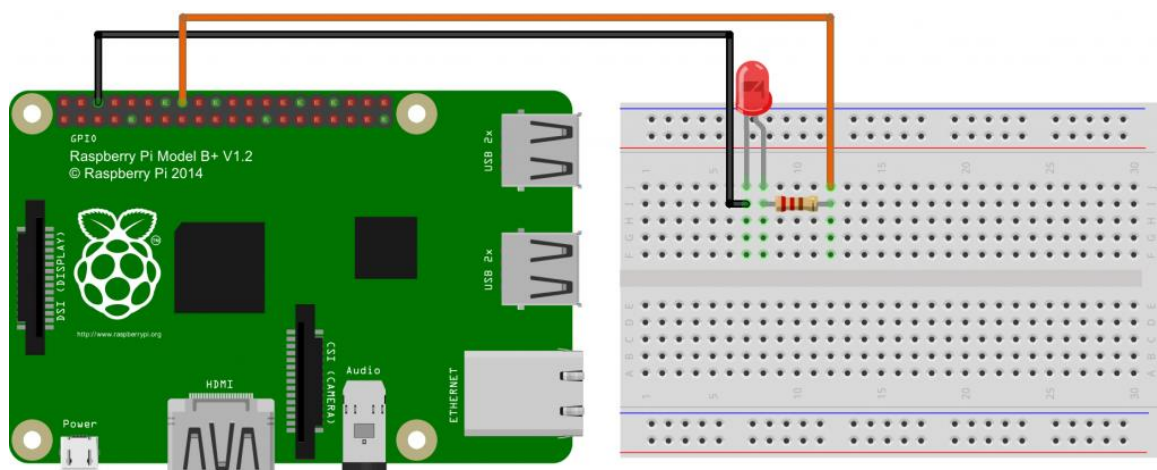


Рисунок 2.6 – Вид макета и соединение с Raspberry PI

В схеме, помимо самого светодиода имеется резистор. Зачем он нужен? Дело в том, что если подключить светодиод напрямую к источнику напряжения контактов 3.3 Вольт (а именно такое рабочее напряжение у Raspberry Pi), то светодиод непременно выйдет из строя и реализовать проект не получится.

Для правильной и соответствующей работы светодиода нужно строго выдерживать рабочий ток и напряжение. Например, обычный светодиод красного цвета имеет рабочий ток 20 мА, при напряжении от 1.8 до 2 Вольт [19]. Резистор позволяет снизить напряжение в цепи и установить соответствующий ток. Такой резистор еще называют токозадающим в электронике [17].

Чтобы рассчитать основной номинал токозадающего резистора следует воспользоваться законом Ома:

$$I = U/R. \quad (1.1)$$

Сила тока, проходящая через некий элемент равна отношению напряжения, приложенного к этому элементу на его сопротивление.

Выразим из этой формулы сопротивление R:

$$R = U/I. \quad (1.2)$$

Как уже было замечено, Raspberry Pi работает с напряжением 3,3 Вольта. Значит, чтобы на светодиоде осталось 2 Вольта, необходимо куда-то убрать лишние 1,3 Вольта. Резистор выполнит это предназначенное дело.

Получается, необходимо подобрать сопротивление резистора таким образом, чтоб он убрал лишние 1,3 Вольта, при токе 20 мА. Подставим эти числа в полученную формулу:

$$R = 1,3V/20mA = 1,3V/0,02 \text{ Ом} = 130/2 = 65 \text{ Ом}.$$

Таким образом, в Raspberry Pi токозадающий резистор для красного светодиода имеет номинал 65 Ом. Именно его и должны установить в схему.

Если под рукой нет резистора именно на 65 Ом, то можно использовать любой другой большего номинала, например, 100 Ом или даже 200 Ом. Просто светодиод будет гореть чуть менее ярко.

2.4 Возможность управления смартфоном

На данном проекте включается возможность управления процессами объекта через смартфон. Для подключения достаточно подключиться к общей сети, которая настроена по ходу разработки программы и ввести IP-адрес в браузерную строку и появится полноценный интерфейс как и на персональном ПК управление.

Алгоритм подключения может быть разной, если в помещении занят компьютер или не имеется свободного устройства, то можно использовать смартфон в качестве управления процессами, но следует подключиться к сети, которая подключена непосредственно Raspberry Pi, если требуется управлять процессами с другой точки мира, то следует добавить адрес управления в список доверенных лиц и можно управлять таки через смартфон, таки через ПК с другой части мира [12].

Альтернативный способ управления Raspberry Pi может сложиться через приложение IBM Bluemix IoT Service, который позволяет управлять проектами IoT непосредственно через смартфон на платформе Android. Все подключенные устройства на Raspberry Pi можно контролировать через приложение.

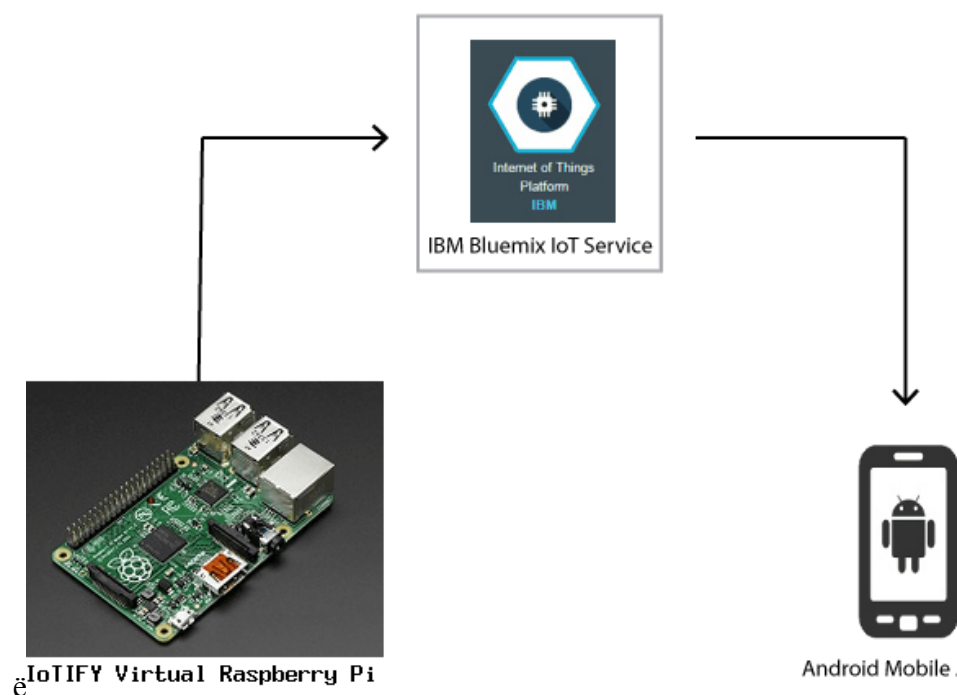


Рисунок 2.7 Приложение для управления

Пользователь решает какой метод является эффективным в управлении процессами объектов, можно осуществить лишь через интерфейс браузера, что является нормальным и не способствует утрате большей части зарядов аккумулятора, но может быть неудобным интерфейс управления, так как полная версия основана на ПК, но проблема может быть решена, если разработать мобильную версию сайта, что является трудоёмкой работой. Если управление уложена на нескольких кнопок, как на данном проекте, то создавать мобильный интерфейс не имеет смысла.

Сейчас имеет популярность программа Telegram и особенность в том, что можно создать телеграм-бота, созданного в управлении объектами и можно соединить с Raspberry Pi, управление может быть командной, аудио-управлением, графическим. Как бы ни было, управление состоит из простых команд, чтобы управлять процессами, данный метод удобен для смартфона и в целом для персонального ПК, эффективность также заключается в удобстве управления, можно задать контроль управления доверенным лицам, так и чисто для административного состава.

На данный момент подключение со смартфона выглядит так, как представлена на рисунке снизу. Так как IP-динамическое, постоянно адрес меняется при подключении к сети, можно приобрести у провайдера постоянный статический адрес, но это будет дополнительной платой и узнать адрес своего прибора не имеет особой сложности, достаточно прописать команду «ifconfig» и в настройках будут все адреса подключаемых к сети.



Рисунок 2.8 – Подключение через смартфон

3 Программное обеспечение

3.1 Среда разработки

Есть различные одноплатные компьютеры и каждые из них лучше взаимодействуют с различными средами разработки, а для Raspberry Pi наилучшим вариантом является современная среда разработки Python, она помогает в разработке новейших программ для управления процессами и всеми разработками для Raspberry Pi. Тонкость работы и понятный интерфейс обеспечивает полноценную работу и функцию для пользователя и имея основные знания в программировании, можно создать данный прибор для управления объектами.

