



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
**«Дальневосточный федеральный университет»**

---

## **ИНЖЕНЕРНАЯ ШКОЛА**

### **Кафедра электроники и средств связи**

Жибарь Марк Артурович

Разработка информационно-вычислительного комплекса с удаленным  
управлением

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**  
по образовательной программе подготовки бакалавров  
«Системы радиосвязи и радиодоступа»  
по направлению 11.03.02 – «Инфокоммуникационные  
технологии и системы связи»

г. Владивосток  
**2018**

Автор работы \_\_\_\_\_  
подпись

«\_\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

Консультант по нормоконтролю

подпись \_\_\_\_\_ (ФИО)

«\_\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

«Допустить к защите»

Руководитель ОП \_\_\_\_\_  
ученое звание

подпись \_\_\_\_\_ (ФИО)  
«\_\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

руководитель ВКР \_\_\_\_\_

должность, ученое звание

подпись \_\_\_\_\_ (ФИО)  
«\_\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

Защищена в ГЭК с оценкой \_\_\_\_\_

Секретарь ГЭК

подпись \_\_\_\_\_ (ФИО)  
«\_\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

«Допустить к защите»

Зав. кафедрой \_\_\_\_\_  
ученое звание

подпись \_\_\_\_\_ (ФИО)  
«\_\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение  
высшего образования

**«Дальневосточный федеральный университет»**

**Инженерная школа**

Кафедра электроники и средств связи

УТВЕРЖДЕНО

Руководитель ОП профессор, д.ф. - м.н.  
(ученая степень, должность)

Л.Г. Стаценко  
(подпись) (ФИО)  
«\_\_\_\_» \_\_\_\_\_ 2018 г.

Заведующий кафедрой профессор, д.ф. - м.н.  
(ученая степень, звание)

Л.Г. Стаценко  
(подпись) (ФИО)  
«\_\_\_\_» \_\_\_\_\_ 2018 г.

**З А Д А Н И Е**  
**на выпускную квалификационную работу**

Студенту (ке) Жибарь Марку Артуровичу Группа Бак5ИТС  
(Фамилия, Имя, Отчество) (номер группы)

1. Наименование темы Разработка информационно-вычислительного комплекса с удаленным управлением

2. Основания для разработки Приказ Сд-33 от 26.02.2018 г.

3. Источники разработки Документация производителей микроконтроллеров, интернет сайты, тематические журналы, статьи

4. Технические требования (параметры) Разработать ИВК с удаленным управлением для системы «Умный дом»

5. Дополнительные требования \_\_\_\_\_

6. Перечень разработанных вопросов Анализ и сравнение существующих систем «Умный дом». Выделение основных выполняемых функций. Выбор аппаратной платформы и необходимых компонентов. Разработка функциональной схемы и управляющей программы. Сравнение разработанной системы с существующими.

7. Перечень графического материала (с точным указанием обязательных плакатов) \_\_\_\_\_

Рисунок 7 - Типовая структурная схема системы «Умный дом»

Рисунок 15 - Общая структурная схема микроконтроллеров AVR

Рисунок 16 - Структурная схема интерфейса I2C

Рисунок 17 - Структурная схема интерфейса USI

Рисунок 21 - Принципиальная схема подключения NodeMCU к Arduino

Рисунок 29 - Принципиальная схема устройства автономного питания

Рисунок 52 - Схема подключения драйвера управления двигателем

Рисунок 53 - Схема реализации функции контроля протечек

Приложение А - Структурная схема системы «Умный дом» для интеграции в здание

### КАЛЕНДАРНЫЙ ГРАФИК ВЫПОЛНЕНИЯ РАБОТЫ

№ п/п	Наименование этапов дипломного проекта (работы)	Срок выполнения этапов дипломного проекта (работы)	Примечание
1.	Анализ технического задания. Поиск и исследование литературы по теме выпускной квалификационной работы	01.05.18 - 11.05.18	
2.	Составление плана работы	11.05.18 - 21.05.18	
3.	Разработка и согласование с руководителем основной части работы	21.05.18 - 04.06.18	
4.	Согласование с руководителем введения, выводов и предложений	04.06.18 - 08.06.18	
5.	Подготовка доклада и графического материала	08.06.18 - 13.06.18	

Дата выдачи задания \_\_\_\_\_

Срок представления к защите \_\_\_\_\_

Руководитель ВКР \_\_\_\_\_

(подпись)

(ФИО)

Студент \_\_\_\_\_

(подпись)

(ФИО)

## **Аннотация**

Тема ВКР: «Разработка информационно-вычислительного комплекса с удаленным управлением».

Объем – 70 страниц, 53 рисунка, 9 таблиц, 2 приложения, список литературы: 15 источников, из них 8 – интернет-источники.

При расчетах и при создании графического материала использовалось программное обеспечение «Proteus», «Adobe Photoshop», «Microsoft Visio 2013».

Структура ВКР включает: введение, выбор способа построения ИВК для системы «Умный дом», разработка ИВК для удаленного управления системой «Умный дом», а также заключение и список литературы.

Во введении представлены цели выполнения ВКР и постановка задач.

В разделе «Актуальность» приводится обзор рынка существующих систем и определение стоимости.

В разделе «Выбор способа построения ИВК для системы «Умный дом» приводится исследование концепции и структуры информационно-вычислительных комплексов в применении к системе «Умный дом» и выделение основных функциональных особенностей системы

В разделе «Разработка ИВК для удаленного управления системой «Умный дом» приводится выбор платформы, обеспечение удаленного управления с помощью Arduino, выбор компонентов устройства автономного питания, выбор измерительных датчиков и устройств управления для интеграции в систему, расчет итоговой стоимости системы и сравнение с существующими системами на рынке.

В заключении приводятся обоснование выбранного метода разработки системы и результаты проделанной работы.

**Ключевые слова:** Информационно-вычислительные комплексы, система «Умный дом», микроконтроллеры, измерительные датчики, исполнительные устройства.

## **ВВЕДЕНИЕ**

Внедрение умных домов в типовом (среднестатистическом) российском домохозяйстве сталкивается с рядом существенных ограничений. С одной стороны, умные технологии сами по себе являются дорогостоящими (например, создание энергоэффективного здания обходится в среднем на 30% дороже обычного). Не в последнюю очередь высокая себестоимость комплектующих объясняется наличием импортных таможенных пошлин и отсутствием локализации в России. Данный фактор приводит к тому, что интеллектуальные решения воспринимаются населением как товары роскоши.

С другой стороны, специфическим для страны барьером развития рынка является высокая консервативность населения в плане внедрения новых технологий (что также сказывается в качестве профессиональной подготовки и образования) и его недостаточная платежеспособность.

Также среди ограничений развития умных домов рассматриваются дешевизна энергоресурсов для населения (например, низкий уровень цен на электроэнергию сдерживает потенциал экономии на квартплате за счет энергоэффективных решений), отсутствие достаточного количества квалифицированных кадров для массовой установки и настройки умных систем (применяющееся в настоящее время оборудование является достаточно сложным в настройке), низкое качество отдельных партий импортного оборудования для умных домов. Кроме того, управляющие компании не всегда обладают ресурсами для эксплуатации системы умных домов.

Одним из ключевых глобальных барьеров развития умных домов является отсутствие единых стандартов к производимому оборудованию. Рынок решений все еще находится на стадии формирования, а потому протоколы связи между устройствами разных производителей значительно отличаются. Незрелость технологий, функциональная несовместимость и ограниченный доступ

устройств к сети являются следствием вертикальной некооперированности основных игроков рынка.

Необходимость формирования единой технологической платформы для преодоления несовместимости систем уже была признана ключевыми участниками рынка.

Значительным затруднением при развитии умных домов эксперты признают маркетинговые ограничения. Ключевое из них связано с недостаточной информированностью потребителей о ценности систем умного дома, недооценкой проблем, которые они могут решить благодаря ним. Более того, системы умного дома все еще являются недостаточно узнаваемыми среднестатистическим потребителем. По причине указанного маркетингового ограничения в настоящее время большинство компаний сделало ставку на «комплектное» и компактное решение по доступной цене вместо отдельных многофункциональных элементов систем.

Дополнительным маркетинговым ограничением можно считать повышенную длительность жизненного цикла умных устройств. Часть из них инсталлируется в стены и иные предметы домашней обстановки, вследствие чего их замена по причине устаревания сопряжена с определенными финансовыми затратами и неудобствами.

По поводу распространённости, согласно исследованиям [2], на сегодняшний день даже в технологически прогрессивных США умными являются не более 6% домов. Этот показатель кажется низким, но в России он не превышает 1% на конец 2017г.

Основная масса интеллектуальных зданий сосредоточена в Москве и Санкт-Петербурге. Однако рост рынка обеспечивается, во многом, за счет регионов. Пока «умными» системами пользуются, в основном, владельцы коммерческой недвижимости: по разным оценкам, от 80 до 90% российских «интеллектуальных» проектов приходится на корпоративный сектор.

Так что пока «Умных» домов в России немного, этот рынок в нашей стране будет продолжать формироваться. Пока это в основном подмосковные коттеджи или элитные квартиры в Москве и Санкт-Петербурге.

Целью данной работы является разработка недорогой, простой и универсальной системы «Умный дом».

Для достижения поставленной цели необходимо решить следующие задачи:

- выбрать оптимальную структуру системы управления умным домом;
- провести анализ и выбрать элементную базу системы;
- решить вопрос удаленного управления;
- обосновать эффективность предлагаемых решений.

# 1 АКТУАЛЬНОСТЬ СИСТЕМЫ «УМНЫЙ ДОМ» В СОВРЕМЕННЫХ РЕАЛИЯХ

## 1.1 Обзор рынка систем «Умный дом»

Если говорить о тенденциях, они в целом оптимистичны: согласно прогнозам исследовательской компании Statista, объемы рынка умного дома в России в ближайшее время будут стремительно расти. Ожидается, что доля умных домов в России в 2019 году дойдет до отметки 1,7%, а к 2022 году достигнет 8,5% [1].

Как правило, прогнозам свойственен некоторый идеализм, но уже сейчас заметно, что крупные компании щедро инвестируют в умный дом: стоит отметить 2014 год, когда Google купила стартап Nest за \$3,2 млрд – умный термостат, автоматически регулирующий температуру в квартире.

Чтобы выделить основные проблемы, необходимо проанализировать рынок существующей продукции. Для этого нужно привести обзор самых популярных проектов и рассмотреть возможности и недостатки каждой системы:

1. Samsung SmartThings (рисунок 1).



Рисунок 1 - Устройства SmartThings

Система, основанная на взаимодействии устройств через хаб - концентратор, подключающийся по Ethernet к домашней сети.

Комплект за \$89,99: соединяет совместимые смарт-устройства, в том числе источники света, динамики, замки, терmostаты, датчики и другое.

Комплект за \$169,99: более интеллектуальный маршрутизатор с функциональностью Smart Home; легко соединяет совместимые камеры, источники света, имеет голосовое управление и т.д.

Комплект за \$249,99: аналог системы за \$169,99 с большей мощностью (рассчитан на больший охват площади).

Цены на комплектующие:

- умная лампа – 3 000 руб.;
- умная розетка – 2 500 руб.;
- датчик движения – 3 100 руб.;
- камера – 6 300 руб.;
- дверной замок – 14 000 руб.;
- концентратор (хаб) – 20 000 рублей.

Недостатки системы:

- необходимость кабельного подключения главного концентратора по Ethernet;
- недостаточная универсальность, связанная с разницей протоколов между устройствами;
- возможность подключения датчиков только данной фирмы

2. Xiaomi Smart Home Kit (рисунок 2).



Рисунок 2 - Комплект системы Smart Home Kit

Система представляет собой хаб - светильник, подключающийся непосредственно в розетку питания и передающий данные по беспроводной Wi-Fi связи с выходом в сеть Интернет.

В особенности входят интеллектуальные датчики для окон и дверей, устройства дистанционного управления с помощью смартфона, возможность подключения к другим смарт-устройствам, Push-уведомления по срабатыванию некоторых датчиков.

Недостатки:

- отсутствие русской локализации интерфейса;
- требуются переходники под розетки евростандарта;
- поддержка только Apple iOS

3. «ТРОН Проект» (рисунок 3).



Рисунок 3 - Интерфейс системы «ТРОН Проект»

Беспроводное управление домом с одного или нескольких смартфонов/планшетов, контроль освещения, штор, вентиляции, безопасности, отопления, мультимедиа и других систем. 3D-модель позволяет управлять целым домом и каждой отдельной комнатой/устройством.

Недостатком системы является слишком высокая стоимость.

4. LittleBits Smart Home Kit (рисунок 4).



Рисунок 4 - Комплект модулей LittleBits Smart Home Kit



Рисунок 5 - Состав комплекта LittleBits Smart Home Kit

Система включает в себя набор деталей - датчиков и исполнительных устройств, которые могут встраиваться в любой бытовой прибор. Имеет компонент cloudBit, обеспечивающий выход в сеть Интернет по Wi-Fi и взаимодействие с сервисом IFTTT, благодаря чему обеспечивается удалённое управление устройствами, в число которых входит: автоматизация ламп, штор, жалюзи, звуков, кондиционеров, вентиляторов, фонарей и т.д. В наборе 14 модулей и MP3-плеер.

Из недостатков отмечена низкие возможности расширения устройств, возможность подключения датчиков только из набора и приличная стоимость.

##### 5. INSTEON (рисунок 6).



Рисунок 6 - Центральный контроллер системы INSTEON

Система имеет собственный центральный контроллер с возможностью тонкой настройки и использует собственный протокол связи с частотой 131,65кГц. Данная частота выбрана из расчетов предотвращения создания помех другим бытовым ВЧ устройствам. Контроллер подключается к домашнему роутеру посредством LAN-порта.

Присутствует возможность управления со смартфона, планшета и Apple Watch. Управление светом (в том числе плавное погашение), камерами Insteon Wi-Fi, контроль температуры, Push-уведомления по срабатыванию некоторых датчиков.

Недостатки:

- закрытый протокол связи, из-за чего расширение системы и совместимость с другими датчиками невозможна;
- большой расход энергии беспроводных датчиков.

К сожалению, у большинства перечисленных выше решений отсутствует возможность гибкой настройки системы датчиков, универсальность, совместимость с разными устройствами и ощутимые цены для рядового пользователя за полный комплект.

Итак, чтобы объективно оценить среднюю стоимость системы на рынке, необходимо составить список вышеперечисленных систем (таблица 1) и посчитать среднее.

Таблица 1 - Готовые системы умного дома

Название	Страна	Цена
Samsung SmartThings	Корея	\$89,99-249,99 в зависимости от мощности и комплектации
Xiaomi Smart Home Kit	Китай	\$66
«ТРОН Проект»	Россия	от \$1740 в зависимости от сложности проекта
LittleBits	США	\$249
INSTEON	США	\$79,99

Нетрудно заметить, что средняя стоимость за комплект составляет около \$400.

## **2 ВЫБОР СПОСОБА ПОСТРОЕНИЯ ИВК ДЛЯ СИСТЕМЫ УМНЫЙ ДОМ**

### **2.1 Исследование концепции и структуры информационно-вычислительных комплексов в применении к системе «Умный дом»**

«Умный дом» - аппаратно-программный комплекс автоматизации и управления различным оборудованием квартиры и дома. Его структура основана на взаимодействии измерительных преобразователей и исполнительных устройств с главным контроллером управления в жилом здании и может вполне являться измерительно-вычислительным комплексом.

Концепция «Умного дома» включает в себя следующие положения:

- интегрированная система управления зданием как обеспечение взаимодействия системы «Умный дом» со всеми возможностями встроенных инженерных систем здания для их комплексной работы;
- устранение всего обслуживающего персонала здания и передача функций контроля подсистемам интегрированной системы управления зданием;
- реализация механизма мгновенного отключения и передачи при необходимости управления человеку любой подсистемой интеллектуального здания;
- обеспечение корректной работы отдельных подсистем в случае отказа общей управляющей системы или других частей системы;
- минимизация обслуживания здания человеком и коммуникационных услуг;
- наличие в здании проложенной коммуникационной среды.

В разработке автоматизированных решений для жилых объектов, основной задачей специалистов являются создание инженерных сетей и их управления.

Инженерные коммуникации «Умного дома» должны позволять осуществлять взаимодействие с системами:

- освещения;
- вентиляции;
- охранно-пожарной сигнализации;
- климат-контроля;
- измерения состояния окружающей среды;
- бесперебойного питания;
- отопления;
- водоснабжения;
- ... и др.

Помимо выполнения основных функций, на информационную систему здания могут возлагаться функции управления информационной инфраструктурой:

- контроль доступа к информации и управление безопасностью;
- управление событиями;
- управление рабочей нагрузкой сети;
- контроль производительности вычислительных и сетевых систем;
- управление Web-серверами;
- управление транспортом данных между устройствами;
- автоматизированное управление хранением данных;
- управление проблемами;
- управление сетью.