Raspberry Pi – самая мощная на данный момент платформа для обработки информации, полученных от различных устройств. Этот одноплатный компьютер может стать основой для создания систем умного дома и устройств распознавания образов. Наиболее удобной поэтому для системы Raspbian является python. Python – один из самых популярных и используемых языков программирования. Изначально, программируемый язык Python должен был быть в роли основного языка программирования для Raspberry Pi. В одноплатном компьютере Raspberry Pi он используется для управления с датчиками через GPIO при помощи специальной библиотеки. Пакет Python уже установлен в операционной системе Raspbian в двух версиях – 2 и 3. Каждый элемент в системе Raspbian перед именем имеют приставку «python-».

Командой «`sudo apt-get update`» и «`sudo apt-get install python3-pi-camera`» можно обеспечить полноценную установку программного языка Python в лучшем виде, так как благодаря такой установке можно легко обновиться до последней версии.

Библиотека RPi.GPIO создана специально для управления портами GPIO на языке python. Эти порты дают возможность собирать данных с датчиков, управлять двигателями, ключами, реле и прочими устройствами с помощью программы. Далее следует зайти в командную строку (терминал) в Raspberry Pi и написать команду «`python setup.py build`» и далее выйдет на экране отчет о выполнении.

```

File Edit Tabs Help
build/temp.linux-armv7l-2.7/source/soft_pwm.o
arm-linux-gnueabi-gcc -pthread -DNDEBUG -g -fwrapv -O2 -Wall -Wstrict-prototyp
es -fno-strict-aliasing -D_FORTIFY_SOURCE=2 -g -fstack-protector-strong -Wformat
-Werror=format-security -fPIC -I/usr/include/python2.7 -c source/py_pwm.c -o bu
ild/temp.linux-armv7l-2.7/source/py_pwm.o
arm-linux-gnueabi-gcc -pthread -DNDEBUG -g -fwrapv -O2 -Wall -Wstrict-prototyp
es -fno-strict-aliasing -D_FORTIFY_SOURCE=2 -g -fstack-protector-strong -Wformat
-Werror=format-security -fPIC -I/usr/include/python2.7 -c source/common.c -o bu
ild/temp.linux-armv7l-2.7/source/common.o
arm-linux-gnueabi-gcc -pthread -DNDEBUG -g -fwrapv -O2 -Wall -Wstrict-prototyp
es -fno-strict-aliasing -D_FORTIFY_SOURCE=2 -g -fstack-protector-strong -Wformat
-Werror=format-security -fPIC -I/usr/include/python2.7 -c source/constants.c -o
build/temp.linux-armv7l-2.7/source/constants.o
arm-linux-gnueabi-gcc -pthread -shared -Wl,-O1 -Wl,-Bsymbolic-functions -Wl,-z
,relro -fno-strict-aliasing -DNDEBUG -g -fwrapv -O2 -Wall -Wstrict-prototypes -D
_FORTIFY_SOURCE=2 -g -fstack-protector-strong -Wformat -Werror=format-security -
Wl,-z,relro -D_FORTIFY_SOURCE=2 -g -fstack-protector-strong -Wformat -Werror=for
mat-security build/temp.linux-armv7l-2.7/source/py_gpio.o build/temp.linux-armv7
l-2.7/source/c_gpio.o build/temp.linux-armv7l-2.7/source/cpuinfo.o build/temp.li
nux-armv7l-2.7/source/event_gpio.o build/temp.linux-armv7l-2.7/source/soft_pwm.o
build/temp.linux-armv7l-2.7/source/py_pwm.o build/temp.linux-armv7l-2.7/source/
common.o build/temp.linux-armv7l-2.7/source/constants.o -o build/lib.linux-armv7
l-2.7/RPi/_GPIO.so
pi@raspberrypi:~/Downloads/RPi.GPIO-0.6.3 $

```

Рисунок 3.1 – Отчет о выполнении команды

Далее необходимо запустить установку библиотеки с помощью простой команды «sudo python setup.py install» [14]. Это последняя стадия для начала работы в среде Python. После ввода на экране появится новый отчет о выполнении успешной команды.

```

File Edit Tabs Help
pi@raspberrypi:~/Downloads/RPi.GPIO-0.6.3 $ sudo python setup.py install
running install
running build
running build_py
running build_ext
running install_lib
creating /usr/local/lib/python2.7/dist-packages/RPi
copying build/lib.linux-armv7l-2.7/RPi/_GPIO.so -> /usr/local/lib/python2.7/dist
-packages/RPi
copying build/lib.linux-armv7l-2.7/RPi/__init__.py -> /usr/local/lib/python2.7/d
ist-packages/RPi
creating /usr/local/lib/python2.7/dist-packages/RPi/GPIO
copying build/lib.linux-armv7l-2.7/RPi/GPIO/__init__.py -> /usr/local/lib/python
2.7/dist-packages/RPi/GPIO
byte-compiling /usr/local/lib/python2.7/dist-packages/RPi/__init__.py to __init
_.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/RPi/GPIO/__init__.py to __
init__.pyc
running install_egg_info
Writing /usr/local/lib/python2.7/dist-packages/RPi.GPIO-0.6.3.egg-info
pi@raspberrypi:~/Downloads/RPi.GPIO-0.6.3 $

```

Рисунок 3.2 – Отчет о установке библиотеки

Библиотека установлена, можно начинать работать. Помимо этой библиотеки существует и другие способы к доступу GPIO, а именно использовать можно библиотеку WiringPi, эта библиотека написана на языке C, используется в основном в средах C, C++, Basic и на других. Она была разработана схожей с библиотекой Wiring на платформе Arduino.

Для программирования и настройки контактов общего назначения вместе с библиотекой используется утилита GPIO для командной строки. Она позволяет записывать и считывать данных с портов Raspberry Pi и управлять ими и различными процессами.

3.2 Листинг программы

Теперь на данном этапе следует приступить к составлению программы, которая будет управлять включением и выключением светодиода по необходимому алгоритму. Для примера, включить можно светодиод на пару секунд, а затем выключить.

Первое, что необходимо сделать — это задать тип нумерации выводов общего назначения. Есть два основных типа. Первый тип — это нумерация разъема, а второй тип — нумерация выводов микропроцессора. На рисунке изображен разъем GPIO.



Рисунок 3.3 – Нумерация выводов Raspberry PI и назначение

Слева на право начинается нумерация для первого типа: 1,2. За ней не подписано, но идет по порядку: 3,4 — 5,6 — и т.д. В конце разъема — 39,40.

Сверху и снизу идет нумерация основного процессора (Broadcom). На данной схеме подключение светодиода выполнена к выводу №12 согласно нумерации BCM.

В программе тип нумерации задается с помощью функции `setmode`:

```
GPIO.setmode(GPIO.BCM)
```

Теперь нужно настроить и задать режим работы вывода №12. Всего существуют два режима: вывод (OUT) и ввод (IN). Нужно управлять

светодиодом, значит необходим режим «вывод». А вот если бы по заданию было прочитать какой-то сигнал, например, с датчика, то следовало бы выбрать второй соответствующий режим «ввод» [15].

```
GPIO.setup(12, GPIO.OUT)
```

Чтобы подать на вывод №12 истину (True), использовать надо функцию output:

```
GPIO.output(12, True)
```

Помимо этих функций, еще понадобится пауза sleep. Эта функция ставит на паузу выполнение программы. В качестве аргумента указывается количество секунд паузы.

```
sleep(5)
```

Наконец, чтобы функции работы с GPIO и sleep заработали, в самом начале программы нужно добавить две библиотеки:

```
from RPi import GPIO  
from time import sleep
```

Итак, вся программа целиком выглядит на программной среде:

```
from RPi import GPIO  
from time import sleep  
GPIO.setmode(GPIO.BCM)  
GPIO.setup(12, GPIO.OUT)  
GPIO.output(12, True)  
sleep(5)  
GPIO.output(12, False)  
GPIO.cleanup()
```

В конце программы желательно рекомендуется добавить функцию cleanup, которая сбросит все настройки GPIO портов, чтобы они не мешали запуску очередной программы.

Усовершенствовать программу возможно таким образом, чтобы светодиод мигал непрерывно с частотой 1 Гц (то есть один раз в секунду). Для этого необходимо будет добавить в программу оператор цикла while [16].

```
from RPi import GPIO  
from time import sleep
```

```
GPIO.setmode(GPIO.BCM)
GPIO.setup(12, GPIO.OUT)
while True:
    GPIO.output(12, True)
    sleep(0.5)
    GPIO.output(12, False)
    sleep(0.5)
GPIO.cleanup()
```

Цикл `while` будет работать пока истинно условие, указанное в нем. Чтобы цикл работал без остановки, в качестве условия поставить можно `True`.

Составленная специальная программа имеет один неприятный момент. Если ее запустить, она войдет в бесконечный цикл и будет невозможной её остановить штатными средствами. А если её остановить нештатно, то не выполнится команда `cleanup`, что нежелательно.

Добавим в программу еще одну конструкцию, которая позволит нам в любой момент прервать работу цикла с помощью комбинации «выключить». При этом программа завершится в нормальном режиме и выполнит полноценную функцию `cleanup` [18].

```
from RPi import GPIO
from time import sleep

GPIO.setmode(GPIO.BCM)
GPIO.setup(12, GPIO.OUT)
try:
    while True:
        GPIO.output(12, True)
        sleep(0.5)
        GPIO.output(12, False)
        sleep(0.5)
except KeyboardInterrupt:
    print 'program stop'
    GPIO.cleanup()
```

3.3 Алгоритм работы программы

Основная работа подошла к концу, при сборке всех элементов следует подключить микрокомпьютер и дать соответствующее питание. Первым делом следует зайти в терминал (командная строка) и набрать команду «`ifconfig`», потом отобразится весь список адресов локальных сетей, беспроводных сетей и к тому

подобное. Подключение может быть через Ethernet и через Wi-Fi, в обоих случаях прибор будет работать. Только при вводе команды следует выбрать из списка подходящий адрес подключения, это может быть в списке ethernet или же wlan, следует выбрать подключенный способ пользователем.

```
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Jun 28 12:38:21 2018 from 192.168.43.207
pi@majordomo:~ $ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether b8:27:eb:2a:b7:ef txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1 (Local Loopback)
    RX packets 2227 bytes 340384 (332.4 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2227 bytes 340384 (332.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.43.60 netmask 255.255.255.0 broadcast 192.168.43.255
    inet6 fe80::ad3b:e44c:959c:6bdb prefixlen 64 scopeid 0x20<link>
    ether b8:27:eb:7f:e2:ba txqueuelen 1000 (Ethernet)
    RX packets 218 bytes 34576 (33.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 314 bytes 55425 (54.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

pi@majordomo:~ $
```

Рисунок 3.4 Отображение адресов

Далее, нужно включить устройство, с которого возможно будет управлять процессом, для примера можно сначала проверить работу с основного микрокомпьютера и в окне браузера вводим адрес из выбранного списка команды, в данном случае «192.168.1.2».

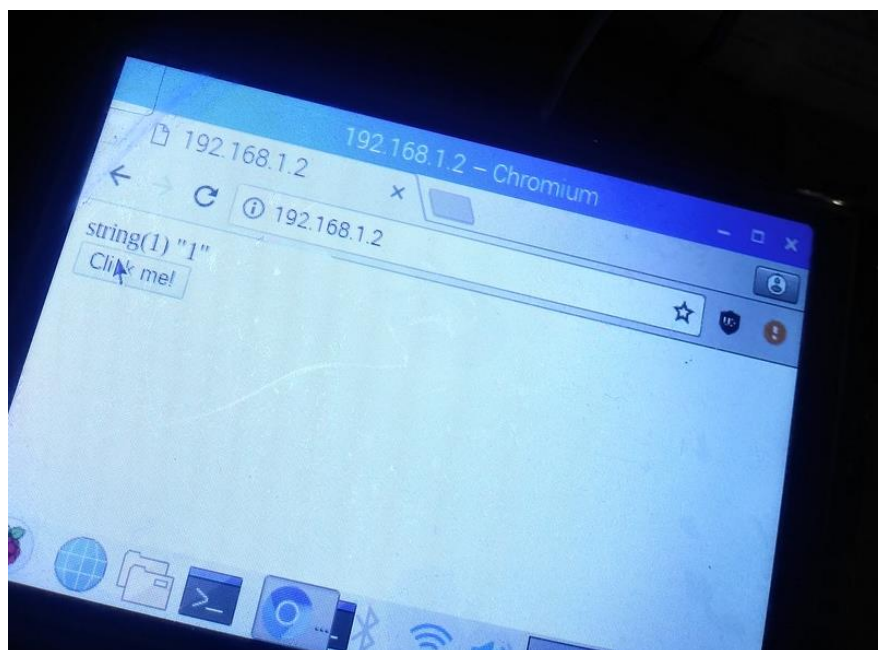


Рисунок 3.5 – Вход на сайт управления

Как видно, устройство успешно подключается к сайту, созданный с компонентами HTML, Apache, MySQL, PHP. Далее при нажатии кнопки «Click Me» можно управлять процессом непосредственно связанный с прибором, подключенный к сети. Стоит отметить, что граница подключения – безграничная, можно управлять с другой точки света при этом имея доступ к сети. При нажатии светодиод загорается.

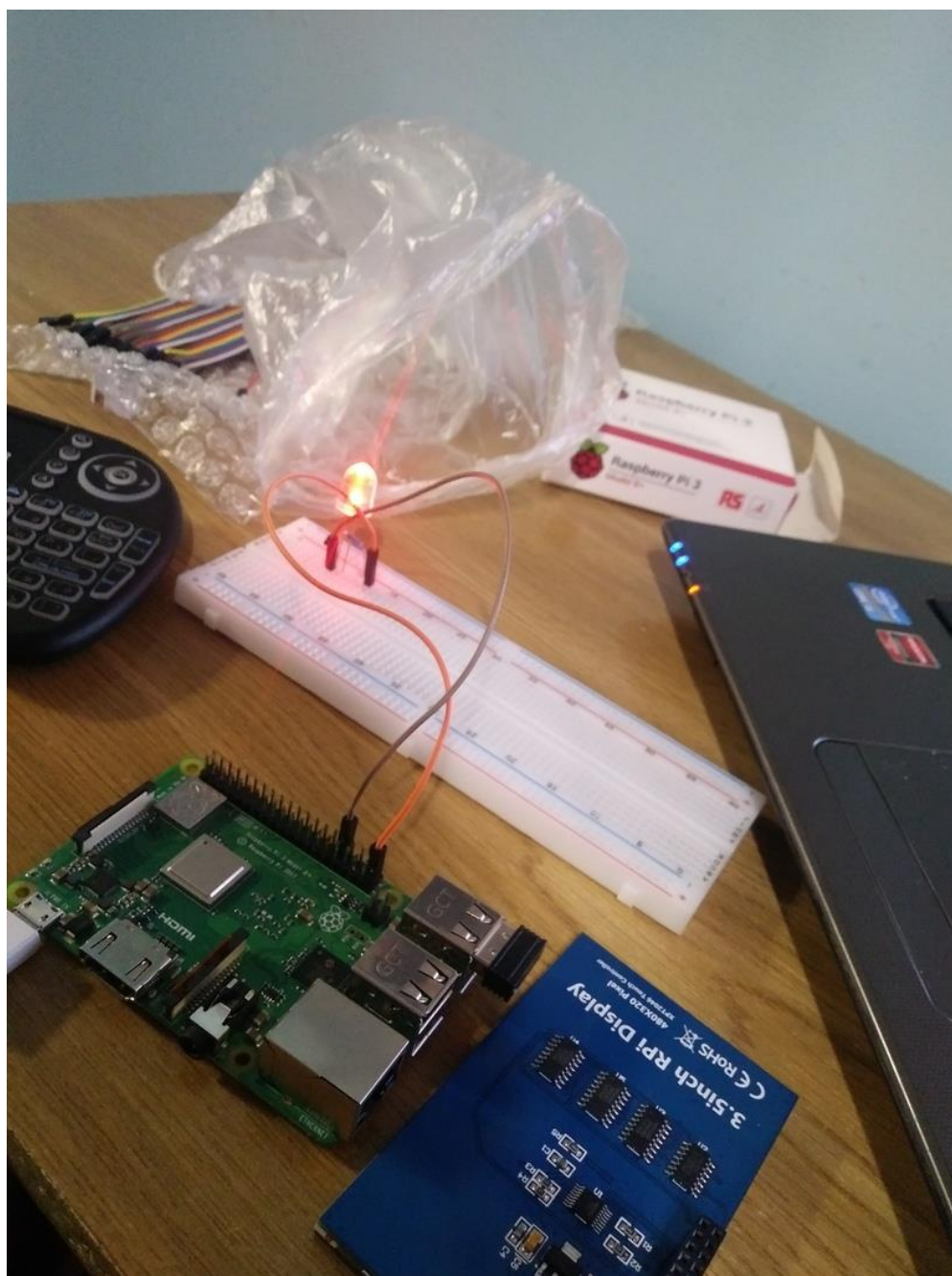


Рисунок 3.6 – Проверка диода

4 Безопасность жизнедеятельности

4.1 Анализ опасных и вредных факторов

Во время трудового дня работника в помещении рабочего стола к вредным и особо критическим и физическим факторам можно добавить: элементы, находящиеся в постоянном движении лабораторной техники (провода, подключаемые к Raspberry Pi), отсутствие чистоты или среднего загрязнения на рабочем столе, значительно завышенная температура оборудования в рабочем месте, относительно повышенная или пониженная температура воздуха в помещении, превышенный уровень шума в рабочей зоне, нестабильные уровни влажности воздуха в рабочем месте, пониженная или повышенная подвижность воздуха в месте, где идет режим работы, значительно высокое напряжение в цепи, так как замыкание, возможно, может произойти через открытое тело работника; повышенный уровень электромагнитного излучения, повышенная напряженность электромагнитного поля, недостаток или даже отсутствие естественного освещения, недостаточная или так же отсутствие освещенности в поверхности рабочей лаборатории, наличие отраженной и прямой блескости от находящихся вблизи, вдали предметов [20].