Последняя функция, управление сетью, включает в себя автоматическое распознавание объектов управления и топологии сети, повышение уровня контроля сетевого оборудования, наличие средств разработки для управления нестандартным сетевым оборудованием, интеграция с уже существующими системами управления сетью в здании и наличие средств ведения политики управления сетью.

Интеллектуальный дом предоставляет возможность создания в рамках интегрированной системы управления подсистемы удаленного управления, или управления на расстоянии. Эта подсистема позволяет получать информацию о событиях и состоянии некоторых параметров и удаленно посыпать команды управления всей системе или ее отдельным компонентам.

Подключение интеллектуального дома к сети позволит осуществлять мониторинг систем жизнеобеспечения, энергосбережения ресурсов, безопасности и т. д. Решение любых вопросов, связанных с упомянутыми системами, а также доставка различных услуг по сети станет гораздо проще и эффективнее. Обмен информацией как внутри дома, так и вне его происходит с помощью стандартных цифровых протоколов связи. Все компоненты системы интегрированы в единое коммуникационное пространство (рисунок 7).

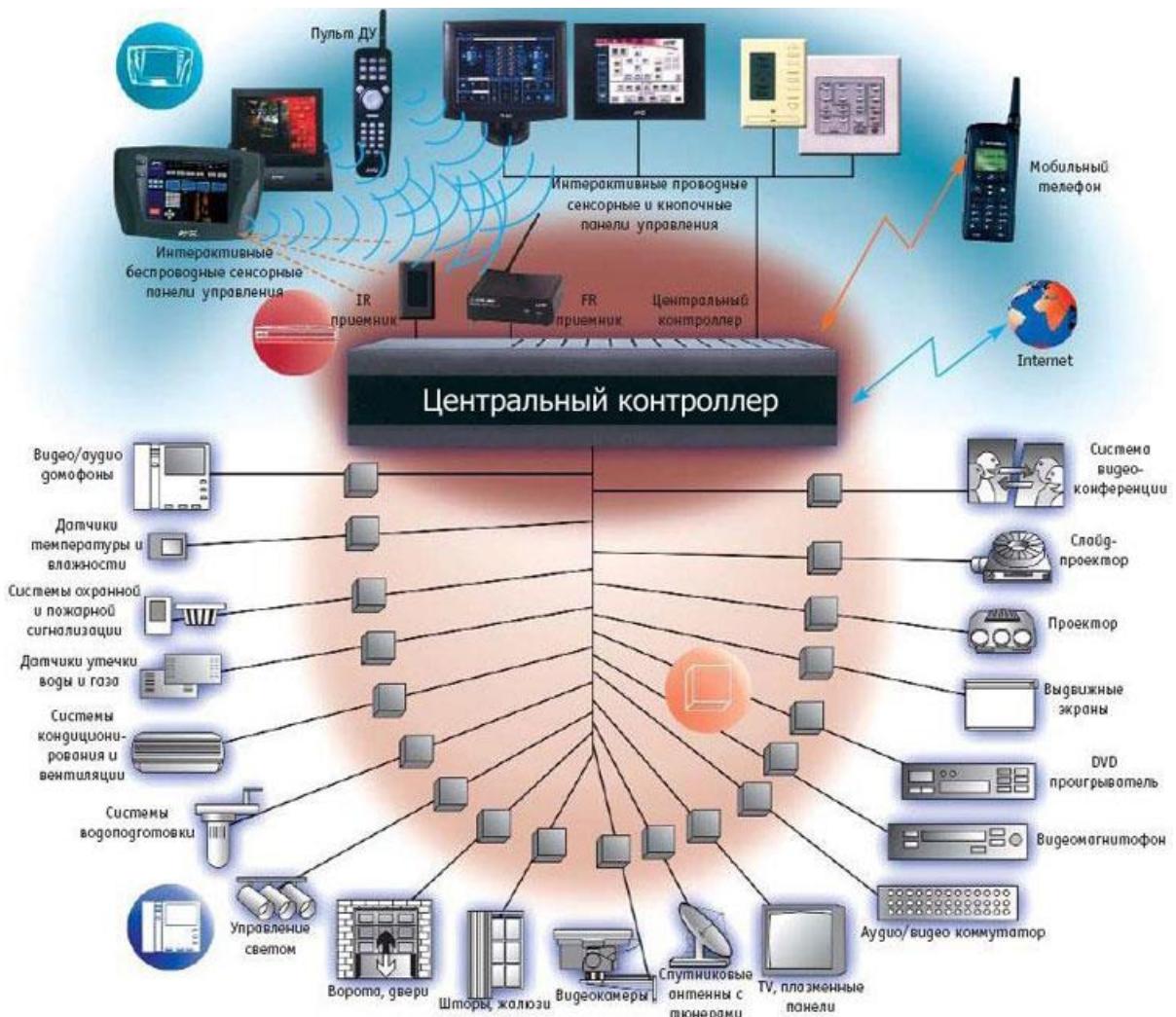


Рисунок 7 - Типовая структурная схема системы «Умный дом»

## 2.2 Выделение основных функциональных особенностей системы «Умный дом»

Для построения проекта необходимо выделить основные особенности системы «Умный дом», на которые больше всего ориентирован потребитель.

По результатам исследования [1], одной из ключевых услуг умного дома, которая пользуется повышенным потребительским спросом и является базовой при установке любой системы, является обеспечение безопасности. Желание иметь в наличии интегрированную систему безопасности вносит основной вклад при принятии решения потенциальным потребителем о покупке компонент умного дома.

Мотивацией потребителя к установке компонент умного дома является и повышение комфорта проживания. Смысл устройств данной категории заключается в том, чтобы взять на себя часть бытовых функций, традиционно выполняемых хозяевами дома, и занять их высвободившееся время.

Также немаловажно создание системы, обеспечивающей энергоэффективность здания. Как правило, она предполагает рациональную организацию энергоснабжения и энергораспределения.

Отдельным вопросом технологического развития умных домов, который активно поднимается в ходе тематических выставок и конференций, является проблема обеспечения связи между устройствами. По сути, основными компонентами умного дома являются контроллер платформы (концентратор (хаб), например, Samsung SmartThings Hub), домашний шлюз и датчики. Для связи могут применяться несколько каналов радиосвязи, сетевые шлюзы (без или с встроенным модемом), сетевой мост (коммутатор). При этом системы могут быть как централизованными, так и децентрализованными. Стоит отметить, что помимо стандартных широко распространенных видов структуры сетей, в качестве реализации может быть связь и по силовой проводке.

Далее нужно выделить основные функции, истекающие из особенностей, что входит в эти функции и какие компоненты должны входить в такую систему.

Функции обеспечения безопасности:

- контроль пожарной опасности: датчики задымления, датчики открытого огня, утечки газа;
- контроль проникновения в дом: датчики движения, разбития стекла, открытия двери, устройство управления освещением, устройство имитации присутствия хозяев;
- контроль водоснабжения: датчики протечки, устройства перекрытия водоснабжения;
- контроль короткого замыкания электросети: устройство обесточивания здания.

Функции обеспечения комфорта:

- контроль и управление освещением: датчики движения, датчики освещенности, устройства управления освещением, устройства управлением жалюзи;
- контроль и управление кондиционированием: датчики влажности, датчики температуры, устройства управления кондиционированием;
- контроль и управление отоплением: датчики температуры, устройства управления отоплением.

Функция организации оптимального энергоснабжения обеспечивается путём интеллектуального распределения времени работы устройств управления и передачи информации пользователю об уровне энергопотребления зданием.

Также важной особенностью системы должна являться возможность работы системы от автономного источника питания в случае сбоев в работе электросети.

Структура сети связи между устройствами определяется из критериев надежности, простоты и дешевизны. Для этого необходимо рассмотреть 4 типа построения сетей в здании и выбрать подходящий:

1. Системы, построенные с использованием силовой электропроводки (PowerLine).



Рисунок 8 - PowerLine адаптеры для создания сети

Главным достоинством такой системы является отсутствие необходимости в прокладке дополнительных кабелей, поскольку для передачи данных применяется обычная электропроводка. Подключение устройств обеспечивается за счёт специальных адаптеров типа «PowerLine». Из недостатков отмечается постоянное наличие помех в сети за счёт работы бытовых приборов, что может вызвать сбои в работе устройств, и высокая стоимость адаптеров.

Данная система для проекта не подходит в силу высокой стоимости и низкой надежности.

## 2. Системы радиошинного типа связи.

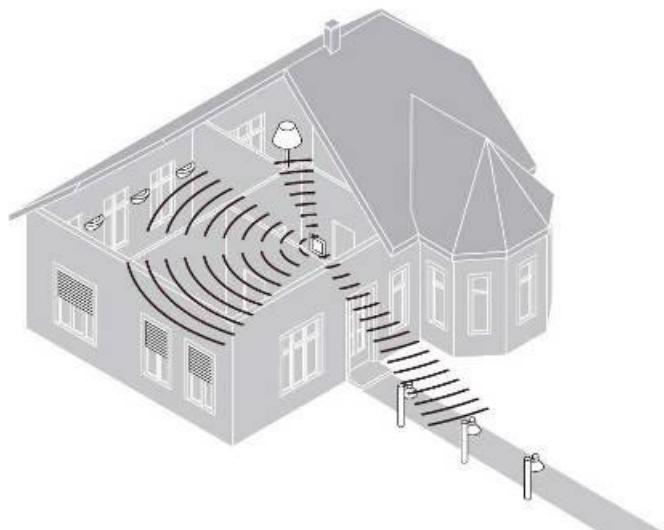


Рисунок 9 - Структура системы радиошинного типа

Такие системы позволяют управлять устройствами с помощью радиосигналов. Основными недостатками системы является ограничение функциональных возможностей по управлению определенными инженерными системами, необходимость постоянного контроля автономного питания и контроля радиосовместимости, возможность занятия радиоканала помехами, вследствие чего вероятны сбои в работе устройств.

Данная система для проекта не подходит ввиду возможности возникновения сбоев в работе.

### 3. С децентрализованным управлением.

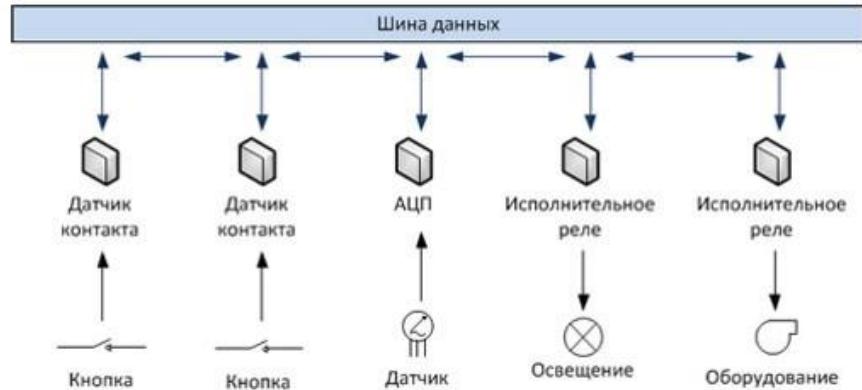


Рисунок 10 - Структурная схема системы с децентрализованным управлением

В данных системах управление осуществляется через сеть так называемых периферийных контроллеров. Каждый из них программируется и обучается отдельно. Данная разработка оптимальна для автономного управления климатом, освещением. Она может работать независимо от головного компьютера и хранить в памяти информацию, необходимую для управления. Однако ее монтаж требуется вести только во время постройки дома, до чистовой отделки.

Данная система для проекта не подходит в силу повышения сложности с программированием каждого устройства, сложности отладки каждого устройства, и резкого повышения стоимости в связи с наличием множества программируемых устройств.

#### 4. С централизованным управлением.

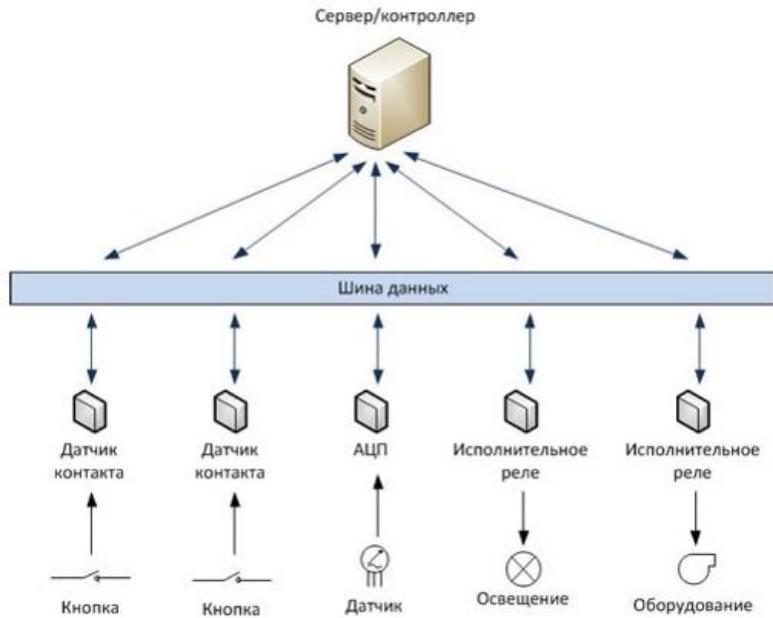


Рисунок 11 - Структурная схема системы с централизованным управлением

Данные системы состоят из панели управления, центрального контроллера и множества исполнительных блоков. Подобные системы позволяют решить задачи практически любой сложности, отличаются продуманной работой с живым видеосигналом, аудиовидеотехникой, системами мультирум, обладают возможностью соединить в единый комплекс все необходимые для жизнеобеспечения системы и свести управление данными системами на один пульт. Однако, при выходе из строя или отключении питания головного компьютера, вся система перестает работать до устранения неисправности.

В итоге из критериев надежности и низкой стоимости для проекта необходимо выбрать проводную и централизованную систему связи между устройствами, основным шлюзом которой будет являться домашний роутер с подключением по Wi-Fi.

Также отключение питания системы будет исключено благодаря возможности автономной работы роутера и главного контроллера.

### **3 РАЗРАБОТКА ИВК ДЛЯ УДАЛЕННОГО УПРАВЛЕНИЯ СИСТЕМОЙ «УМНЫЙ ДОМ»**

#### **3.1 Выбор платформы**

Для решения задачи разработки системы с централизованным управлением, необходимо выбрать платформу управления, которая будет выступать в качестве главного контроллера. Для этого можно рассмотреть самые популярные:

1. Raspberry Pi 3



Рисунок 12 - платформа Raspberry Pi 3

Основные характеристики:

Тип	Model A	Model B
Цена	\$25	\$35
System-on-a-chip (SoC)	Broadcom BCM2835 (CPU + GPU)	
CPU	700 МГц ARM11 (ядро ARM1176JZF-S), возможен разгон до 1 ГГц	
GPU	Broadcom VideoCore IV	
Стандарты	OpenGL ES 1.1/2.0, OpenVG 1.1, Open EGL, OpenMAX	
Аппаратные кодеки	H.264 (1080p30, high-profile);	

	MPEG-2 и VC-1 (лицензия продаётся отдельно)	
<b>Память (SDRAM, общая)</b>	256 Мбайт	512 Мбайт; 256 Мбайт (до 15.10.2012)
<b>Порты USB 2.0</b>	1	2
<b>Видеовыход</b>	1 x HDMI 1.3a (CEC), 1 x RCA (576i/480i, PAL-BGHID/M/N, NTSC, NTSC-J)	
<b>Аудиовыход</b>	Гнездо 3,5 мм, HDMI	
<b>Карт-ридер</b>	SD/MMC/SDIO	
<b>Сеть</b>	-	Ethernet-порт RJ45 10/100 Мбит/с
<b>Интерфейсы</b>	20 x GPIO (SPI, I <sup>2</sup> C, UART, TTL); MIPI CSI-2, MIPI DSI	
<b>Энергопотребление</b>	500 мА (2,5 Вт)	700 мА (3,5 Вт)
<b>Питание</b>	5 В через порт micro-USB или GPIO	
<b>Размеры</b>	85,6x56x21 мм	
<b>Масса</b>	54 г	

Платформа имеет большие вычислительные возможности и компактную компоновку всех необходимых устройств, но для данной работы не подходит в силу высокой сложности программной системы, что, в свою очередь, снизит универсальность в системе «Умный дом». Также платформа имеет цену в районе 25\$.

## 2. Particle Photon

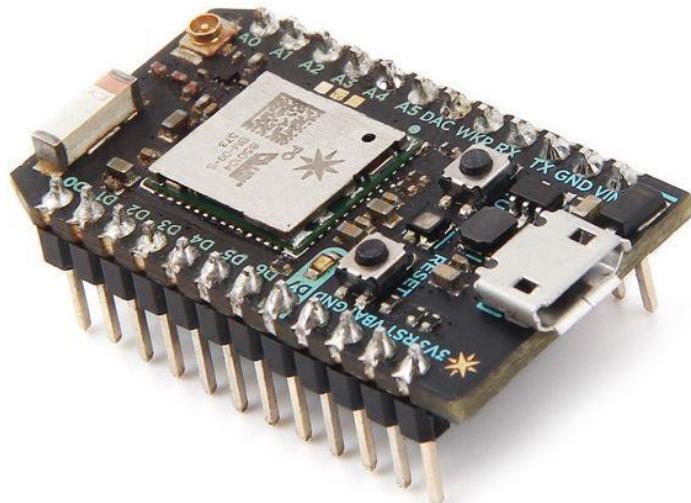


Рисунок 13 - платформа Particle Photon

Particle Photon – это завершенная платформа для разработки, которая включает в себя работу с Wi-Fi и служит для облегчения создания подключаемых устройств. Particle Photon совмещает мощный микроконтроллер ARM Cortex M3 и Wi-Fi чип Broadcom в крошечном, миниатюрного размера модуле, называемом PØ (P-zero). Particle Photon использует микроконтроллер STM32F205RG Cortex M3.