Психологический и эмоциональный стресс, перенапряжение органов слуха и зрения нужно отнести к психофизиологическим критическим и вредным производственным факторам, которые должны строго соблюдать. Следует более тщательно рассмотреть имеющуюся в рабочем месте технику и точно определить степень ее воздействия на работников.

Персональному компьютеру соответствуют такие особо серьезные и вредные производственные факторы такие как: повышенный уровень электромагнитного излучения, повышенный уровень ионизирующего излучения, пониженная и повышенная аэроионизация воздуха, отраженная и прямая блескость, лишняя работа на костно-мышечный аппарат и перегрузки мышц кистей рук, напряжение зрительного органа, умственное напряжение, эмоциональная нагрузка, однообразность труда.

При обычной работе с персональным компьютером имеют положение быть также и эргономические факторы, которые негативно влияют на здоровье работника: так как при функционировании с компьютером относится к зрительной работе, которая подвергается напряжению, то в первую очередь идет особая нагрузка на зрение. Этот критический фактор можно найти во многих документах, которые регламентируют деятельность сотрудника с ПЭВМ. Для предотвращения отрицательного влияния на зрительные средства необходимо строго соблюдать режим отдыха с зарядкой и работы.

Рекомендательно, труд лабораторного работника соответствует группе В, так как требует интерактивного рабочего процесса с большим множеством программного обеспечения для наилучшей работы. Стоит отметить, что на зрительную усталость влияет освещение рабочего стола работника. Наиболее видна важность данного условия в то же время при работе с электронной и

бумажной документацией. В соответствии СанПиН [9]: степень иллюминации рабочего участка при работе за компьютером должен иметь 300-500 лк. В случае этом, монитор и источники света должны быть в порядке таким образом, чтобы не иметь на поверхности экрана бликов.

Важную негативную значимость играет и нежелательные позы во время функционирования с компьютером. Продолжительное присутствие в бездейственной позе несет за собой к нагрузке мышц, а за ними и воспаление, а так же связок, сухожилий спины и ног, создаются болезни позвоночника и суставов (остеохондроз, тендинит и другие), а при постоянном напряжении рук и вред кисти и сухожилий (так называемый синдром лучезапястного сустава или туннельный синдром).

Сделаем анализ участка, где находятся работники:

- рабочий участок расположен на шестом этаже;
- категория расположения: лаборатория;
- габариты кабинета: $6 \times 4 \times 3$ (соответственно ширина, высота, длина);
- основами свечения являются: светильники – 3 шт., в обоих осветительных приборах по 2 люминесцентные лампы;
- в помещении находится одно окно, габариты окна: 2×1.5 , и витрина с типом декоративности размером 1×1 ;
- цвет комнаты, мебели строго должна способствовать формированию соответствующих условий для зрительного принятия;
- для возможного предохранения от повышенной ослепительности окон следует применять жалюзи;
- в служебном кабинете присутствуют 2 работника, которые работают в утреннюю смену. График работы: с 9:00 до 18:00. На картинке 4.1 расположена схема установления ламп в рабочем столе. Природное освещение обладает повышенной биологической и санитарно-эпидемиологической сыростью, потому что показывается условно подходящим для зрения человека спектральным построением и оказывает желательное воздействие на самочувствие работника психологически в целом.

Основным из особо опасных производственных факторов в рабочем участке, возможно, считать недостаток, либо вообще полное его отсутствие, природного освещения.

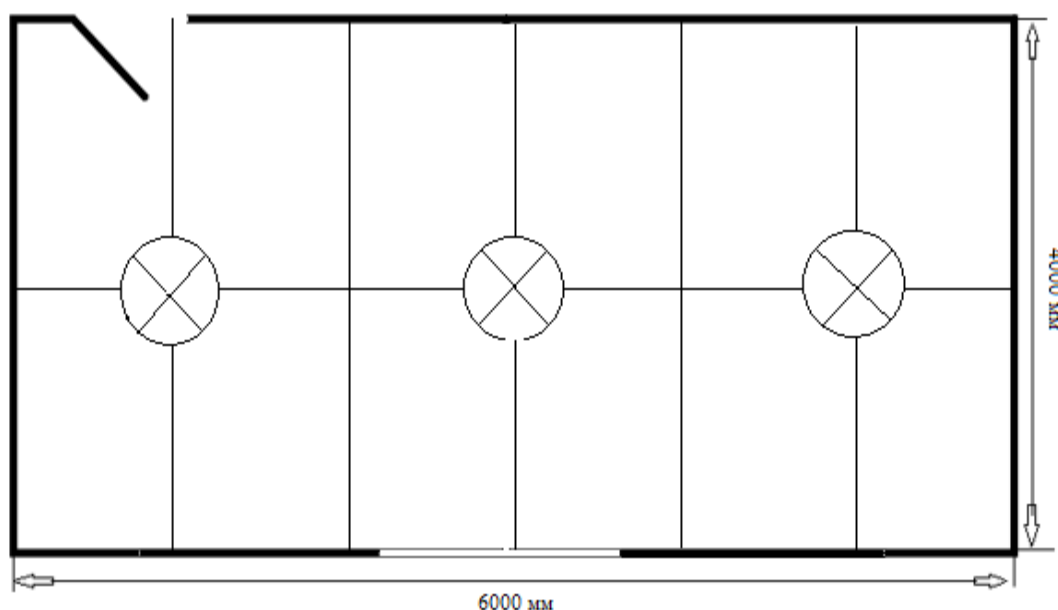


Рисунок 4.1 – Расположение ламп в участке

Общепроизводственный свет, который был безошибочно спроектирован и заодно сделан, является благоприятным условием, который помогает совершенствованию факторов зрительного функционирования, а еще полезно сопоставляет на уменьшение усталости, целебно сказывается на увеличении продуктивности работы. Несомненно, утвердительно показывается на производственную микросреду, и этим показывает желательное психологическое влияние на работников, снижая риск травматизма и повышает безопасность труда.

При реализации работ типов высокой визуальной безошибочности (самый маленький размер объекта различения 0,3–0,5 мм) формат коэффициента природного освещения строго должна быть не ниже 1,5%, а при визуальной работе средней корректности (минимальный параметр объекта различения 0,5–1,0 мм) КЕО должен быть не ниже 1,0%. Как источник искусственного освещения обычно используются люминесцентные лампы категории ЛД или ЛБ, которые попарно объединяются и становятся светильниками. Эти светильники должны находиться над функционирующими поверхностями в равномерно-прямоугольном порядке. Критерии к освещенности в участках, где расположены персональные компьютеры, следующие: при совершении визуальных работ повышенной безошибочности суммированная освещенность должна быть 300 лк, а комбинированная – 750 лк; аналогичные критерии при реализации работ средней пунктуальности – 200 и 300 лк целесообразно.

4.2 Расчетная часть

4.2.1 Расчет искусственного освещения

Экспозиция функционирующей поверхности, образующая светильниками общего освещения в общей комбинировующей системы, должна формировывать не менее 10% для комбинированного освещения при тех носителях света, которые используются для локального освещения. При этом освещенность должна быть не менее 300 лк при лампах типа люминесцентных. Как правило, норма освещенности при искусственном освещении отображены в таблице 4.1.

Таблица 4.1 – Величина освещенности при искусственном освещении

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта различения, мм	Разряд зрительной работы	Освещенность, лк
			Искусственное освещение
Средней точности	Свыше 0,5 до 1,0	IV	300

Обобщенно, синтетическое освещение располагается с помощью электрических источников света следующих типов: лампы накаливания и лампы люминесцентные.

В рабочем месте будут использованы люминесцентные лампы, потому что у таких ламп существует множество пользы:

а) если анализировать по спектральной структуре, то информации лампы сходен к природному, дневному освещению;

б) у ламп типа «люминесцентных» повышенный КПД (относительно ламп накаливания: относительно выше в 1,5-2 раза);

в) описываются увеличенной светоотдачей (в 3-4 раза больше, сравнительно с лампами накаливания);

г) достаточно долгосрочный период использования.