Технические характеристики:

- Particle PØ Wi-Fi модуль
- Broadcom BCM43362 Wi-Fi чип
- 802.11b/g/n Wi-Fi
- STM32F205 120Mhz ARM Cortex M3
- Рабочее напряжение: 3,3 В
- Напряжение питания: 3,6–6 В
- Цифровые входные/выходные выводы: 18
- Аналоговые входные выводы (АЦП): 5
- Аналоговые выходные выводы (ЦАП): 1
- Порты UART: 1
- Порты SPI: 2
- Порты I2C: 1

- Порты CAN: 1
- Флеш-память: 1 Мбайт
- SRAM: 128 Кбайт
- Тактовая частота: 120 МГц
- Размеры: 36,58 x 20,32 мм
- Установленный на плате RGB светодиод индикации состояния
- 18 портов GPIO смешанных сигналов и дополнительные периферийные устройства

- Открытая архитектура
- Операционная система реального времени (FreeRTOS)

Данная платформа также имеет сложную операционную систему, а количество выводов по сравнению с остальными платформами невелико, что повлияет на возможности расширения системы.

### 3. Arduino

Платформа Arduino пользуется большим спросом среди людей, не специализирующихся на программировании микроконтроллеров (прежде всего, за счет понятного интерфейса среды программирования и наличия большого количества поддерживаемых библиотек, упрощающих процесс разработки программ).

Основные преимущества платформы:

- полностью открытая архитектура, за счёт чего обеспечивается возможность для создания и развития своего программного обеспечения (ПО);
- простой и в то же время гибкий язык программирования;
- постоянное развитие и усовершенствование программных проектов за счёт популярности;
- низкая стоимость контроллеров (в диапазоне от 150руб. до 2000руб.);
- большая разновидность платформ для различных применений;
- компактность готовых решений;

- универсальность, способность решать любые задачи, ограниченные только производительностью контроллера.

Arduino имеет множество разновидностей аппаратной части. Для выбора подходящей платформы необходимо для каждой составить список возможностей.

- Arduino UNO: Самая популярная платформа на контроллере ATmega328P; Имеет 6 аналоговых и 14 цифровых; Флэш-память 32 КБ, ОЗУ 2 КБ; USB-порт type B; EEPROM 1 КБ; Частота до 16 МГц. Плата продается в двух вариантах контроллера связи по USB: на CH340 стоимостью около \$3 и на Atmel ATMEGA 8U2 стоимостью \$5;
- Arduino Nano: возможности те же, что и в UNO, но более компактная по размерам по причине более интегрированного контроллера. Стоимость около \$2;
- Arduino Leonardo: На микроконтроллере ATmega32u4, 12 аналоговых входов и 20 цифровых портов, 32 КБ Флэш-память, MicroUSB порт. Стоимость около \$7-8;
- Arduino Mega 2560: На микроконтроллере Atmega2560, 2 аналоговых выхода, 16 аналоговых входов и 54 цифровых порта, 256 КБ Флэш-память, ОЗУ 8 КБ, UART 4 шт. Стоимость около \$7-8;
- Arduino Due: На микроконтроллере Cortex AT91SAM3X8E, 54 цифровых порта, в числе которых 12 с ШИМ, 12 аналоговых входов и 2 аналоговых выхода, 512 КБ Флэш-память, ОЗУ 96 КБ, 2 MicroUSB порта. Имеет самую высокую производительность. Вообще говоря, контроллер здесь уже не AVR, а Cortex, построенный на архитектуре ARM, поэтому имеет рабочее напряжение 3,3 В (а не 5 В). Процесс программирования иной, в отличии от AVR. Стоимость около \$15.

Исходя из составленного списка, можно заключить, что наиболее подходящей будет являться Arduino Mega 2560 благодаря большому количеству портов, за счет которых система будет иметь широкие возможности расширения,

средние параметры производительности, которые полностью удовлетворят запросам функциональности, а также приемлемую стоимость около \$8.

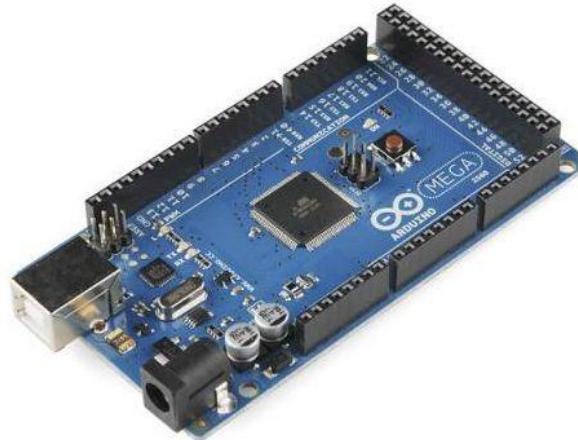


Рисунок 14 - платформа Arduino Mega 2560

### 3.2 Обзор всех возможностей платформы Arduino

Для более полного представления возможностей платформы необходимо составить общий обзор архитектуры микроконтроллера AVR.

Atmel AVR представляет собой семейство универсальных 8-разрядных микроконтроллеров на основе общего ядра с различными встроенными периферийными устройствами. Возможности МК AVR позволяют решить множество типовых задач, возникающих перед разработчиками радиоэлектронной аппаратуры [4].

Структурная схема AVR изображена на рисунке 15.

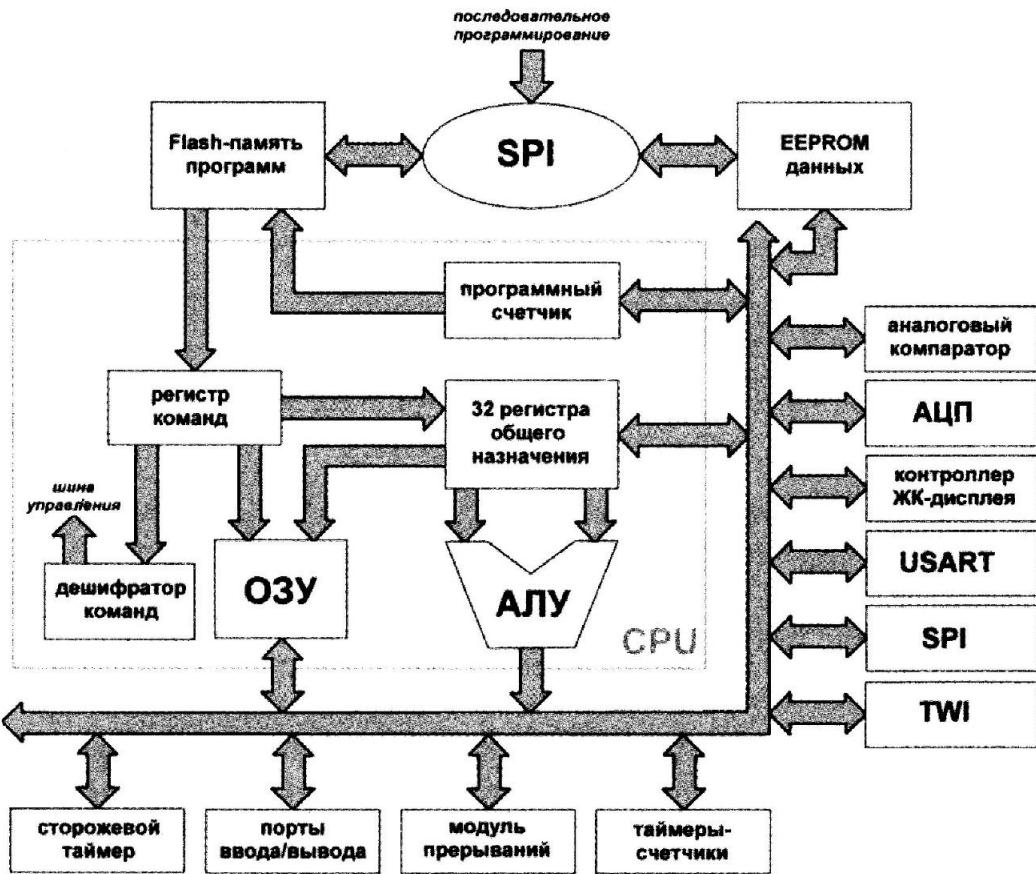


Рисунок 15 - Общая структурная схема микроконтроллеров AVR

### Особенности микроконтроллеров AVR:

- производительность порядка 1 MIPS/МГц. MIPS (Millions of Instructions Per Second, миллион команд в секунду) - характеристика производительности быстродействия процессора. Показывает число определённых инструкций, выполняемых процессором за одну секунду;
- усовершенствованная RISC-архитектура. Концепция RISC (Reduced Instruction Set Computing, вычисления с сокращенным набором команд) предполагает наличие набора команд, состоящего из минимума компактных и быстро выполняющихся инструкций. Концепция RISC упрощает устройство ядра и ускоряет его работу - типовая инструкция выполняется за один такт;
- раздельные шины памяти команд и данных. AVR имеет архитектуру, в которой области памяти программ и данных разделены. Раздельные шины этих областей памяти значительно ускоряют выполнение программ: данные и команды могут выбираться одновременно;

- flash-память программ с возможностью внутрисистемного перепрограммирования и загрузки через последовательный канал прямо в готовой схеме;
- встроенные средства для обработки аналоговых сигналов: аналоговый компаратор и многоканальный 10-разрядный АЦП;
- последовательные интерфейсы SPI, TWI(I2C) и UART(USART), для реализации аппаратным обменом данными с большинством датчиков и др. внешних устройств;
- возможность работы при частоте процессора от 1 МГц до 16 МГц;
- диапазон входных напряжений, рекомендуемый: 7-12 В, предельный: 6-20 В;
- различные режимы энергосбережения. Выход из «спящих» режимов по сторожевому таймеру или по внешним прерываниям.

Периферийные устройства контроллеров AVR:

- от 1 до 7 8-разрядных автономных портов ввода/вывода с возможностью программной установки высокого входного сопротивления;
- 3 таймера-счётчика, один 16-разрядный, два 8-разрядных. Имеют возможность предварительной загрузки значений и могут работать от тактовой частоты процессора непосредственно, либо от внешнего опорного сигнала. Допускают работу в PWM режимах, т.е. в качестве 8 или 16-битовых широтно-импульсных модуляторов (ШИМ). Поддерживают многоканальный, независимый режим ШИМ;
- аналого-цифровой преобразователь (АЦП). Многоканальный (подключенный к мультиплексору с несколькими входами) 8-битный с возможностью замены аналогового компаратора. Входы мультиплексора могут работать по отдельности или объединяться в пары для измерения дифференциальных сигналов.

Поддерживаемые последовательные порты:

1. UART (Universal Asynchronous Receiver-Transmitter), или универсальный асинхронный приёмопередатчик (УАПП). Узел вычислительных устройств, предназначенный для организации связи с другими цифровыми

устройствами. Преобразует передаваемые данные в последовательный вид так, чтобы было возможно передать их по одной физической цифровой линии другому аналогичному устройству. Представляет собой логическую схему, с одной стороны подключённую к шине вычислительного устройства, а с другой имеющую два или более выводов для внешнего соединения. Используется для передачи данных через последовательный порт компьютера.

2. SPI (Serial Peripheral Interface), или последовательный периферийный интерфейс. Является полнодуплексной шиной - данные передаются одновременно в обе стороны. Скорость работы шины лежит в пределах 1-50 МГц. Использует 4 линии для обмена данными: SCLK - Serial Clock - тактовый сигнал (от ведущего), MOSI — Master Output, Slave Input - данные от ведущего к ведомому, MISO - Master Input, Slave Output: данные от ведомого к ведущему, SS — Slave Select: выбор ведомого; устанавливается ведущим.

3. TWI (I2C) - Inter-Integrated Circuit или «Схема внутренней связи». Позволяет наращивать функционал и добавлять новые блоки, не задействуя новые выводы микроконтроллера. Макс. скорость передачи 100 Кбит/с и макс. кол-во устройств 128. Структурная схема I2C изображена на рисунке 16.

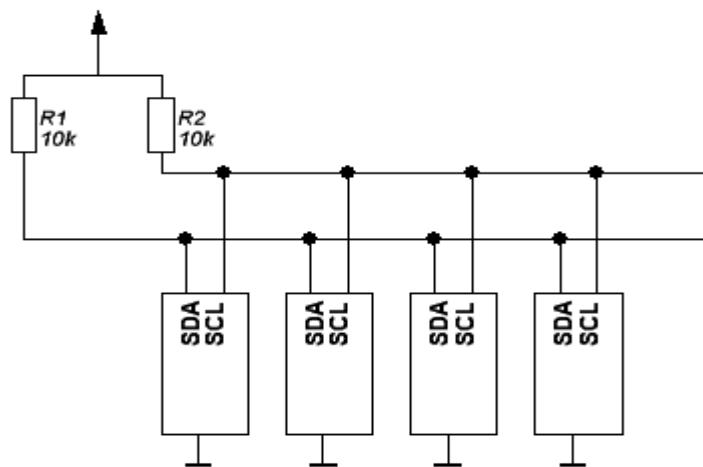


Рисунок 16 - Структурная схема I2C

4. Универсальный последовательный интерфейс USI (Universal Serial Interface). Представляет собой обычный сдвиговый регистр (USIDR) с регистром управления (USICR) и статус (USISR) без каких-либо буферов данных. В его состав входит 4-битный счетчик тактовых импульсов, который управляется от

регистра USISR. С интерфейсом связано аппаратное прерывание. Его возможно использовать в качестве трехпроводного SPI, в качестве двухпроводного TWI, для имитации UART, и для остальных применений. Сдвиг в регистре USIDR управляется либо программно, либо от прерывания переполнения TimerO, либо от внешнего источника. Структурная схема USI изображена на рисунке 17.

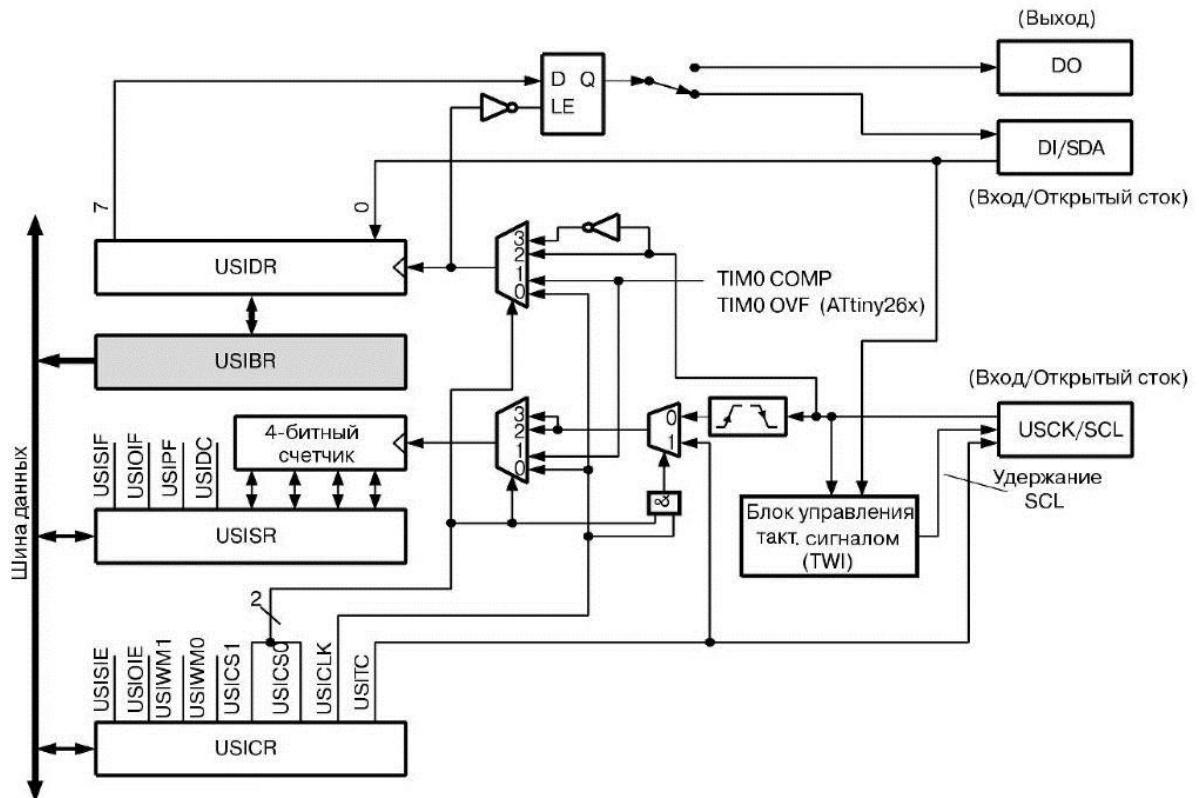


Рисунок 17 - Структурная схема интерфейса USI

Поддерживаемые режимы энергосбережения контроллера:

1. **Idle mode.** Режим, в котором останавливается ядро процессора и устройство управления выборкой команд из памяти. При этом все периферийные устройства продолжают функционировать;
2. **Power Down mode.** Режим, в котором останавливаются все узлы микроконтроллера, кроме сторожевого таймера, системы обработки внешних асинхронных прерываний и модуля TWI. Выход из режима осуществляется от внешних прерываний. Потребление в этом режиме составляет 10-20 мкА;
3. **Power Save mode.** Такой же, как и Power Down mode, но в отличии от последнего, при PS mode остаётся включенным таймер, работающий в асин-

хронном режиме от тактового генератора. Выход из режима осуществляется по прерываниям таймера-счётчика. Обычно режим используется для реализации часов реального времени;

4. Standby mode. Режим энергосбережения, выход из которого осуществляется за 6 тактов.

Потребление МК AVR при напряжении 5 В и частотах 4-8 МГц лежит в пределах до 10 мА. Как правило сильно зависит от напряжения питания и тактовой частоты. На рисунке 18 для наглядности приведен график зависимости тока потребления от напряжения питания:

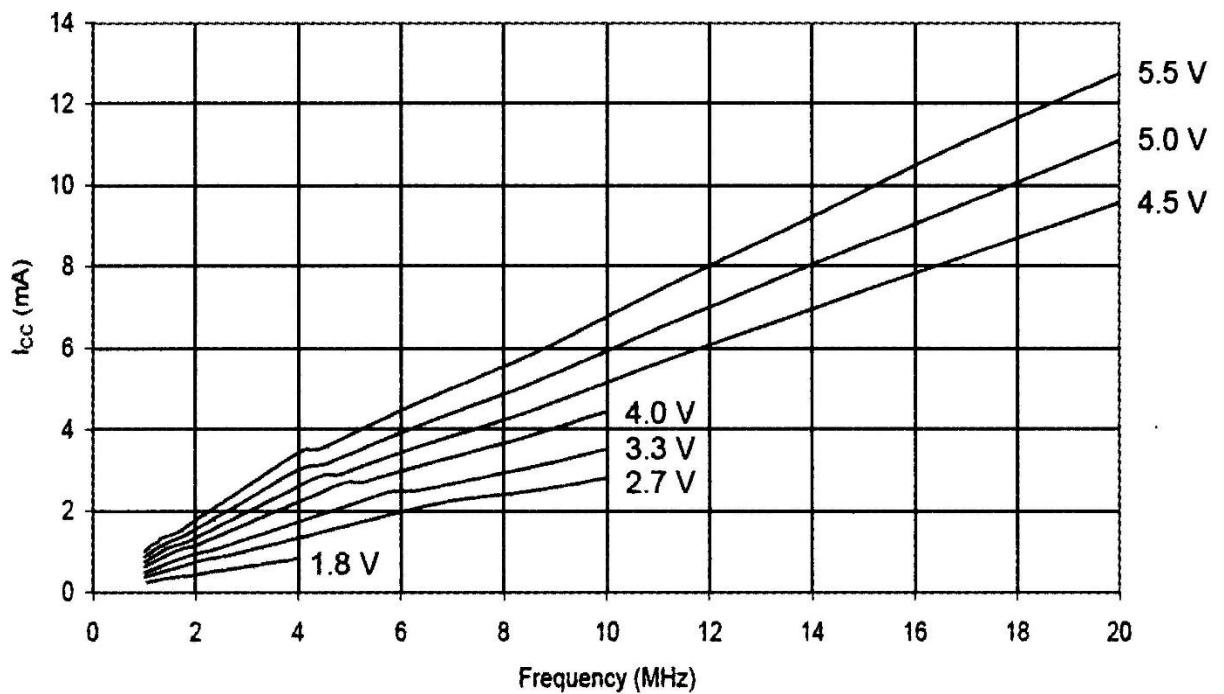


Рисунок 18 - График зависимости тока потребления от напряжения питания и тактовой частоты МК AVR

На графике видно, что микроконтроллер в номинальном режиме работы потребляет десятки миллиампер, благодаря чему можно осуществить поддержание работы системы в автономном режиме.

В итоге из вышеприведенного обзора можно наглядно оценить универсальность и надежность платформы, 4 типа интерфейса позволит подключать множество различных датчиков, а режимы энергосбережения позволяют системе

эффективно использовать энергию, потребляемую контроллером, в том числе и при работе от автономного источника энергии.

### **3.2 Обеспечение удаленного управления с помощью Arduino**

В качестве устройства для связи платформы с домашним роутером выбран Wi-Fi модуль NodeMCU на основе микроконтроллера ESP12E, как более мощное устройство, способное работать по протоколу связи TCP.

Стоимость данного модуля составляет около \$6. Модуль изображен на рисунке 19:



Рисунок 19 - Модуль NodeMCU

Характеристики модуля [10]:

- режимы: P2P (клиент), soft-AP (точка доступа);
- номинальное напряжение: 3,3 В;
- максимальная выходная мощность: 19,5 дБ·мВт (89 мВт);
- беспроводной интерфейс: Wi-Fi 802.11 b/g/n 2,4 ГГц;
- входное напряжение: 3,7–20 В;
- портов ввода-вывода свободного назначения: 11;
- максимальный потребляемый ток: 220 мА;
- частота процессора: 80 МГц;

- объём ПЗУ: 64 КБ;
- объём ОЗУ: 96 КБ;
- габариты:  $45 \times 30$  мм.

Данный модуль подключается к Arduino с помощью платы расширения посредством интерфейса связи UART (Serial - порт), но для обеспечения электрической совместимости необходимо преобразование уровней сигналов из 3,3 В (ESP12E) в 5 В (Arduino) с помощью платы преобразования уровней, которая изображена на рисунке 20:

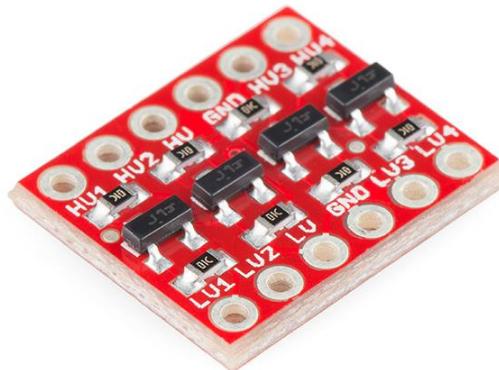


Рисунок 20 - Преобразователь уровней 3.3 В - 5 В

Стоимость преобразователя \$3.

Передача данных через интерфейс UART осуществляется посредством AT команд в режиме SoftwareSerial (для чего в код программы добавляется соответствующая библиотека и в программе используется класс Serial).

Схема подключения Arduino к NodeMCU изображена на рисунке 21:

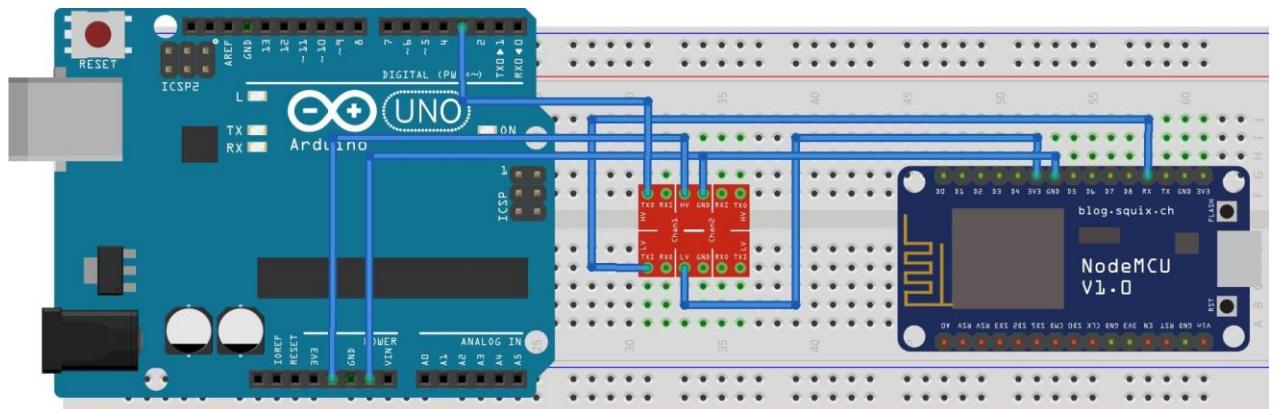


Рисунок 21 - Схема подключения NodeMCU к Arduino

Принципиальная схема подключения Arduino к NodeMCU изображена на рисунке 22:

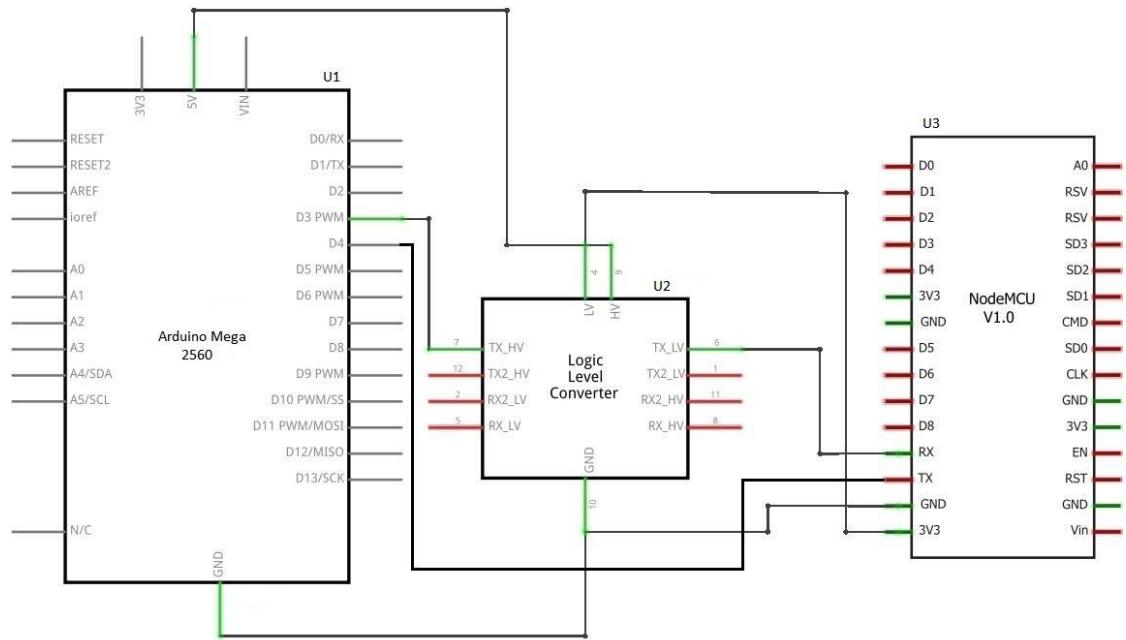


Рисунок 22 - Принципиальная схема подключения NodeMCU к Arduino

Как правило, модуль имеет возможности программирования и подключения устройств, что и Arduino, но следует выделить некоторые аспекты, благодаря которым выбор сделан в пользу Arduino:

- программирование модуля NodeMCU значительно усложнится в силу меньшего количества существующих библиотек, вследствие чего ухудшается универсальность и повышается сложность в наладке разрабатываемой системы;
- решаемые системой задачи лучше распределить на несколько устройств для повышения надежности системы (следствие выбора централизованной системы, рассмотренного выше).

Для обеспечения удаленного управления Arduino со смартфона, самым оптимальным будет выбор облачных сервисов, которые позволяют управлять системой визуально с помощью смартфона, на который устанавливается приложение. Как правило, эти сервисы очень просты в программировании нужных функций и реализации связи программного кода в устройстве без ущерба функциональности [6].

## 1. Blynk

Самый популярный и простой сервис с особенным интерфейсом, позволяющим подключаться как к физическим пин-выводам, так и к виртуальным для гибкости логики обработки. Из преимуществ можно отметить подробную документацию по программированию. Из недостатков стоит отметить, что приложение бесплатно только для ограниченного набора компонентов, устанавливаемых на экране (как правило это примерно 1 график и 5 элементов управления). Если появится необходимость в расширении, то стоимость выйдет в среднем в \$4. Интерфейс приложения Blynk изображен на рисунке 23:

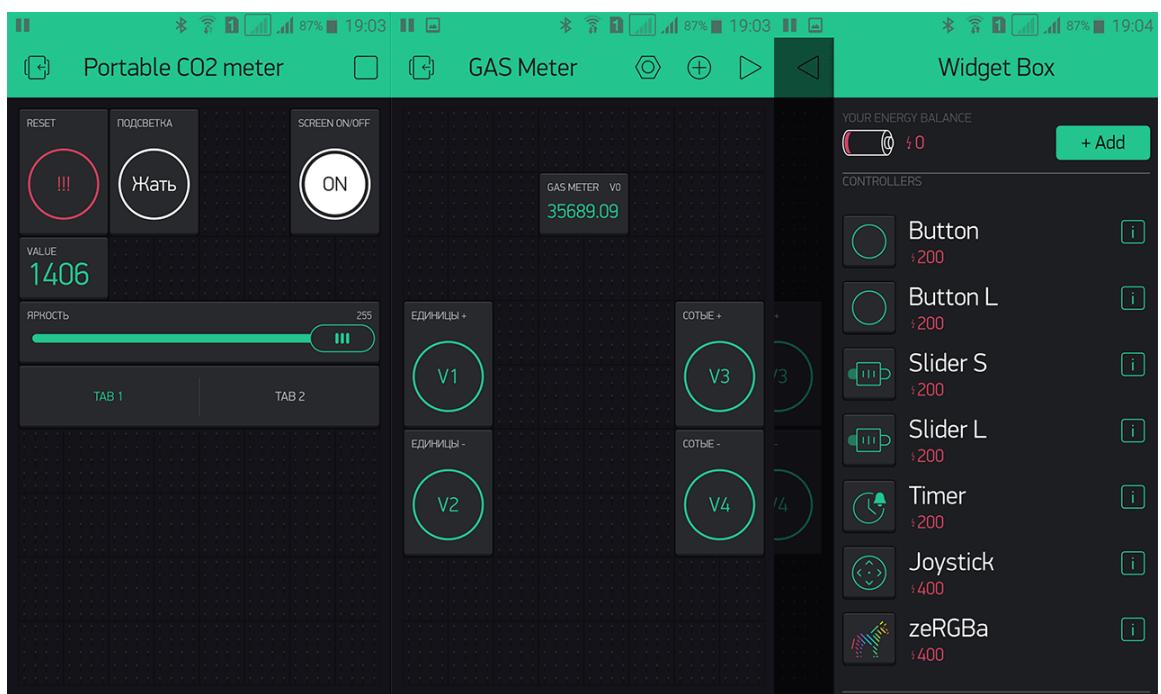


Рисунок 23 - Интерфейс приложения Blynk на смартфоне

## 2. Cayenne

Сервис идентичный Blynk, но в отличии от него позволяющий собирать компоненты управления через веб-сайт и полностью бесплатный. Связь приложения с устройством будет осуществляться аналогично Blynk, интеграцией кусков кода бизнес-логики в среду разработки. Но первоначально проект предназначался для минимального количества усилий на программирование на стороне устройства клиента. Поэтому в Cayenne можно найти множество плат и вариантов подключения к сети, среди которых различные устройства конкрет-

ных производителей. С одной стороны, разнообразие снижает усилия на программирование, с другой, снижает универсальность системы.