Ниже приведен подсчет и относится только к офису, с данными характеристиками: ширина – 4 метра, длина – 6 метров [10].

Сперва сделаем подсчет электросвещенности кабинета и совершим сравнение с приведенным величиной – это нужно для обнаружения в достаточной мере данного ненатурального освещения в комнате. Подсчитаем высоту светильника подвеса над функционирующей наружностью:

$$H=h-h_p-h_c, \quad (5.1)$$

где h_c – дистанция от светильника до заграждения, $h_c=0,05$ м;

h_p – высота функционирующей наружности над полом, $h_p=0,75$ м;

h – высота участка, $h=3$ м.

$$H = 3 - 0,75 - 0,05 = 2,2 \text{ м.}$$

Чтобы точно выяснить более выгодную дистанцию между двумя светильниками нужно применить данную формулу (5.2):

$$L = \lambda * H. \quad (5.2)$$

где λ - касательные энергетические, а так же включая светотехнические, рентабельные дальности между расположенными двумя светильниками ($\lambda = 1.2 \div 1.4$).

$$L = 1,2 * 2,2 = 2,64 \text{ м.}$$

Показатель комнаты определяется по формуле (5.3):

$$i = \frac{S}{H * (A + B)}, \quad (5.3)$$

где S – площадь комнаты, $S = 24 \text{ м}^2$;

H – определенная высота подвеса, $H = 2,2 \text{ м}$;

A – ширина комнаты, $A = 4 \text{ м}$;

B – длина комнаты, $B = 6 \text{ м}$.

Применив формулу (5.3), получим:

$$i = \frac{24}{2,2 * (4 + 6)} = 1,09.$$

Как только выявили освещения и, планировав участка, употребляется данный коэффициент, как коэффициент отсвета. От текущего коэффициента будут тесно связаны параметры освещаемых устройств, которые применяются для достижения нужного степени освещенности комнаты. Если коэффициент отсвета потолка, стен, пола высоки, то применяются светильники с наименьшей значительностью. Коэффициенты отражения наружностей определены и написаны ниже в таблице 4.2.

В данном случае коэффициенты отражения стен, пола и потолка равны:

$P_{\text{ст}} = 30\%$;

$P_{\text{пот}} = 50\%$;

$P_{\text{пол}} = 10\%$ (темная поверхность).

Приведенной ниже таблицы 4.3, анализируем коэффициент употребления, основываясь на показателе участка.

Таблица 4.2 – Коэффициенты отражения

Характеристика поверхности	Коэффициент отражения, %
Белая поверхность (белые плитка/обои, побелка)	70
Светлая поверхность (ясный потолок, ясные обои, лучезарная краска)	50
Поверхность средней яркости (светлые линолеум/паркет, несветлые обои/краска)	30
Тусклая поверхность (тусклые паркет/краска/линолеум/обои)	10

Таблица 4.3 – Величина коэффициента употребления для светильников с лампами категории люминесцентные, %

i	Р _{пот} , % 70 Р _{ст} , % 50 Р _{пол} , % 30	Р _{пот} , % 50 Р _{ст} , % 30 Р _{пол} , % 10	Р _{пот} , % 30 Р _{ст} , % 10 Р _{пол} , % 10
0,5	28	21	18
1,0	49	40	36
3,0	73	61	58
5,0	80	67	65

При требовании, что показатель участка равен 1,09, то $\eta = 40\%$;

K_3 – коэффициент запаса: $K_3=1,2$.

Чтобы определить освещенность применяем формулу (5.4):

$$E = \frac{N * \Phi_{\text{л}} * \eta * n}{S * K_3 * z}, \quad (5.4)$$

где E – необходимая освещенность горизонтальной плоскости, лк;

N – число всех светильников;

$\Phi_{\text{л}}$ – световой обилие одной лампы, лм;

η – коэффициент употребления осветительной инсталляции;

n – количество ламп в одном светильнике;

S – площадь участка, м²;

K_3 – коэффициент запаса/резерва;

z – коэффициент неровности освещения.

Вычисленные величины освещенного потока распространённых энергоисточников света описаны в таблице 4.4.

Таблица 4.4 – Вычислительные величины светового потока распространённых энергоисточников света, фл.

Типы ламп	Ф _л , лм	Типы ламп	Ф _л , лм	Типы ламп	Ф _л , лм
ЛТБ 40 – 4	2450	ЛТБ 80 – 4	4300	ДРИ 250	18700
ЛБ 40 – 4	2850	ЛБ 80 – 4	4960	ДРИ 400	32000

Буквенная номенклатура описывает спектр света, например:

- ЛБ – лампа бесцветного света;
- ЛБЦ – лампа белого света с точной цветопередачей;
- ЛД – лампа дневного света;
- ЛДЦ – лампа дневного света с точной цветопередачей;
- ЛХБ – лампа холодно – белого света;
- ЛТБ – лампа светло – белого света.

Максимальным световым трансфером и подходяще для зрения работника диапазон света описывается лампа теплого белого (ЛТБ) света.

Для основного освещения мы взяли люминесцентные лампы категории ЛТБ 80 – 4, световой поток у них, который, равен $\Phi_{л} = 4300$ лм.

В формулу (5.4) применяем все доступные величины:

$$E = \frac{3 * 4300 * 0,4 * 2}{24 * 1,2 * 1,1} = 325,75 \text{ Лк}$$

Приведенная освещенность формирует 300 Лк, а освещенность в комнате 325,75 лк. Вследствие этого, синтетического иллюминации в комнате является вполне достаточным.

4.3 Расчет уровня шума производимого персональным компьютером на работе

Повышенная степень шума является небезопасным влиянием, препятствующей производственной среде. Главными носителями шума представляют печатающие устройства, сканеры, приборы, которые кондиционируют воздух, включая вентиляторы, которые охлаждают систему ноутбуков и компьютеров.

Степень шума, создающийся от некоторых источников, функционирующих в одно и то же время, определяется с применением принципа энергетического сложения лучеиспусканий предельно отдельных источников и определяем его по формуле:

$$L_i = 10 * \lg \sum_{i=1}^{t=n} 10^{0,1} \quad , \quad (5.3)$$

где L_i – уровень звукового давления i -го источника шума;

n – является числом источников шума.

Выведенные информации подсчета сравниваются с относительно желательными величинами степени шума в рабочем участке. В некоторых случаях, когда подсчитанные результативные информации расчета являются повышенными допустимого по нормам степени шума применяются не основные меры для его понижения. К вышеуказанным мерам располагаются: использование звукоизоляции при помощи звукопоглощающих материалов, понижение шума в источнике, изменение направления основного источника шума.

Обычное рабочее место работника-оператора персонального компьютера совершенствованы таким оборудованием: жесткий диск, который находится в системном блоке, кулеры, которые выполняют работу охлаждения персонального компьютера, экран/монитор, клавиатура, сканер и микрокомпьютер. Необходимо сделать обзор каждому оборудованию.

Жесткий диск. В системном блоке применяются в основном один жесткий диск. В наше время дисковый накопитель в режиме ожидания имеет степень шума в в диапазоне 25-26 дБ. А в режиме работы поиска существующих информации шум повышается на 4-6 дБ.

Кулеры. В достаточном мере охлаждения системы, на корпусе установленного в системном блоке - имеет 3 охлаждающих кулера с уровнем шума в диапазоне 26 дБ. При максимальном возрастании продуктивности системы шум повышается на 1-2 дБ.

Монитор. На рабочем месте, где работает оператор, употребляется один ЖК монитор. Он имеет степени шума в точности 17 дБ. Если поменять изменения в работе системы, то производимый им шум не меняется.

Клавиатура. При использовании кнопки для набора текста предполагаемый уровень шума, который создает клавиатура, равен в районе 10 дБ.

Печатные устройства. Сканер и принтер при использовании в работе создают уровень шума в диапазоне 45 дБ. Степени звукового давления и характеристика каждого носителя шума показаны в таблице 5.5.

Таблица 5.5 - Представление характеристик носителей шума

Источник шума	Количество, штук	Уровень шума, дБ
Жесткий диск	1	25-26
Кулер	3	26
Монитор	1	17
Клавиатура	1	10
Принтер	1	45
Сканер	1	45

$$L_{\Sigma}=10 \cdot \lg(10^{2,6}+3 \cdot 10^{2,6}+10^{1,7}+10^1+10^{4,5}+10^{4,5})=48.122 \text{ дБ.}$$

В соответствии ГОСТ 12.1.003-76, уровень шума на рабочем месте операторов и инженеров-программистов не должен относительно увеличиваться в районе 50дБА, а в аудиториях переработки информации на расчётных машинах - 65дБА. По вычисленным расчётам, приблизительный уровень шума, создаваемый устройством в кабинете не превосходит значения допустимого. События по уменьшению уровня шума не нужны.

Вывод. В текущем дипломном проекте был проведен полный анализ опасных, критических и вредных факторов, влияющих негативно на сотрудника во время проведения работ на рабочем месте. Были приняты меры и рассмотрены уменьшения риска получения травм, несущих за собой постепенно накапливающиеся недомогания со здоровьем. При работе в лаборатории нужно желательно рассмотреть верное расположение оборудования, устройств и правильно подобранной мебели. Необходимо более детально рассмотреть технику и анализировать уровень ее воздействия на работников. Нужно проводить полный инструктаж с работниками, чтобы обезопасить их от опасностей и ждущих рисков на рабочем месте. В свою очередь, сотрудники просто обязаны следовать рекомендациям, которые позволяют полностью нейтрализовать и исключить или значительно уменьшить негативное воздействие указанных условий для работы.

5 Технико – экономическое обоснование

Оценка результативности претворения данного проекта в жизнь реализована на сравнении двух вариантов, критерием которого является анализ и выявление максимального результата от созданных работ при вложении минимума приведенных затрат «З_i». Используя данный вариант методики возможен вариант, в котором приведенные затраты имеют минимальное значение. Расчет выполняется по формуле:

$$З_i = T_z + E_n K_{pri} \rightarrow \text{минимум} \quad (5.1)$$

где T_{zi} - текущие затраты по производству конкретных видов продукции в расчете на 1 год по i-му варианту, тенге;

E_n - норма прибыли на капитал - норматив эффективности капитальных вложений, установленный инвестором, (20%);

K_{pri} - капитальные вложения по i-му варианту инвестиционного проекта, тенге.

5.1 Расчет капиталовложения для оборудования

Для осуществления проекта основными рассчитываются капиталовложения [5]. Они включают в себя нужные и необходимые вложения для приобретения оборудования и на создание программного обеспечения (ПО) продукта:

$$K_{пр} = K_{об} + K_{пп}, \quad (5.2)$$

где $K_{пр}$ – капиталовложения необходимые для осуществления проекта;

$K_{об}$ – капиталовложения необходимые для приобретения оборудования;

$K_{пп}$ – капиталовложения на создание программного обеспечения продукта.

Капиталовложения для покупки оборудования следует рассчитывать по формуле:

$$K_{об} = C_{об} + Z_{тр} + Z_m, \quad (5.3)$$

где $K_{об}$ – капиталовложения на оборудование;

$C_{об}$ – стоимость оборудования;

$Z_{тр}$ – затраты на транспортировку;

Z_m – затраты на монтаж.

Расходы на транспортировку ориентировочно составляют 5 – 10% от общей стоимости самого оборудования и рассчитываются по формуле:

$$Z_{тр} = C_{об} * 0.05. \quad (5.4)$$

Затраты на проведение монтажных работ необходимо брать из интервала от 5 до 7% от стоимости оборудования, рассчитывается по формуле:

$$Z_m = C_{об} * 0.05. \quad (5.5)$$

5.1.1 Расчет капиталовложений на оборудование для первого варианта реализации проекта

Все затраты на прибор, необходимого для создания проекта по начальному варианту составляют 35 080 тенге. Прайс-лист требуемого оборудования и их цены показаны в таблице 5.1

Таблица 5.1 – Стоимость оборудования для начального варианта реализации проекта

Наименование	Количество штук	Цена за единицу (тг.)	Итого (тг.)
Raspberry PI 3 b+ (микрокомпьютер)	1	30000	30000
Монтажная плата	1	500	500
Соединительные провода (комплект)	1	500	500
Резистор	1	30	30
Светодиод	1	50	50
Источник питания 12 В - 1 А	1	4000	4000
Итого:			35 080

При использовании формул (5.4) и (5.5) затраты на транспортировку и монтаж для первого варианта составляют:

$$Z_{\text{тр}} = 35080 * 0.05 = 1754 \text{ тенге,}$$

$$Z_{\text{м}} = 35080 * 0.04 = 1403 \text{ тенге.}$$

Исходя из этого, для начального варианта реализации проекта капиталовложения на оборудование по формуле (5.5) составляют:

$$K_{\text{об1}} = 35080 + 1754 + 1403 = 38\,237 \text{ тенге.}$$

5.1.2 Расчет капиталовложений на оборудование для второго варианта

При втором варианте в создании проекта входит общая стоимость оборудования, нужная сумма для реализации проекта составила 46 080 тенге. Список со стоимостью необходимого оборудования представлен в таблице 5.2.

При расчете по формулам (5.4) и (5.5) затраты на транспортировку и монтаж для второго варианта составили:

$$З_{тр} = 46080 * 0.05 = 2304 \text{ тенге,}$$

$$З_{м} = 46080 * 0.04 = 1843 \text{ тенге.}$$

Применение формулы (5.5), помогло нам рассчитать общие капиталовложения для второго варианта реализации проекта:

$$K_{об2} = 46080 + 2304 + 1843 = 50227 \text{ тенге.}$$

Таблица 5.2 – Стоимость оборудования для второго варианта реализации проекта.

Наименование	Количество штук	Цена за единицу (тг.)	Итого (тг.)
Raspberry Pi 3 b+ (микрокомпьютер)	1	30000	30000
Дисплей 3.5 LCD	1	7000	7000
Клавиатура Mini Keyboard	1	4500	4500
Соединительные провода (комплект)	1	500	500
Резистор	1	30	30
Источник питания 12 В - 1 А	1	4000	4000
Светодиод	1	50	50
Итого:			46080

5.2 Расчет капиталовложений на создание программного обеспечения

Расчет капиталовложений на создание программного обеспечения будет единым для обоих вариантов реализации проекта, так как мы решили, что используем микрокомпьютер Raspberry Pi в двух случаях.

Ниже приводится формула для расчет капиталовложения необходимого на разработку программного продукта $K_{пп}$:

$$K_{пп} = З_{фот} + O_{сн} + M + P_c + P_{ам} + P_{э} + P_{нак}, \quad (5.6)$$

где $З_{фот}$ – фонд оплаты труда, предназначенный для разработчиков ПП, тенге;

$O_{сн}$ – отчисления по социальному налогу, тенге;

M – затраты на материалы, тенге;

P_c – расходы на специальные программные средства, необходимые для разработки проектного решения, тенге;

$P_{ам}$ – расходы на амортизацию техники, тенге;

$P_{эл}$ – расходы на электроэнергию, тенге;

$P_{\text{нак}}$ – накладные расходы, тенге.

Фонда оплаты труда $Z_{\text{фот}}$ определяется по формуле:

$$Z_{\text{фот}} = Z_{\text{осн}} + Z_{\text{доп}}, \quad (5.7)$$

где $Z_{\text{осн}}$ – основная заработная плата, тенге;

$Z_{\text{доп.}}$ – дополнительная заработная плата, тенге.

Основной оклад разработчиков ПО рассчитывается по формуле:

$$Z_{\text{осн}} = t_{\text{тр}} * Z_{\text{дн}}, \quad (5.8)$$

где $t_{\text{тр}}$ – трудоемкость разработки программного продукта (чел/дни);

$Z_{\text{дн}}$ – дневная заработная плата разработчика (тенге).

Трудоемкость рассчитывается путем сложения затрат труда по отдельным этапам разработки ($t_{\text{тр}}$):

$$t_{\text{тр}} = t_{\text{оп}} + t_{\text{алг}} + t_{\text{бл}} + t_{\text{пр}} + t_{\text{отл}} + t_{\text{док}}, \quad (5.9)$$

где $t_{\text{оп}}$ - предварительная подготовка описания задачи, чел/час;

$t_{\text{алг}}$ - разработка алгоритма решения задачи, чел/час;

$t_{\text{бл}}$ - составление блок-схемы алгоритма, чел/час;

$t_{\text{пр}}$ - программирование, чел/час;

$t_{\text{отл}}$ - отладка программы на ЭВМ, чел/час;

$t_{\text{док}}$ - подготовка документации, чел/час.

Трудоемкость, при её расчете, требует использование ключевого показателя, определяемый как примерное или условное количество команд операторов вовлеченных в разработке ПО, обозначается символом «Q» и находится по формуле:

$$Q = q * c, \quad (5.10)$$

где Q – условное число операторов;

q – предполагаемое число операторов зависит от типа, значения данного коэффициента приведены в таблице 3;

c – это коэффициент, учитывающий сложность программы и ее новизну (таблица 5.3).

ПП по степени новизны делятся на группы (4):

1) разработка принципиально новых задач (группа А);

2) разработка оригинальных программ (группа Б);

3) разработка программ с использованием типовых решений (группа В);

4) разовая типовая задача (группа Г).