Интерфейс приложения Cayenne изображен на рисунке 24:

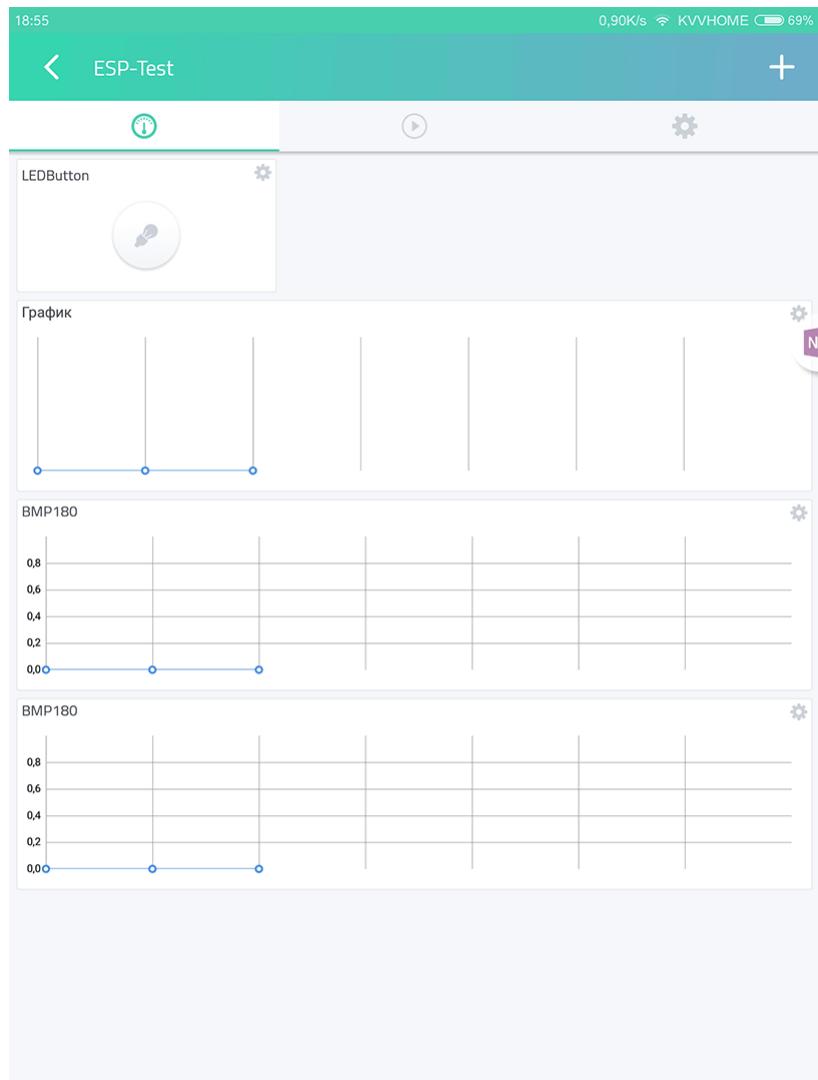


Рисунок 24 - Интерфейс рабочей панели Cayenne на планшете

### 3. Virtuino

Аналогично Blynk, сервис предполагает управление элементами только с помощью мобильного приложения. Из недостатков были отмечены скучность в выборе компонентов и нестабильность в работе при неустойчивой связи смартфона.

В итоге следует выбрать сервис Blynk, как наиболее надежный, с высокой универсальностью, но имеющий вполне приемлемую стоимость.

Программа управления умным домом через сервис Blynk посредством Wi-Fi модуля NodeMCU находится в приложении Б.

### 3.3 Разработка и выбор компонентов устройства автономного питания

Устройство автономного питания будет состоять из платы контроля заряда на контроллере TP4056 (рисунок 25), сетевого адаптера AC-DC 220В-5В (рисунок 27) и батареи Li-poly 3,7 В, емкостью 2000 mAh (рисунок 28). Емкость батареи выбрана из расчета оптимального соотношения продолжительность работы/цена.

Плата будет подключена к сетевому адаптеру и отслеживать отключение электроэнергии. В случае отключения, питание будет подаваться от батареи.

Для автономного питания Arduino и датчиков необходим преобразователь напряжения DC-DC 3,7(5)-12 В (рисунок 26):

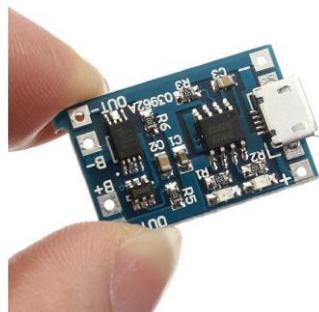


Рисунок 25 - Плата с контроллером заряда P4056



Рисунок 26 - Преобразователь напряжения

5(3,7)-12 В



Рисунок 27 - Адаптер питания 220 В - 5 В 2 А



Рисунок 28 - Батарея Li-Poly 3,7 В 2 А

Устройство автономного питания собирается по принципиальной схеме, изображенной на рисунке 29:

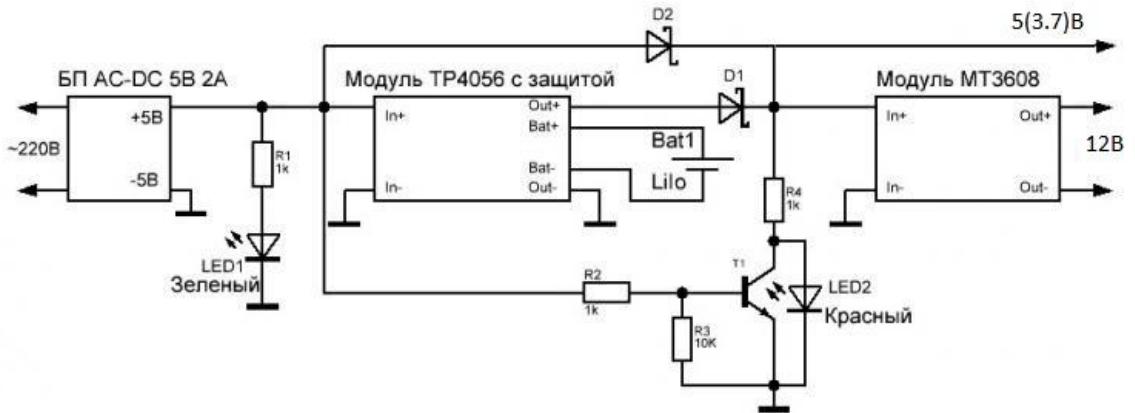


Рисунок 29 - Принципиальная схема устройства автономного питания

Устройство снабжено LED индикаторами для отображения состояний: зеленый-заряжено, красный-процесс заряда.

Стоимость компонентов составляет:

- плата с контроллером TP4056: \$3;
- сетевой адаптер AC-DC: \$3;
- батарея Li-poly 3,7 В 2000 mAh: \$10;
- преобразователь напряжения DC-DC: \$1;
- другие мелкие компоненты и плата: ~\$2.

### 3.4 Выбор измерительных датчиков для интеграции в систему

Датчики будут выбираться из соображений низкой стоимости, высокой совместимости с платформой, но без ущерба необходимой функциональности для системы.

1. Датчик газа и задымления MQ-9 (рисунок 30):



Рисунок 30 - Датчик газа MQ-9

Датчик газа, построенный на базе газоанализатора MQ-9 для обнаружения наличия в окружающем воздухе углеводородных газов (метан, н-бутан, пропан), дыма (взвешенные частицы, являющиеся результатом горения) и водорода.

В датчике встроен нагревательный элемент, необходимый для получения химической реакции. Для получения стабильных показателей сенсор один раз прогревается в течение 48 часов.

Имеет 3 вывода для передачи аналогового сигнала.

Характеристики:

- напряжение питания 1,5/5 В;
- потребляемый ток 70 мА;

Диапазон измерений:

- водород: 0,3 – 5 промилле;
- метан: 5 – 20 промилле;
- пары спиртов: 0,1 – 2 промилле;
- пропан: 0,2 – 5 промилле;
- бутан: 0,3 – 5 промилле.

Стоимость датчика составляет \$1,5.

Схема подключения датчика приведена на рисунке 31:

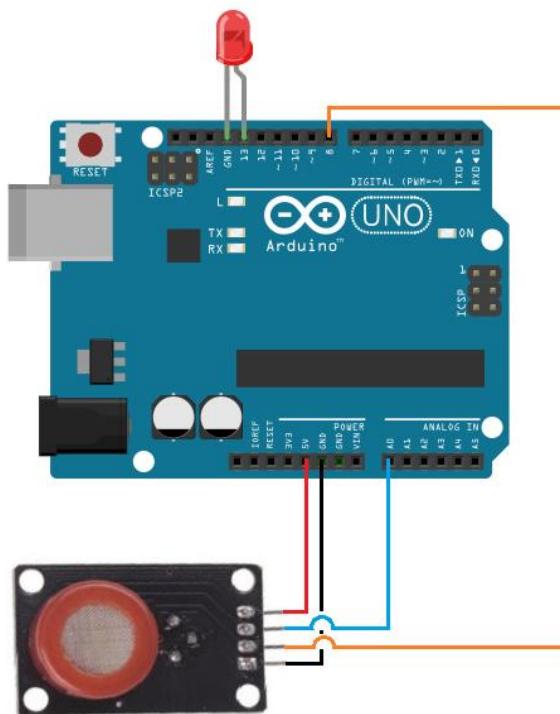
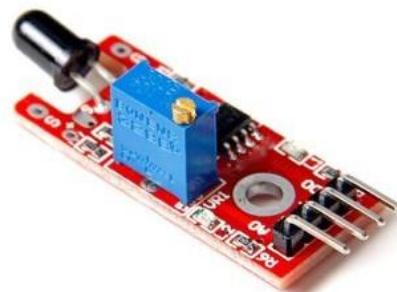


Рисунок 31 - Схема подключения датчика газа MQ-9 к Arduino

## 2. Датчик открытого огня KY-026 (рисунок 32):



### Рисунок 32 - Датчик огня KY-026

KY-026 - инфракрасный датчик пламени, реагирующий на инфракрасное излучение в диапазоне длин волн 760-1100 нм., свойственное пламени свечи, например. На плате датчика имеется 2 светодиода - индикации питания и индикации обнаружения огня. Для лучшей фильтрации посторонних источников света, применяют алгоритм детектирования низкой частоты. Это возможно благодаря тому, что пламя свечи меняет свою интенсивность с частотой 15-20 Гц.

Имеет 3 вывода для передачи цифрового сигнала.

Характеристики:

- напряжение питания: 3-5,5 В;
- угол обнаружения пламени: 60 градусов;
- размеры  $36 \times 16$  мм.

Стоимость датчика составляет \$1,5.

Схема подключения датчика приведена на рисунке 33:

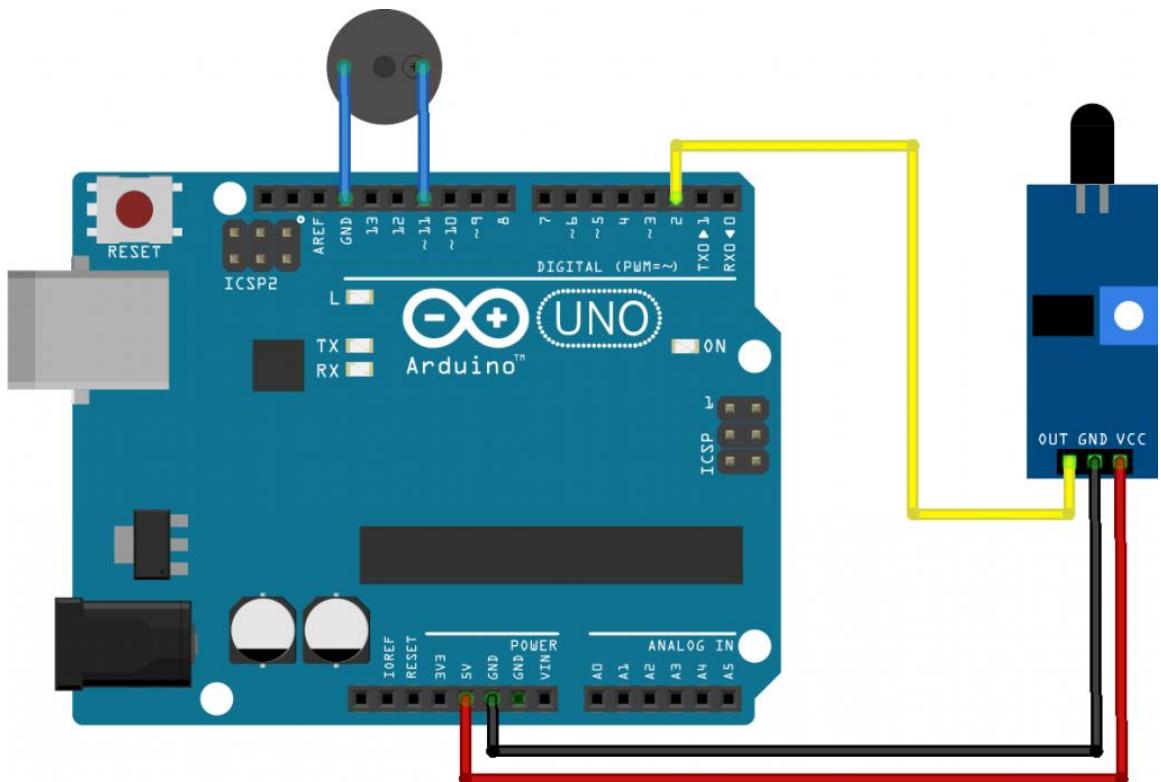


Рисунок 33 - Схема подключения к Arduino

Принципиальная схема подключения датчика приведена на рисунке 34:

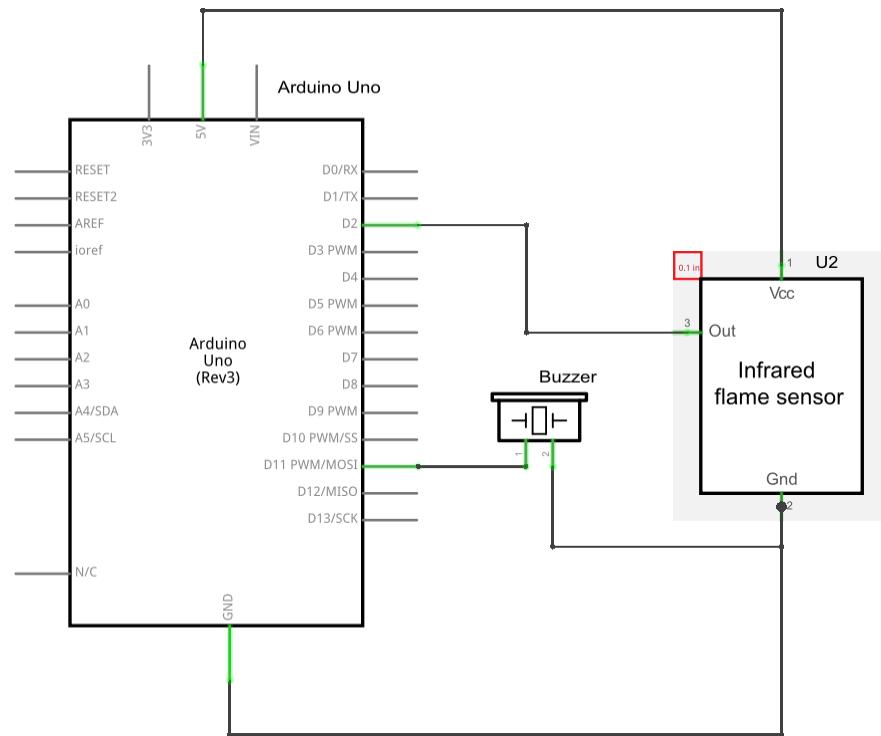


Рисунок 34 - Принципиальная схема подключения к Arduino

### 3. Датчик движения HC-SR501 (рисунок 35):

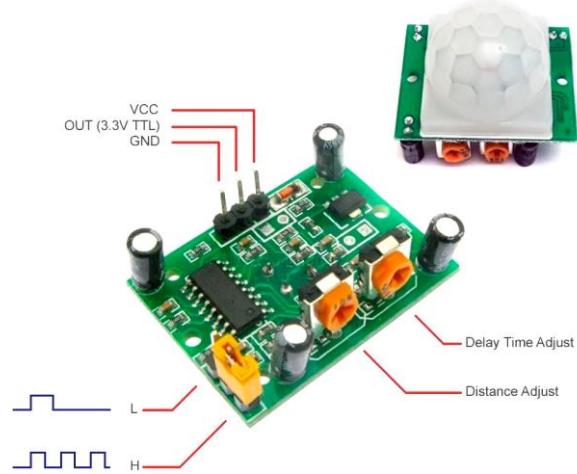


Рисунок 35 - Датчик HC-SR501

Датчик использует явление пироэлектрического эффекта - появления электрического поля диэлектрика при изменении температуры.

## Характеристики:

- диапазон рабочего напряжения: 4,5-20 В;
- потребляемый ток 50 мкА;
- диапазон рабочих температур: от -15 до +70 градусов Цельсия;
- напряжение на выходе 3,3 В;
- размер: 32 × 24 мм.;
- угол обнаружения: 110 градусов;
- дистанция срабатывания (регулируемая): 3-7 метров

Датчик имеет 3 вывода для передачи цифрового сигнала 3,3 В. Но у Arduino цифровые входы используют 5 В, поэтому подключение будет происходить к аналоговому входу Arduino.

Стоимость датчика составляет \$1,5.

Схема подключения датчика приведена на рисунке 36:

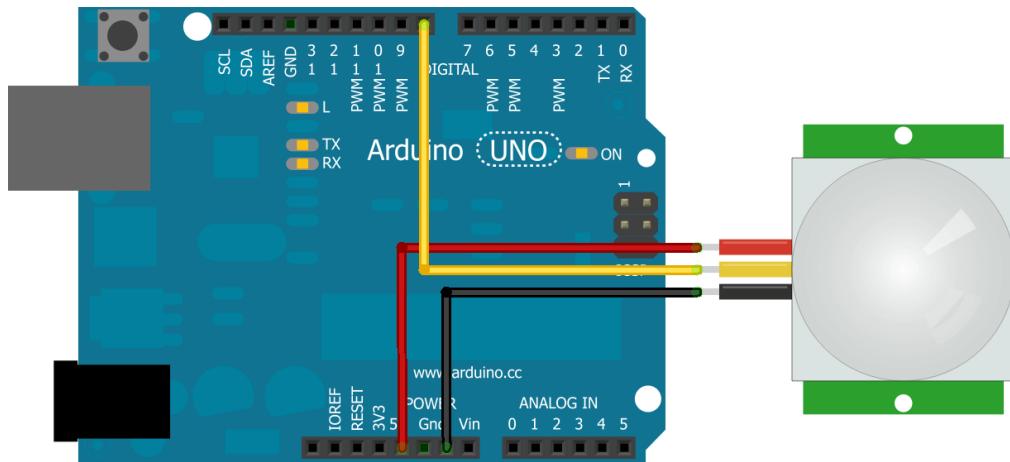


Рисунок 36 - Схема подключения HC-SR507

4. Датчик разбития стекла (рисунок 37):



Рисунок 37 - Проводной датчик разбития стекла

Особенности:

- программное 8-битное микропроцессорное управление цифровым сигналом с частотой 8 МГц, инфраструктурный анализ давления, цифровая фильтрация сигнала;
- уникальный акустический микрофонный корпус и корпус конструкции;
- ручные настройки чувствительности, во избежание ложной тревоги;
- устойчив к воздействию высокой температуры, коррозии.

Технические характеристики:

- ток потребления (ждущий режим): 5 мА;
- ток потребления (При воздействии): 80 мА;
- желтый светодиод: указывает на уровень шума окружающей среды;
- красный светодиод: отображение фиксации сигнала по триггеру;
- температурный диапазон работы: от 10 до 50 °C;
- конструкция корпуса: ABS огнестойкий пластик.

В комплект входит:

- детектор стекла;
- руководство пользователя на английском;

– упаковка.

Стоимость датчика составляет \$10.

5. Датчик открытия двери (рисунок 38) и схема подключения (рисунок 39):



Рисунок 38 - Герконовый датчик

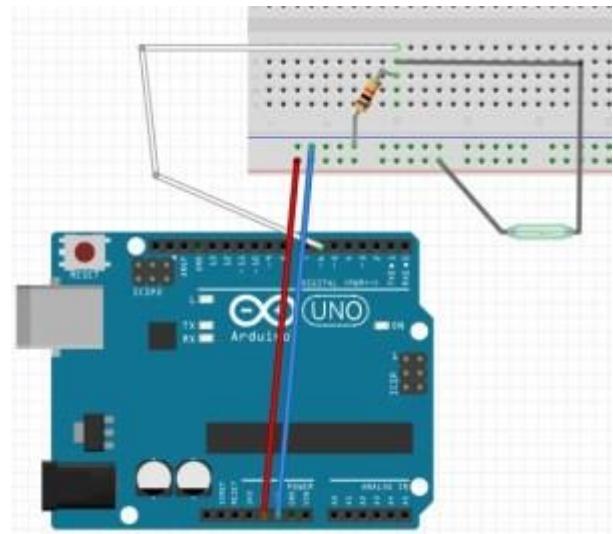


Рисунок 39 Схема подключения к Arduino

Датчик открытия двери состоит из герметизированного магнитоуправляемого контакта, который обычно встраивается в двери.

Стоимость датчика составляет \$1.

6. Датчик протечки (рисунок 40):



Рисунок 40 - Комплектация датчика протечки

Позволяет определить протечку проявлением влаги на сенсорной плате. Плата представляет собой переменный резистор, замыкаемый водой в разных местах, что вызывает изменение сопротивления.

Вторая часть датчика представляет собой сдвоенный компаратор на микросхеме LM393 для преобразования значения сенсора в аналоговый и цифровой сигналы

Датчик имеет возможности отслеживания влаги в двух режимах:

- аналоговый, для определения уровня влаги изменением напряжения на аналоговом выходе в диапазонах от 0 до 5 В;
- цифровой, для индикации превышения допустимого уровня влаги сигналом 5 В. Уровень порога срабатывания регулируется подстроенным резистором на плате.

Стоимость датчика составляет \$1.

Схема подключения приведена на рисунке 41:

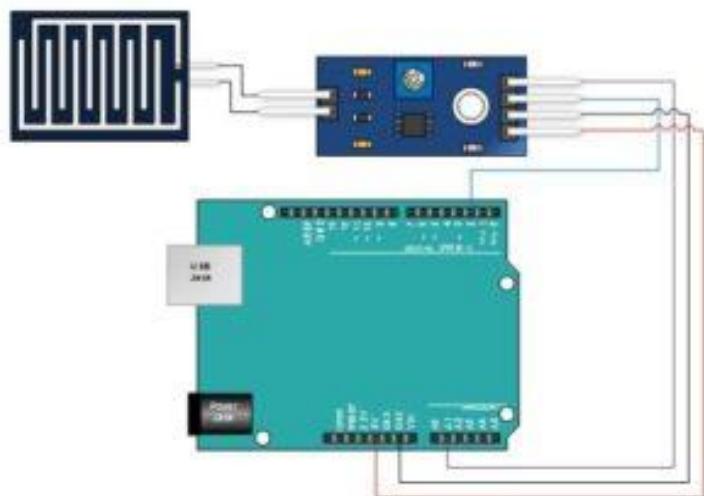


Рисунок 41 - Схема подключения датчика влаги к Arduino

7. Датчик уровня освещенности CY-30 (рисунок 42):

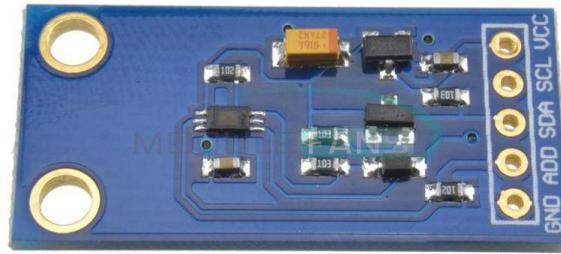


Рисунок 42 - Датчик освещенности

Модуль платы CY-30 включает в себя датчик освещенности (люксметр) BH1750 и несколько дополнительных транзисторов для преобразования уровней сигналов и возможности подключения к линиям данных SDA и SCL. Сам датчик способен передавать данные по интерфейсу I2C.

Стоимость датчика составляет \$1,5.

Схема подключения приведена на рисунке 43:

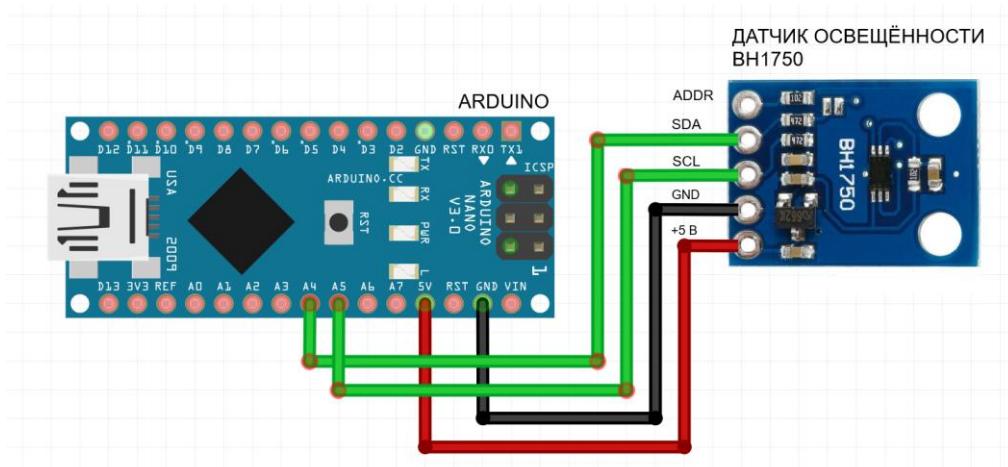


Рисунок 43 - Схема подключения датчика освещенности к Arduino

8. Датчик влажности и температуры DHT22 (рисунок 44):

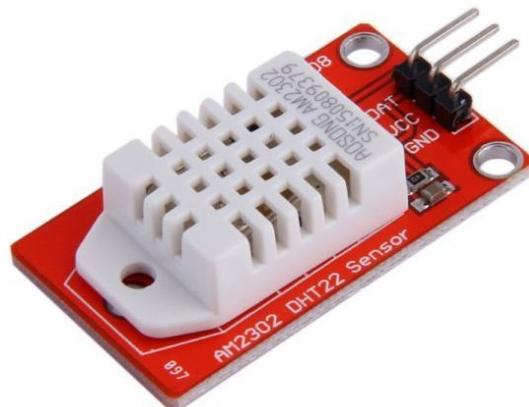


Рисунок 44 - Датчик DHT22

DHT22 состоит из емкостного датчика влажности, термистора и простого 8-битного микроконтроллера для преобразования аналоговых сигналов в цифровой и передает данные через интерфейс 1-Wire.

Характеристики:

- напряжение питания от 3 до 5 В;
- максимальный ток потребления 2,5 мА;
- диапазон измерения уровня влажности от 0 % до 100 %, при точности измерений от 2 % до 5 %;
- диапазон измерения температуры от -40 до 125 °C при точности измерений  $\pm 0.5$  градусов;
- частота измерений до 2-х в секунду;
- размер  $15,1 \times 25 \times 7,7$  мм.

Стоимость датчика составляет \$4.

Схема подключения датчика приведена на рисунке 45:

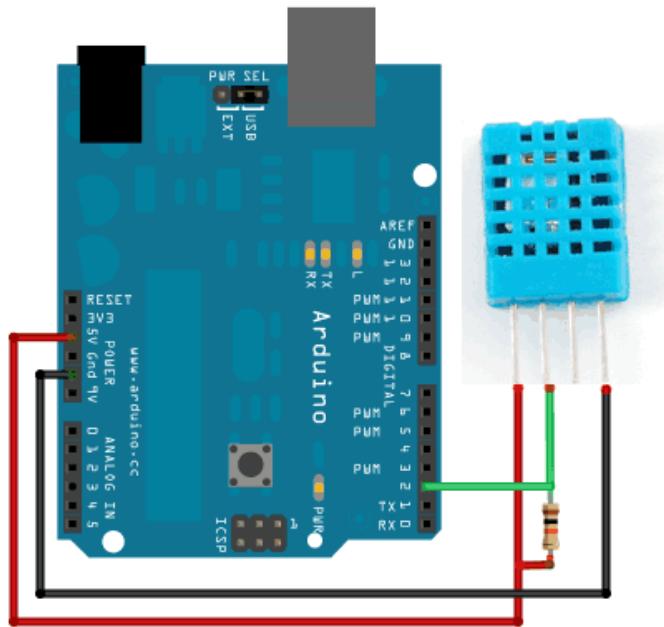


Рисунок 45 - Схема подключения DHT22 к Arduino

9. Датчик переменного тока SEL98100NI (рисунок 46):



Рисунок 46 - Датчик тока

Неинвазивный датчик тока, с максимальным пределом измерения до 100А. Принцип работы основан на измерении электромагнитного поля нагруженного провода, по которому течет ток. Подключается к Arduino по схеме, изображенной на рисунке 47:

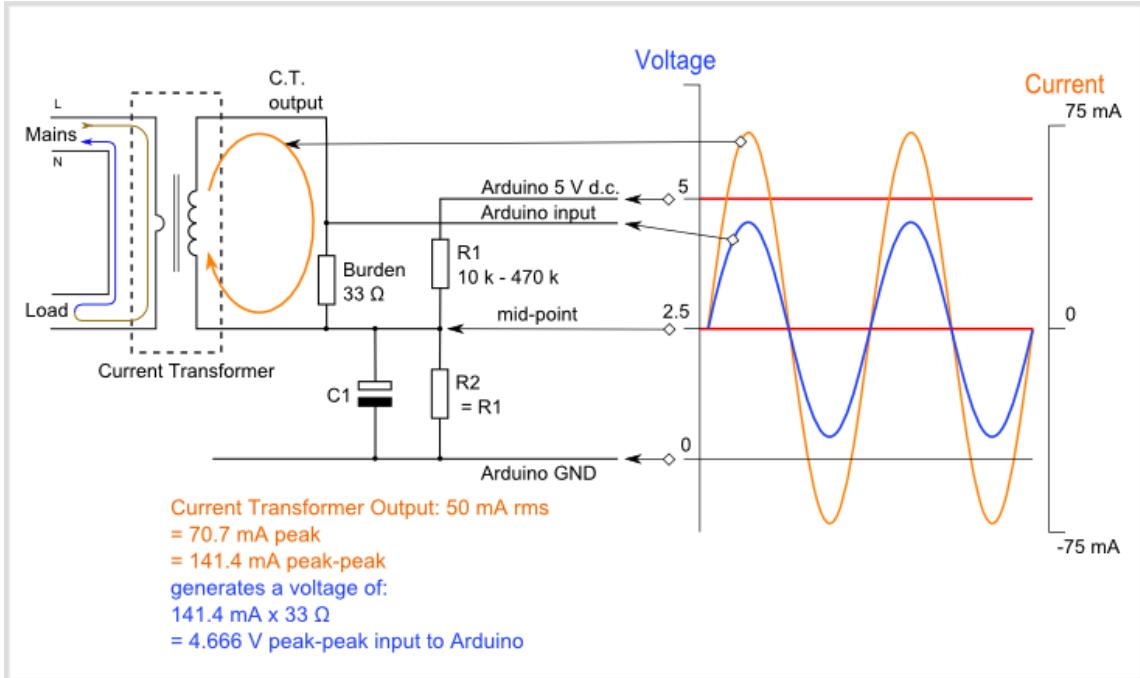


Рисунок 47 - Принцип работы и схема подключения датчика тока к Arduino

Стоимость датчика составляет \$11.

Программы считывания измерительных данных с датчиков приведены в приложении Б.

### 3.5 Выбор устройств управления для интеграции в систему

Выбор устройств управления зависит от структуры конкретной инженерной системы здания в силу их разнообразия, поэтому далее будет приведено несколько примеров конкретных устройств, на основе которых могут быть собраны остальные.

- устройство управления на основе твердотельного реле [13] (рисунок 48):

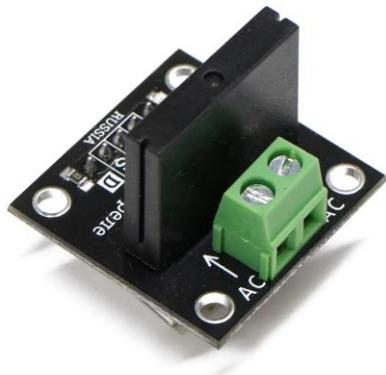


Рисунок 48 - Trema модуль, твердотельное реле

Trema модуль, твердотельное реле (*SSR - Solid-state relay*) - предназначено для замыкания выходной цепи переменного тока (с номинальным напряжением до 240 В), при подаче управляющего сигнала 5 В постоянного тока на вход модуля. Данный модуль можно использовать для управления приборами, подключаемыми к сети 220 В (осветительные, бытовые и т.д. с током потребления до 2 А).

Благодаря отсутствию механических движущихся частей, твердотельное реле имеет больший срок службы и не издаёт характерных звуков, по сравнению с обычными электромеханическими реле. Цифровая схема твердотельного реле связана с силовой частью, симметричным триодным фототиристором. В модуле используется реле с функцией пересечения ноля (*zero cross function*, рисунок 49). Данная функция предусматривает возможность замыкания выходной цепи, только когда переменное напряжение нагрузки пересекает ноль, благодаря чему, ток нагрузки увеличивается плавно. Такой тип включения продлевает срок службы устройств, подключённых к силовой части модуля, но делает невозможным замыкание цепей нагрузки постоянного тока.