Таблица 5.3 – Предполагаемое число операторов «q»

Тип задачи	Пределы изменений коэффициента
Задачи учета	от 1400 до 1500
Задачи оперативного управления	от 1500 до 1700
Задачи планирования	от 3000 до 3500
Многовариантные задачи	от 4500 до 5000
Комплексные задачи	от 5000 до 5500

Таблица 5.4 – Коэффициент сложности и новизны «с».

Язык программирования	Группа сложности	Степень новизны			
		А	Б	В	Г
Высокого уровня	1	1,38	1,26	1,15	0,69
	2	1,30	1,19	1,08	0,65
	3	1,20	1,10	1,00	0,60
Низкого уровня	1	1,58	1,45	1,32	0,79
	2	1,49	1,37	1,24	0,74
	3	1,38	1,26	1,15	0,69

Рассматриваемое в данном проекте программное обеспечение написано на языке высокого уровня при использовании типовых решений и является в степени «А» новизны [6].

Формула (5.10) для расчета условного количества команд операторов:

$$Q = 1600 * 1,20 = 1920 \text{ (команд)}.$$

Следующим шагом будет расчет времени, необходимого на создание ПП на каждом этапе.

Время, требуемое на предварительную подготовку разъяснения задачи $t_{оп}$ (фактическое время) в среднем от 3-х до 5-ти дней по 6 часов:

$$t_{оп} = 30 \text{ чел / час}.$$

Время, которое тратится на разработку алгоритма решения задачи $t_{алг}$ рассчитываем по формуле:

$$t_{\text{алг}} = Q / (50 k), \quad (5.11)$$

где k – коэффициент, который характеризует квалификацию программиста по опыту его работы находится в таблице 5.5.

Таблица 5.5 – Коэффициент характеризующий квалификацию программиста

Опыт работы	Коэффициент квалификации
До двух лет	0,8
2-3 года	1
3-5 лет	1,1 – 1,2
5-7 лет	1,3 – 1,4
более 7 лет	1,5 – 1,6

Необходимое для разработки алгоритма время рассчитывается по формуле (5.11):

$$t_{\text{алг}} = 1920 / (50 * 1.1) = 34 \text{ чел/час.}$$

Время затрачиваемое для создания блок схемы $t_{\text{бл}}$ определяется так же как $t_{\text{алг}}$ соответствующей формуле (5.11):

$$T_{\text{бл}} = 1920 / (50 * 1.1) = 34 \text{ чел/час.}$$

Время затрачиваемое собственно на написание программы на языке высокого уровня $t_{\text{пр}}$:

$$t_{\text{пр}} = Q * 1,5 / (50 * k). \quad (5.12)$$

Исходя из вышеуказанных данных время, использованное на написание программы по формуле (5.12) составит:

$$t_{\text{пр}} = 1920 * 1,5 / (50 * 1.1) = 52 \text{ чел/час.}$$

Требуемое для отладки и тестирования программы время $t_{\text{отл}}$:

$$t_{\text{отл}} = Q * 4,2 / 50 * k. \quad (5.13)$$

Полученный результат в следствие произведения расчета по формуле (5.13):

$$t_{\text{отл}} = 1920 * 4,2 / 50 * 1.1 = 146 \text{ чел/час.}$$

При расчете времени на подготовку документов $t_{\text{док}}$, данные берутся по факту и примерно составляют от 3-х до 5-ти рабочих дней по 8 часов:

$$t_{\text{док}} = 25 \text{ чел / час.}$$

Формуле (5.9) нам показала, что трудоемкость разработки программного продукта будет следующей:

$$t_{\text{тр}} = 30 + 34 + 34 + 52 + 146 + 25 = 321 \text{ чел/час или 41 чел/дней.}$$

В соответствии с месячным окладом и количеством рабочих дней рассчитывается дневная оплата за работы (в среднем 22 рабочих дня). В таблице 5.6 показана информация по окладу специалистов вовлеченных в разработку ПП.

Таблица 5.6 – Оклады специалистов

Специалист – Исполнитель	Количество, человек	Заработная плата в месяц, (тенге)
Программист	1	190 000
Итого		190 000

Соответственно, дневная оплата работы программиста составит:

$$З_{\text{дн}} = 190000/22=8636 \text{ тенге.}$$

Основная зарплата по формуле 5.8 составит:

$$З_{\text{осн}} = 41 * 8636 = 354076 \text{ тенге.}$$

Дополнительная зарплата составит 10 % от основной и будет рассчитываться по формуле:

$$З_{\text{доп}} = З_{\text{осн}} * 0.1. \quad (5.14)$$

И составляет:

$$З_{\text{доп}} = 354076 * 0.1 = 35407 \text{ тенге.}$$

Таким образом, фонд оплаты труда составит:

$$З_{\text{фот}} = 354076 + 35407 = 389483 \text{ тенге.}$$

Согласно ст. 358 п. 1 НК РК социальный налог составляет 9.5% от дохода работника, и рассчитывается по формуле:

$$О_{\text{сн}} = (З_{\text{фот}} - З_{\text{по}}) * 9.5\%. \quad (5.15)$$

где $З_{\text{по}}$ - пенсионные отчисления, которые не облагаются социальным налогом и доля от фонда оплаты труда составляет 10%:

$$З_{по} = З_{фот} * 0.1. \quad (5.16)$$

Пенсионные отчисления рассчитаны по формуле (5.16) равны:

$$З_{по} = 389483 * 0.1\% = 38948 \text{ тенге.}$$

Социальный налог основанный на вышеперечисленных расчетах и подсчитанный по формуле (5.15) равен:

$$О_{сн} = (354076 - 35407) * 9.5\% = 30273 \text{ тенге.}$$

Расходы на амортизацию включают себя непосредственно отчисления на амортизацию от стоимости техники, которая применяется при создании ПП, для чего используется формула:

$$P_{ам} = \frac{C_{обор} * H_a * N}{100 * 12 * t}, \quad (5.17)$$

где H_a – норма амортизации, составляющая (25%);

$C_{обор}$ – первоначальная стоимость техники, тенге;

N – время использования персонального техники, (41 дней);

t – количество рабочих дней в месяце, дни.

Расходы амортизационные согласно формуле (5.18) составляют:

$$P_{ам} = \frac{46080 * 25 * 41}{100 * 12 * 22} = 1789 \text{ тенге.}$$

Затраты на электроэнергию вычисляются по формуле:

$$P_э = M * k_з * T * C_{кВт-ч}, \quad (5.18)$$

где M – мощность ЭВМ, кВт;

$k_з$ – коэффициент загрузки (0.8);

$C_{кВт-ч}$ – стоимость 1 кВт·ч электроэнергии, тенге/ кВт·ч;

T – время работы, час (328 ч., так как пользование техникой 41 дней, учитывая коэффициент загрузки).

Данные о затратах на электроэнергию были определены формуле (5.19) и указаны в таблице 5.7:

$$P_{эл} = 0.12 * 0.8 * 321 * 16.65 = 513 \text{ тенге.}$$

Таблица 5.7 - Затраты на электроэнергию

Наименование оборудования	Паспортная мощность, кВт	Коэффициент загрузки	Время работы оборудования для разработки ПП, час	Цена эл/э, тенге/кВт-час;	Сумма, тенге
Микрокомпьютер	0.12	0.8	321	16.65	513
ИТОГО затраты на электроэнергию					513

Накладные расходы, ($P_{\text{нак}}$) и составляют от 40 до 60% от основной заработной платы [2]:

$$P_{\text{нак}} = Z_{\text{осн}} * N_{\text{рн}}/100\%, \quad (5.20)$$

где $P_{\text{нак}}$ – накладные расходы на ПП (тенге);

$N_{\text{рн}}$ – норматив накладных расходов -20%.

$$P_{\text{нак}} = 354076 * 0.2 = 70815 \text{ тенге.}$$

В итоге, капитальные вложения на разработку программного продукта составили:

$$K_{\text{пп}} = 354076 + 35407 + 513 + 70815 = 460811 \text{ тенге.}$$

Сводные результаты расчета затрат на разработку ПП представлены в таблице 5.8.

Таблица 5.8 – Затраты на разработку программного продукта

Статьи затрат	Сумма, тенге
Фонд оплаты труда	354076
Социальный налог	30273
Амортизация	1789
Электричество	513
Накладные расходы	70815
Итого:	460811

В заключении, общая сумма капиталовложений требуемая для проекта рассчитывается согласно формуле (5.2) и составляет:

Для первого варианта:

$$K_{\text{пр1}} = 38237 + 460811 = 499048 \text{ тенге.}$$

Для второго варианта:

$$K_{\text{пр2}} = 50227 + 460811 = 511038 \text{ тенге.}$$

5.3 Расчет эксплуатационных затрат

Рассматриваемый проект, в процессе эксплуатации предполагает затраты на его содержание и обслуживание, данная статья расходов называется «эксплуатационные затраты» и рассчитывается по формуле [7]:

$$\Phi_{\text{оп}} + C_{\text{нал}} + M + \mathcal{E} + A_{\text{м}} + H_{\text{р}}, \quad (5.21)$$

где

$\Phi_{\text{оп}}$ – фонд оплаты труда персонала;

$C_{\text{нал}}$ – социальный налог;

M - материальные затраты в том числе затраты на запасные части (0.5% от капиталовложений);

\mathcal{E} – затраты на электроэнергию;

$A_{\text{м}}$ – амортизационные отчисления;

$H_{\text{р}}$ – накладные расходы (30% от заработной платы).

Персонал в данном проекте не требуется, поэтому эксплуатационные затраты рассчитываются укрупнено, и формула (5.21) преобразовывается в следующий вид:

$$T_3 = M + \mathcal{E} + A_{\text{м}}. \quad (5.21)$$

В расходы по электроэнергии (\mathcal{E}) включаются затраты на оборудование ($P_{\text{эл.обор.}}$) и дополнительные расходы ($P_{\text{эл.доп.}}$):

$$\mathcal{E} = P_{\text{эл.обор}} + P_{\text{эл.доп.}} \quad (5.22)$$

$$P_{\text{эл.обор}} = N * t_{\text{раб}} * C_{\text{кВт-ч}}, \quad (5.23)$$

где N – мощность потребления, кВт (0.12 кВт);

$t_{\text{раб}}$ - время работы, ч/год (2112);

$C_{\text{кВт-ч}}$ - стоимость 1 кВт·ч электроэнергии, тенге/ кВт·ч (16.65).

Дополнительные затраты на электроэнергию рассчитываются в соответствии в укрупненным показателем, который составляет 5% от расходов на оборудование:

$$P_{\text{эл.доп}} = P_{\text{эл.обор}} * 0.05. \quad (5.24)$$

Расходы на электроэнергию по формуле (5.24) составляют:

для первого варианта:

$$P_{\text{эл.обор1}} = 0.12 * 2112 * 16.65 = 4219,77 \text{ тенге.}$$

Дополнительные затраты на электроэнергию для первого варианта:

$$P_{\text{эл.доп1}} = 4219,77 * 0.05 = 210 \text{ тенге.}$$

Соответственно, расходы на электроэнергию составляют:

для первого варианта:

$$\Xi_1 = 4219,77 + 210 = 4430,75 \text{ тенге.}$$

Амортизационные отчисления рассчитываются по формуле:

$$A_{\text{м}} = \frac{H_A * K_{\text{нр}}}{100}, \quad (5.25)$$

где $A_{\text{м}}$ – затраты на амортизацию, тенге;

K – капитальные вложения, тенге;

H_A – норма амортизации (25%).

Затраты на амортизацию для первого варианта составляют:

$$A_{\text{м1}} = \frac{0.25 * 499048}{100} = 1097 \text{ тенге.}$$

А для второго варианта амортизация равняется:

$$A_{\text{м2}} = \frac{0.25 * 511038}{100} = 1277 \text{ тенге.}$$

Эксплуатационные расходы по всем необходимым значениям и расчету по формуле (5.22) для первого варианта составили:

$$T_{31} = 4430,75 + 1097 = 5527,75 \text{ тенге.}$$

Итоги расчетов эксплуатационных затрат показаны в таблице 5.9.

Таблица 5.9 – Сводная таблица расходов в процессе эксплуатации

Наименование	Обозначение	Вариант №1
Расходы на электр.зн. (тг.)	Э	4430,75
Амортизационные отч. (тг.)	A _м	1097
Итого:(эксплуат. расходы)	Э _{зат}	5527,75

5.4 Оценка экономической эффективности реализации проекта

По формуле (5.1) приведенные затраты для первого варианта составят:

$$З_{i1} = 5527,75 + 0.2 * 38237 = 13175 \text{ тенге.}$$

Приведенные затраты первого варианта меньше приведенных затрат второго варианта, т.е. $З_{пр1} < З_{пр2}$ данный проект для реализации годен [8].

Вывод: Наша стартап-команда провела анализы исследования, в ходе которой был произведен основной расчет сравнительной экономической эффективности по созданию системы управления объектами основанных на современных технологиях, по варианту получения максимального результата от проекта при минимуме приведенных затрат. Была рассчитана себестоимость проекта. Для аналогового варианта создания при использовании более дорогостоящих компонентов также были вычислены приведенные затраты. Они и доказали, что можно будет целесообразнее и эффективнее внедрение бюджетного варианта, как более экономически эффективное финансовое решение.

Заключение

Во время выполнения дипломной работы закрепились знания по программированию на языке Python, изучены основы современной электроники, развились навыки автоматизации системы. Данная работа несёт перспективный характер, помогая разобраться в системе управления удаленным доступом и создавая всё новейшее в разработке данных процессов.

Во время выполнения работы были использованы актуальные платформы Raspberry Pi и знания кодирования, программирования и добавления элементов на актуальный прибор.

Проект считается актуальным на данный момент и можно прикрепить на различные принципиальные приборы, которые работают на основе специалиста, если при грамотном создании алгоритма и скриптов, то можно добиться автоматического выполнения задач, которые были заложены в эти алгоритмы и упрощает одну работу на предстоящей предприятии. Проекты ограничиваются лишь воображением, это наглядный пример управления интернетом. Имея под рукой карманный микрокомпьютер и имея доступ к сети, можно выполнять различные функции для создания различных систем и управляя с любой точки мира.

Дальнейшие развития обновляются со временем, на данный проект можно добавить доступ контроля, который позволит определенным лицам пользоваться прибором. Также можно установить кондиционер для охлаждения видеокарты, так как прибор может автономно пользоваться весь день и охлаждение будет поддерживать стабильность работы.

Список литературы

- 1 Стандарт организации, учебно-методические и учебные работы, СТ НАО 56023-1910-04-2014, Алматы.
- 2 Светодиоды и подключения LP01 / LP02. Техническое описание.
- 3 Жаров С.А. Основы сетевой безопасности: Криптографические алгоритмы и протоколы. – ВРс.: Винтерфэл, 2012.
- 4 Проекты на Raspberry Pi для злых гениев / McGraw-Hill Education TAB. 2017.
- 5 З.Д. Еркешева. Методические указания к выполнению экономической части дипломных работ для студентов специальности 5В071600 – Приборостроение. – Алматы: АУЭС, 2017 – 29 с.
- 6 З.Д. Еркешева. Методические указания к выполнению экономической части дипломных работ для студентов специальности 35В060200 – Информатика. – Алматы: АУЭС, 2016 – 32 с.
- 7 Ребрин Ю.И. Основы экономики и управления производством: конспект лекций. – Изд-во ТРТУ, 2009. – 145с.
- 8 Липсиц И.В. Экономика: Учебник для вузов. – Омега-Л, 2006. – 656 с.
- 9 Приходько Н.Г., Жандаулетова Ф.Р., Амренова А.Ж. Основы пожарной безопасности: Учебное пособие – А.: АУЭС. 2015. – 109с.
- 10 Приходьков К.В., Бастраков А.В., Савкин А.Н., Авдеюк О.А., Крохалев А.В. Исследование сгорания водородо-воздушных смесей // Современные проблемы науки и образования – Волгоград: 2013. – № 3.
- 11 Официальная страница Raspberry Pi – www.raspberrypi.org
- 12 E-book Raspberry Pi Computer Peter Bauer, Wildstrabe 11, A 2100, Korneuburg, Austria.
- 13 Raspberry Pi User guide, Eben Upton, Gareth Halfacree, 2012.
- 14 Raspberry Pi Cookbook: Software and Hardware Problems and Solutions, 12.2013, Simon Monk.
- 15 Python Crash Course: A Hands-On, Project-Based Introduction to Programming, 8.12.2015, Eric Matthes.
- 16 Effective Python: 59 Specific Ways to Write Better Python, 2015, Brett Slatkin.
- 17 Петин В.А. Arduino и Raspberry Pi в проектах Internet of Things, 2-е изд. СПб.: БХВ-Петербург, 2019. 432 с. Электроника . ISBN 978-5-9775-3951-7.
- 18 Python Programming with Raspberry Pi, 28.04.2017, Sai Yamanoor.
- 19 Raspberry Pi IoT Projects: Prototyping Experiments for Makers, 12.08.2016, John C. Shovic.
- 20 Make: Action: Movement, Light, and Sound with Arduino and Raspberry Pi, Simon Monk, 23.09.2016.