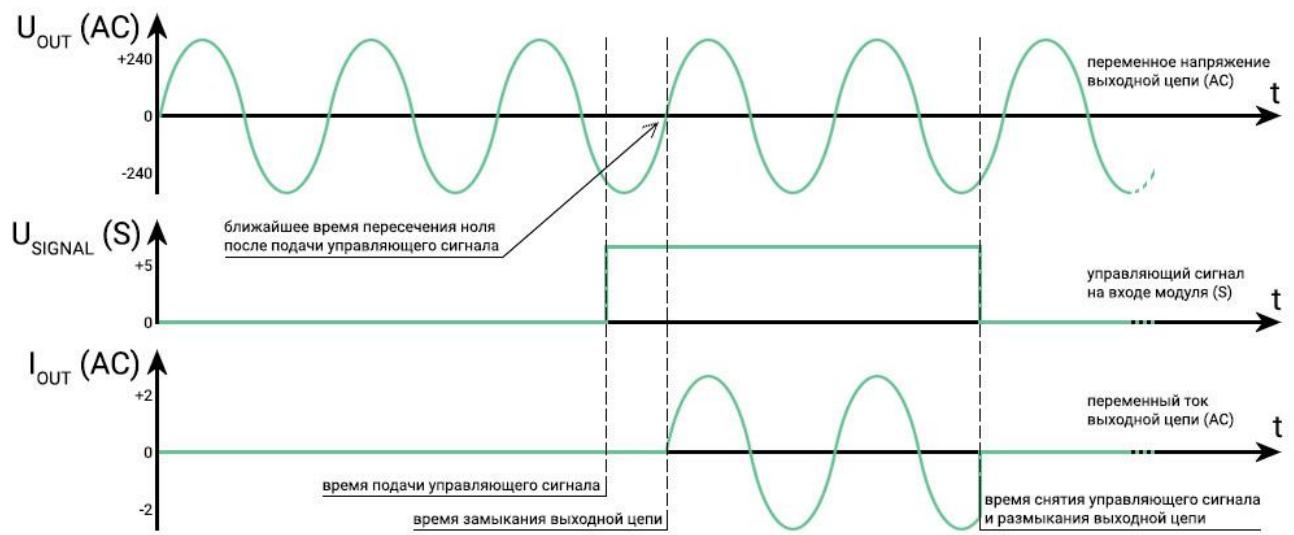


Рисунок 49 - Функция пересечения нуля (zero cross function)

Из графика видно, что после подачи управляющего сигнала на вход модуля, реле ожидает, когда приложенное к выходной цепи напряжение «пересечёт» ноль и только в этот момент замыкает выходную цепь. Следовательно, в выходной цепи не происходит резкого увеличения тока.

Характеристики модуля:

- напряжение срабатывания на входе (S): >4 В (номинально 5 В);
- максимальное напряжение на входе (S): <6 В;
- максимальный ток, на входе (S) модуля: <12 мА, при  $V_{CC} = 5$  В;
- номинальное напряжение нагрузки (AC): 120 ... 240 В, переменного тока 50/60 Гц;
- пиковые значения напряжения нагрузки (AC): 75 ... 264 В, переменного тока 50/60 Гц;
- падение напряжения в цепи нагрузки (AC): <1.6 В;
- номинальный ток нагрузки (AC): 0,1 ... 2 А;
- пиковое значение тока нагрузки (AC): 30 А (не повторяющееся и не дольше 20 мс);
- ток утечки в цепи нагрузки (AC): 1 мА при напряжении 100 В и 1,5 мА при напряжении 200 В;
- максимальное время срабатывания: <1 мс после пересечения нуля;

- рабочая температура: -30 ... 80 °C, при относительной влажности воздуха <85 %;
- температура хранения: -30 ... 100 °C;
- габариты: 30 × 30 × 30 мм. (с учётом колодки выводов);
- вес: 9 г.

Стоимость модуля составляет \$4.

Trema-модуль подключается к любому из выводов Arduino. Выход (AC) модуля (клеммник) подключается в разрыв фазы силовой цепи переменного тока.

Программа управления Trema-модулем находится в приложении Б.

Схема подключения изображена на рисунке 50:

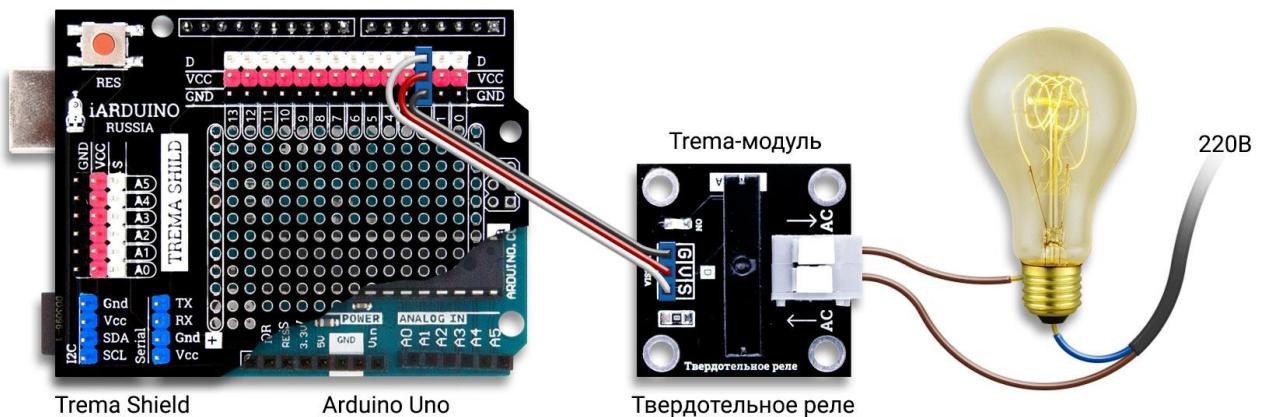


Рисунок 50 - Схема подключения устройства управления на основе твердотельного реле к Arduino

- устройство управления на основе драйвера управления двигателем L398N [14] (рисунок 51):



Рисунок 51 - Модуль управления двигателем

Модуль драйвера управления двигателем позволяет управлять одним или двумя моторами постоянного тока, либо шаговым двигателем с потребляемым током до 2 А.

Стоимость модуля составляет \$2.

Схема подключения приведена на рисунке 52:

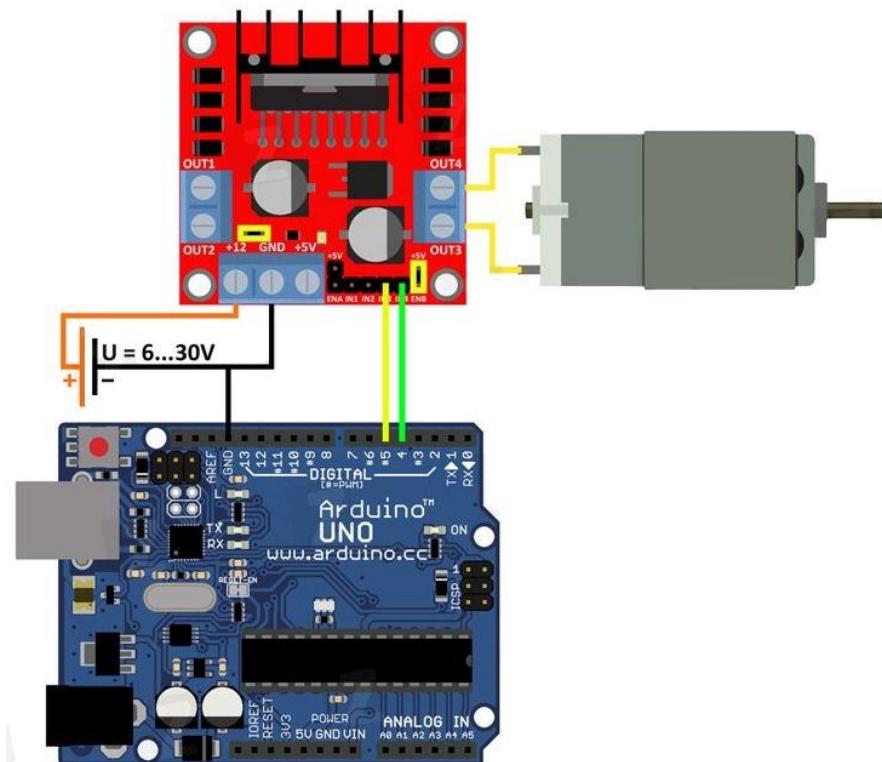


Рисунок 52 - Схема подключения драйвера управления двигателем

Программа управления двигателем приведена в приложении Б.

Для наглядности использования устройств управления можно привести в пример схему подключения Arduino к датчику контроля влаги и модулю управления двигателем (рисунок 53) для реализации функции контроля протечек:

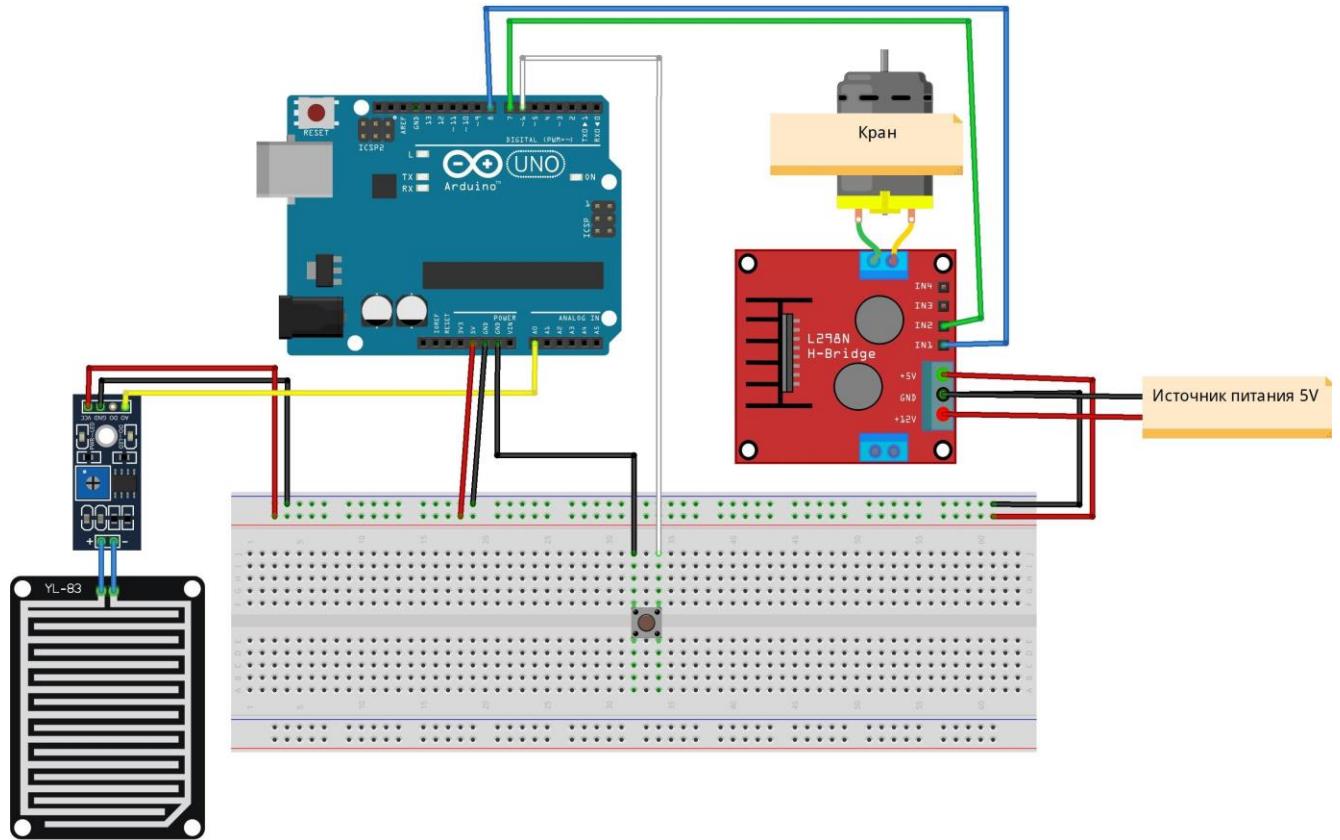


Рисунок 53 - Схема реализации функции контроля протечек

Программа реализации функции контроля протечек находится в приложении Б.

### 3.7 Расчет итоговой стоимости системы и сравнение с существующими системами на рынке

Далее будет составлен список всех компонентов, исходящих из выбранных функций со стоимостью каждого и рассчитана итоговая стоимость системы.

- Функция контроля пожарной опасности (список компонентов в таблице 2).

Таблица 2 - Перечень компонентов для функции контроля пожарной опасности

Наименование компонента	Количество	Стоимость
Датчик газа и задымления MQ-9	3 шт.	\$4,5
Датчик открытого огня KY-026	3 шт.	\$4,5

- Функция контроля проникновения в дом (список компонентов в таблице 3).

Таблица 3 - Перечень компонентов для функции контроля проникновения в дом

Наименование компонента	Количество	Стоимость
Датчик движения HC-SR501	3 шт.	\$4,5
Датчик разбития стекла	2 шт.	\$20
Датчик открытия двери	2 шт.	\$2
Трета-модуль для управления освещением	2 шт.	\$8
Трета-модуль для управления устройством имитации хозяев	1 шт.	\$4

- Функция контроля водоснабжения (список компонентов в таблице 4).

Таблица 4 - Перечень компонентов для функции контроля водоснабжения

Наименование компонента	Количество	Стоимость
Датчик протечки	2 шт.	\$2
Модуль драйвера на L398N для управления устройством перекрытия водоснабжения	2 шт.	\$4

- Функция контроля короткого замыкания электросети (список компонентов в таблице 5).

Таблица 5 - Перечень компонентов для функции контроля короткого замыкания электросети

Наименование компонента	Количество	Стоимость
Датчик тока	1 шт.	\$11
Трета-модуль для управления устройством обесточивания здания	1 шт.	\$4

- Функция управления освещением (список компонентов в таблице 6).

Таблица 6 - Перечень компонентов для функции управления освещением

Наименование компонента	Количество	Стоимость
Датчик движения HC-SR501	2 шт.	\$3
Датчик уровня освещенности CY-30	2 шт.	\$3
Трета-модуль для управления освещением	2 шт.	\$8
Модуль драйвера на L398N для устройства управления жалюзи	2 шт.	\$4

- Функция управления кондиционированием (список компонентов в таблице 7).

Таблица 7 - Перечень компонентов для функции управления кондиционированием

Наименование компонента	Количество	Стоимость
Датчик влажности и температуры DHT22	2 шт.	\$8
Трета-модуль для устройства управления кондиционированием	1 шт.	\$4

- Функция управления отоплением (список компонентов в таблице 8).

Таблица 8 - Перечень компонентов для функции управления отоплением

Наименование компонента	Количество	Стоимость
Датчик, интегрирующийся в систему отопления (погружной)	1 шт.	Зависит от типа датчика, но в среднем \$15
Trema-модуль для управления отоплением	1 шт.	\$4

В результате итоговая стоимость всех подключаемых к платформе устройств занимает \$120.

Далее нужно свести все устройства системы в один список (таблица 9) и посчитать общую стоимость затрат на систему.

Таблица 9 - Общий список компонентов для построения системы

Наименование компонента	Стоимость
Все подключаемые к платформе устройства	\$120
Компоненты для системы автономного питания	\$19
Платформа Arduino Mega	\$8
Wi-Fi модуль NodeMCU ESP12E	\$6
Преобразователь логических уровней	\$3
Предполагаемые затраты на сервис Blynk	\$4
Прочие компоненты (в т.ч. для монтажа проводов и сами провода)	~\$50

В итоге общая стоимость затрат на построение системы «Умный дом» составляет около \$210, что ниже средней стоимости существующих систем в \$400, а с учетом высокой универсальности и возможностями расширения имеет перед ними преимущества.

## ЗАКЛЮЧЕНИЕ

В ходе выполнения работы можно сделать вывод, что рынок существующих систем находится все еще на стадии формирования. Количество универсальных систем с возможностями, покрывающими все необходимые потребности человека, оставляет желать лучшего, вследствие чего количество подключенных «Умных домов» в России крайне низкое. Но благодаря высокой универсальности платформы Arduino вполне возможно собрать недорогую и эффективную систему управления умным домом.

Выполненные в работе задачи:

- проведен анализ и сравнение существующих систем;
- проведен анализ типовой структуры и определены необходимые функции системы, в которых нуждается потребитель;
- произведен выбор оптимального типа аппаратной платформы Arduino;
- произведен выбор структуры сети для данной платформы;
- выбран беспроводной модуль для обеспечения удаленного управления системой;
- выбран оптимальный облачный сервис удаленного управления;
- произведен подбор необходимых компонентов, которые обеспечат работу выбранных функций системы;
- разработана функциональная схема системы управления умным домом;
- разработаны принципиальные схемы;
- разработаны программы считывания измерений с датчиков, управления двигателем, управления твердотельным реле;
- проведен расчет стоимости компонентов разработанной системы умного дома, подтвердивший экономическую эффективность данного решения.

## СПИСОК ЛИТЕРАТУРЫ

1. Маркетинговое исследование рынка умных домов в цифровой экономике. А.Е. Китаев, И.И. Миронова:  
<http://injoit.org/index.php/j1/article/download/486/469>
2. Умные дома в мировых и российских реалиях. Статистика и реальные проекты:  
<https://robo-hunter.com/news/umnie-doma-v-mirovih-i-rossiiskih-realiyah-statistika-i-realnie-proekti9028>
3. Test: Smart Home Kits Leave the Door Wide Open – for Everyone. 29th April 2014:  
<https://www.av-test.org/en/news/news-single-view/test-smart-home-kits-leave-the-door-wide-open-for-everyone/>
4. Практическое программирование МК Atmel AVR языке ассемблера - Ревич Ю. (2014)
5. Atmel 8-bit AVR Microcontroller ATmega328P Datasheet
6. Феномен Blynk и его альтернативы:  
<https://blog.kvv213.com/2017/02/fenomen-blynk-i-ego-a>
7. Проекты с использованием контроллера Arduino, Электроника - Петин В.А. (2014)
8. Создание программно-аппаратного комплекса «Управление освещением в умном доме» на микроконтроллерах Arduino
9. Многоядерные процессы, параллельное программирование, ПЛИС, системы обработки сигналов. Сборник статей - (2014)
10. ESP8266 Datasheet EN v4.3
11. Arduino, датчики и сети для связи устройств. 2-е изд. - Иго Том (2015)
12. Как подключать датчики на большие расстояния. Аппаратная платформа Arduino:

<http://arduino.ru/forum/obshchii/kak-podklyuchat-datchiki-na-bolshie-rasstoyaniya>

13. Твердотельное реле (Trema-модуль) - Описания, примеры, подключение к Arduino:

<https://wiki.iarduino.ru/page/tverdotelnoe-rele-trema-modul/>

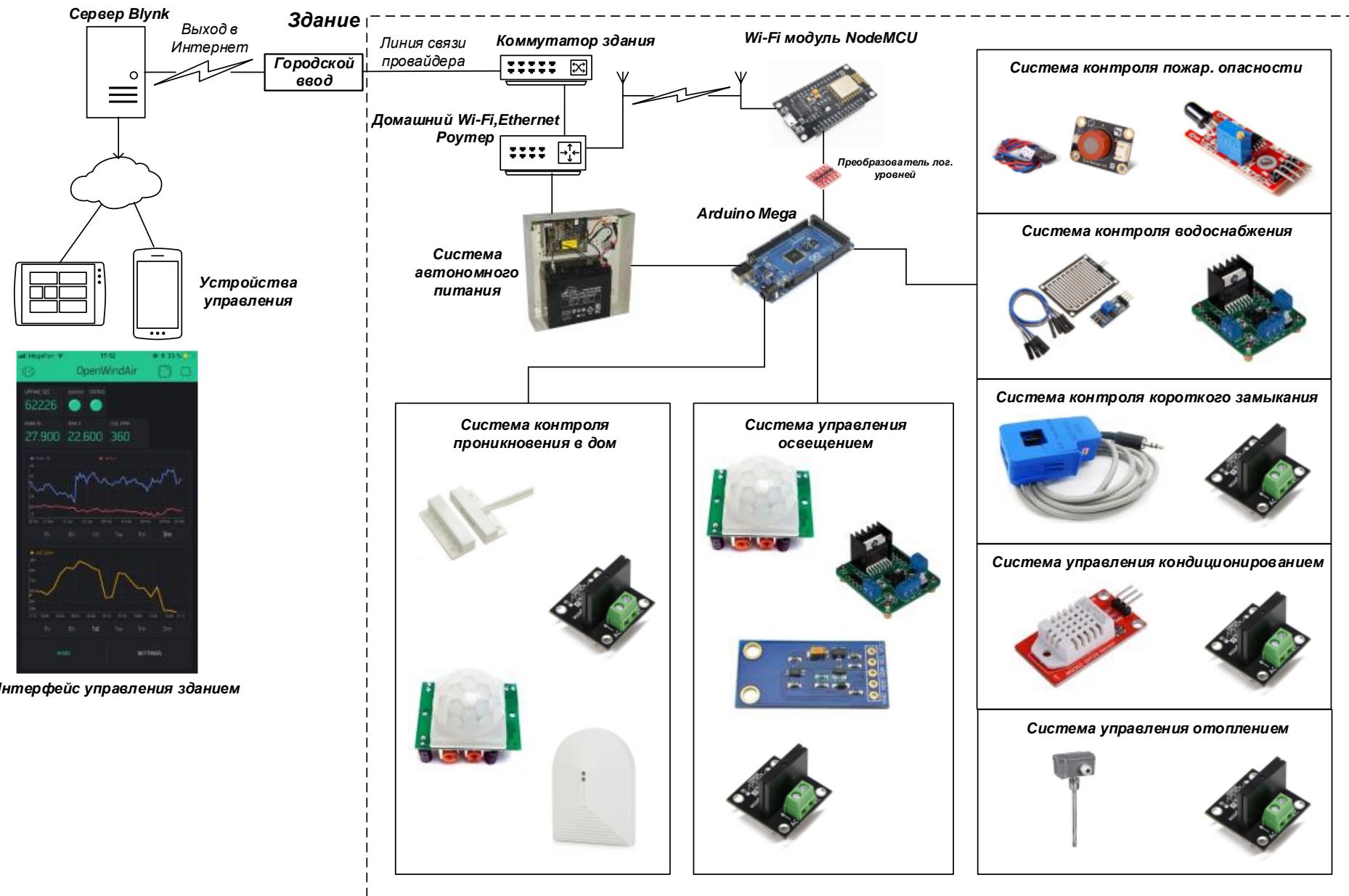
14. Управление Реле по MQTT на Arduino

<http://arduino.ru/forum/programmirovanie/upravlenie-rele-po-mqtt-na-arduino>

15. blynk-server\_README.md at master · blynkkk\_blynk-server · GitHub

<https://github.com/blynkkk/blynk-server/blob/master/README.md>

## Структурная схема системы «Умный дом» для интеграции в здание



## Приложение Б

### Примеры программ управления системой "Умный дом"

#### 1. Управление через сервис Blynk посредством Wi-Fi модуля NodeMCU

```
#define BLYNK_PRINT Serial

#include <ESP8266_Lib.h>
#include <BlynkSimpleShieldEsp8266.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "YourAuthToken";

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";

// Hardware Serial on Mega, Leonardo, Micro...
#define EspSerial Serial1

// or Software Serial on Uno, Nano...
//#include <SoftwareSerial.h>
//SoftwareSerial EspSerial(2, 3); // RX, TX

// Your ESP8266 baud rate:
#define ESP8266_BAUD 115200

ESP8266 wifi(&EspSerial);

void setup()
{
    // Debug console
    Serial.begin(9600);

    delay(10);

    // Set ESP8266 baud rate
    EspSerial.begin(ESP8266_BAUD);
    delay(10);

    Blynk.begin(auth, wifi, ssid, pass); }
```

## Продолжение приложения Б

```
void loop()
{
    Blynk.run();
}
```

### 2. Управление твердотельным реле Trema-модуль:

```
void setup()
{
    pinMode(2, OUTPUT);
}

void loop()
{
    digitalWrite(2, HIGH); delay(2000); // Подаем логическую «1» на вход модуля (включаем лампочку 200 В)
    digitalWrite(2, LOW ); delay(2000); // Снимаем логическую «1» со входа модуля (выключаем лампочку 200 В)
}
```

### 3. Управление двигателем

```
const int pwm = 2; // вывод 2 настраивается, как ШИМ
// выводы для выбора направления вращения двигателя
const int in_1 = 8;
const int in_2 = 9;

void setup()
{
    pinMode(pwm,OUTPUT);           // Мы должны настроить ШИМ вывод на выход
    pinMode(in_1,OUTPUT);          // Логические выводы так же настраиваются на выход
    pinMode(in_2,OUTPUT);
}

void loop()
{
    // Для вращения по часовой стрелке необходимо установить in_1 = High , in_2 = Low
    digitalWrite(in_1,HIGH);
    digitalWrite(in_2,LOW);

    analogWrite(pwm,255); // установка pwm в значение 255
```

## Продолжение приложения Б

```
// Мы можем изменить скорость вращения, изменив выходное значение pwm.  
// Вращение двигателя по часовой стрелке 3 секунды.  
delay(3000);  
  
// Остановка  
digitalWrite(in_1,HIGH);  
digitalWrite(in_2,HIGH);  
delay(1000) ;  
  
// Для вращения против часовой стрелки необходимо установить IN_1 = LOW , IN_2 =  
HIGH  
digitalWrite(in_1,LOW);  
digitalWrite(in_2,HIGH);  
delay(3000);  
  
//Остановка  
digitalWrite(in_1,HIGH);  
digitalWrite(in_2,HIGH);  
delay(1000);  
}
```

### 3. Реализация функции контроля протечек

```
bool _bounseInputD6S = 0;  
bool _bounseInputD6O = 0;  
unsigned long _bounseInputD6P = 0UL;  
  
void setup()  
{  
pinMode(6, INPUT);  
digitalWrite(6, HIGH);  
pinMode(7, OUTPUT);  
pinMode(8, OUTPUT);  
  
_bounseInputD6O = digitalRead(6);  
}  
void loop()  
{  
bool _bounceInputTmpD6 = (digitalRead (6));  
if (_bounseInputD6S)  
{  
if (millis() >= (_bounseInputD6P + 40))
```

## Продолжение приложения Б

```
    {_bounceInputD6O= _bounceInputTmpD6; _bounceInputD6S=0;}  
}  
else  
{  
    if (_bounceInputTmpD6 != _bounceInputD6O )  
        {_bounceInputD6S=1; _bounceInputD6P = millis();}  
}  
  
//Плата:1  
digitalWrite(7, !(_bounceInputD6O));  
digitalWrite(8, ( (analogRead (0)) <= (350));  
  
}
```

### 4. Считывание измерительных данных с датчика газа и задымления MQ-9

```
// библиотека для работы с датчиками MQ (Troyka-модуль)  
#include <TroykaMQ.h>  
  
// имя для пина, к которому подключен датчик  
#define PIN_MQ9      A0  
// имя для пина, к которому подключен нагреватель датчика  
#define PIN_MQ9_HEATER 13  
  
// создаём объект для работы с датчиком  
// и передаём ему номер пина выходного сигнала и нагревателя  
MQ9 mq9(PIN_MQ9, PIN_MQ9_HEATER);  
  
void setup()  
{  
    // открываем последовательный порт  
    Serial.begin(9600);  
    // запускаем термоцикл  
    // в течении 60 секунд на нагревательный элемент подаётся 5 вольт  
    // в течении 90 секунд — 1,5 вольта  
    mq9.cycleHeat();  
}  
  
void loop()  
{  
    // если прошёл интервал нагрева датчика  
    // и калибровка не была совершена
```

## Продолжение приложения Б

```
if (!mq9.isCalibrated() && mq9.atHeatCycleEnd()) {  
    // выполняем калибровку датчика на чистом воздухе  
    mq9.calibrate();  
    // выводим сопротивление датчика в чистом воздухе (Ro) в serial-порт  
    Serial.print("Ro = ");  
    Serial.println(mq9.getRo());  
    // запускаем термоцикл  
    mq9.cycleHeat();  
}  
// если прошёл интевал нагрева датчика  
// и калибровка была совершена  
if (mq9.isCalibrated() && mq9.atHeatCycleEnd()) {  
    // выводим отношения текущего сопротивление датчика  
    // к сопротивлению датчика в чистом воздухе (Rs/Ro)  
    Serial.print("Ratio: ");  
    Serial.print(mq9.readRatio());  
    // выводим значения газов в ppm  
    Serial.print(" LPG: ");  
    Serial.print(mq9.readLPG());  
    Serial.print(" ppm ");  
    Serial.print(" Methane: ");  
    Serial.print(mq9.readMethane());  
    Serial.print(" ppm ");  
    Serial.print(" CarbonMonoxide: ");  
    Serial.print(mq9.readCarbonMonoxide());  
    Serial.println(" ppm ");  
    delay(100);  
    // запускаем термоцикл  
    mq9.cycleHeat();  
}  
}
```

### 5. Считывание измерительных данных с датчика открытого огня KY-026

```
int led = 13 ;// назначение пина встроенного светодиода  
int flameDigitalPin = 10; // назначение пина цифрового входа датчика  
int flameAnalogPin = A0; // назначение пина аналогового входа датчика  
  
int valueDigital ;// переменная для цифрового значения  
float valueAnalog; //переменная для аналогового значения
```

## Продолжение приложения Б

```
void setup ()  
{  
    pinMode (led, OUTPUT) ;  
    pinMode (flameDigitalPin, INPUT) ;  
    pinMode (flameAnalogPin, INPUT) ;  
    Serial.begin(9600);  
}  
  
void loop ()  
{  
    valueAnalog = analogRead(flameAnalogPin);  
    Serial.println(valueAnalog); // вывод аналогового значения в монитор порта  
  
    valueDigital = digitalRead (flameDigitalPin) ;// чтение цифрового значения  
    if (valueDigital == HIGH) // когда на цифровом входе высокий уровень светодиод горит  
    {  
        digitalWrite (led, HIGH);  
    }  
    else  
    {  
        digitalWrite (led, LOW);  
    }  
    delay(1000);  
}
```

### 6. Считывание измерительных данных с датчика движения HC-SR501

```
void setup() {  
//Установить соединение с монитором порта  
    Serial.begin(9600);  
}  
  
void loop() {  
//Считываем пороговое значение с порта A0  
    if(analogRead(A0) > 500) {  
//Сигнал с датчика движения  
        Serial.println("Есть движение!"); }  
    else {  
//Нет сигнала  
        Serial.println("Всё тихо...");  
    }  
}
```

## Продолжение приложения Б

### 7. Считывание измерительных данных с датчика открытия двери

```
int Door_Led_Pin = 13; // выбрать контакт для светодиода
int Door_Sensor_Pin = 6; // контакт для датчика
int val = 0; // переменная для хранения состояния датчика

void setup() {
    pinMode(Door_Led_Pin, OUTPUT); // установить Door_Led_Pin как выход
    pinMode(Door_Sensor_Pin, INPUT); // установить Door_Sensor_Pin как вход
}

void loop() {
    val = digitalRead(Door_Sensor_Pin); // читать Door_Sensor_Pin
    if (val == HIGH) { // Если Door_Sensor N.C. (без магнита) -> HIGH : Дверь открыта / LOW : Дверь закрыта
        // Если Door_Sensor N.O. (nc с магнитом) -> HIGH : Дверь открыта / LOW : Дверь за
        // крыта
        digitalWrite(Door_Led_Pin, LOW); //выключить светодиод Door_Led
    } else {
        digitalWrite(Door_Led_Pin, HIGH); //включить светодиод Door_Led
    }
}
```

### 8. Считывание измерительных данных с датчика уровня освещенности CY-30

```
#include <Wire.h>
#include <BH1750.h>
BH1750 lightMeter;

void setup(){
    Serial.begin(9600);
    lightMeter.begin();
    Serial.println("Running...");
}

void loop() {
    uint16_t lux = lightMeter.readLightLevel();
    Serial.print("Light: ");
    Serial.print(lux);
    Serial.println(" lx");
    delay(500); }
```

## Продолжение приложения Б

### 9. Считывание измерительных данных с датчика влажности и температуры DHT22

```
#include "DHT.h"                                // Подключаем библиотеку DHT

#define DHTPIN 2                                 // Пин к которому подключен датчик
#define DHTTYPE DHT22                            // Используемый датчик DHT 22 (AM2302), AM2321
DHT dht(DHTPIN, DHTTYPE);                      // Инициализируем датчик

void setup()
{
    Serial.begin(9600);                         // Задаем скорость передачи данных
    Serial.println("DHT22 тест!");                // Выводим текст
    dht.begin();                               

}

void loop()
{
    delay(2000);                               // Задержка в 2 с
    float h = dht.readHumidity();               // Считывание влажности в процентах
    float t = dht.readTemperature();            // Считывание температуры в градусах цельсия

    if (isnan(h) || isnan(t))                  // Проверяем, получилось считать данные
    {
        Serial.println("Ошибка чтения DHT22"); // Выводим текст
        return;
    }

    float hic = dht.computeHeatIndex(t, h, false); // Расчет индекса влажности
    Serial.print("Влажность: ");                 // Выводим текст
    Serial.print(h);                           // Отображаем влажность
    Serial.print(" %\t");                      // Выводим текст
    Serial.print("Температура: ");              // Выводим текст
    Serial.print(t);                           // Отображаем температуру
    Serial.println(" *C ");                     // Выводим текст
}
```

# Оглавление

АННОТАЦИЯ .....	5
ВВЕДЕНИЕ.....	6
1    АКТУАЛЬНОСТЬ СИСТЕМЫ «УМНЫЙ ДОМ» В СОВРЕМЕННЫХ РЕАЛИЯХ .....	9
2    ВЫБОР СПОСОБА ПОСТРОЕНИЯ ИВК ДЛЯ СИСТЕМЫ УМНЫЙ ДОМ .....	15
2.1    Исследование концепции и структуры информационно-вычислительных комплексов в применении к системе «Умный дом» .....	15
2.2    Выделение основных функциональных особенностей системы «Умный дом» .....	18
3    РАЗРАБОТКА ИВК ДЛЯ УДАЛЕННОГО УПРАВЛЕНИЯ СИСТЕМОЙ «УМНЫЙ ДОМ» .....	24
3.1    Выбор платформы .....	24
3.2    Обзор всех возможностей платформы Arduino .....	29
3.2    Обеспечение удаленного управления с помощью Arduino .....	35
3.3    Разработка и выбор компонентов устройства автономного питания .....	40
3.4    Выбор измерительных датчиков для интеграции в систему .....	41
3.5    Выбор устройств управления для интеграции в систему .....	53
3.7    Расчет итоговой стоимости системы и сравнение с существующими системами на рынке .....	58
ЗАКЛЮЧЕНИЕ .....	62
СПИСОК ЛИТЕРАТУРЫ .....	63
ПРИЛОЖЕНИЕ А .....	65
ПРИЛОЖЕНИЕ Б .....	66