

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ ГАГАРИНА Ю.А.»

Институт прикладных информационных технологий и коммуникаций

Кафедра «Прикладные информационные технологии»

Направление 09.03.03 «Прикладная информатика»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
«Разработка программного обеспечения отслеживания приема
прописанных лекарств пациентом. Подсистема пациента»

Студент (ка) _____ Ормели Александр Федорович _____
фамилия, имя, отчество

курс 4 группа б2-ПИНФ41

Руководитель

доцент каф. ПИТ, к.т.н. _____ 11.06.20 А.В. Ермаков _____
должность, ученая степень, уч. звание подпись, дата Инициалы Фамилия

Допущен к защите

Протокол № 12 от «11» июня 2020 года

Зав. кафедрой «Прикладные информационные технологии»

кандидат технических наук, доцент _____ 11.06.20 О.А. Торопова _____
ученая степень, уч. звание подпись, дата Инициалы Фамилия

Саратов 2020г

Содержание

ВВЕДЕНИЕ	4
1 Анализ предметной области	6
1.1 Формирование требований к системе	6
1.2 Сравнительный анализ существующих решений	7
1.2.1 Система GlowCap.....	7
1.2.2 Система AdhereTech	9
1.2.3 Система MedReady 1700.....	11
1.2.4 Система MedCenter System.....	12
1.2.5 Система Xiaomi Zayata Portable Smart Pill Dispenser	14
1.3 Итоги.....	15
2 Разработка аппаратного обеспечения системы	16
2.1 Описание процессов в системе.....	16
2.1 Сравнение микроконтроллера и микрокомпьютера	20
2.1.1 Плата на микроконтроллере.....	20
2.1.2 Плата на микрокомпьютере	21
2.2 Сравнение технических характеристик плат на микроконтроллере ..	23
2.2.1 Плата Arduino Uno.....	23
2.2.2 Плата Arduino Leonardo.....	24
2.2.3 Плата Iskra Neo	26
2.2.4 Плата Arduino Mini.....	27
2.2.5 Плата Arduino Micro	28
2.2.6 Плата Arduino Mega.....	30
2.2.7 Плата Arduino Due	31
2.2.8 Плата Iskra JS	33
2.2.9 Плата Strela	34
2.2.10 Плата Arduino Yún.....	36
2.2.11 Плата Netduino 2	37
2.2.12 Плата Netduino Plus 2.....	39
2.3 Выбор способа передачи данных	40
2.3.1 Связь по Мобильной сети	41
2.3.2 Связь по сети Bluetooth	41

2.3.3 Связь по сети Интернет	42
2.4 Сравнение Ethernet-модулей для Arduino Uno	43
2.4.1 Плата расширения Ethernet Shield W5100	44
2.4.2 Модуль подключения ENC28J60	45
2.5 Выбор датчика открытия банки	46
2.6 Результат сравнительного анализа аппаратной части	48
2.7 Реализация аппаратной части.....	49
2.8 Процесс подключения и эксплуатация устройства	53
3 Разработка программного обеспечения серверной части системы	58
3.1 Язык программирования PHP	59
3.2 Серверная платформа	60
3.3 Детали реализации	61
ЗАКЛЮЧЕНИЕ	65
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	67

ВВЕДЕНИЕ

В настоящее время цифровые технологии внедрены во все сферы человеческой деятельности. Развитие медицины также невозможно без внедрения новых технологий по лечению, реанимированию, транспортировке больных. Одной из проблем, которые могут быть решены с использованием информационных технологий является контроль за приемом лекарств пациентами. Достаточно распространена ситуация, когда амбулаторный больной получает курс лечения, включающий лекарства, однако при этом он постоянно пропускает их прием. Одной из множества причин, которой люди объясняют несоблюдение режима приёма медикаментов, является забывчивость [1]. Забывчивость может возникать на подсознательном уровне из-за множества психологических факторов, таких как психологический защитный механизм отрицания. Для некоторых людей заболевание является постоянным источником беспокойства, а лекарственные средства - постоянным напоминанием об этом. Так же беспокойство может возникать из-за возможных побочных эффектов лекарственных средств [2].

Пропуск приема медикаментов у пациента может привести к серьезным последствиям. По данным статистики, каждый год 125000 человек умирают от сердечно-сосудистых заболеваний [3] из-за несоблюдения режима приема лекарств. Так же, если пациент не принимает все прописанные специалистом антибиотики, могут происходить повторные вспышки не долеченных инфекционных заболеваний и появляться проблемы, которые связаны с появлением бактерий, устойчивых к антибиотикам [4].

Целью работы является создание аппаратно-программное платформы автоматизации процесса контроля приема медикаментов пациентом,

находящимся на амбулаторном лечении, которая позволит, с одной стороны напоминать пациенту о необходимости приема лекарств в соответствии с расписанием, и собирать статистику о приеме лекарств для внешних систем с другой стороны.

Для достижения цели необходимо решить следующие задачи:

1. Провести анализ предметной области и сформулировать требования к решению.
2. Провести анализ существующих решений и определить их применимость для решения поставленных задач.
3. Спроектировать и разработать аппаратное обеспечение с возможностью отслеживания приема лекарств и доступом в сеть Интернет.
4. Разработать программное обеспечение для сбора сведений о приеме лекарств с аппаратного обеспечения с веб-интерфесом для просмотра статуса аппаратного обеспечения.

1 Анализ предметной области

1.1 Формирование требований к системе

Данная система разрабатывается для использования как врачом, так и пациентом. Для врача важно, чтобы система хранила информацию о режиме приема медикаментов, а также позволяла его настроить и предоставляла доступ ко времени и дате приема этих медикаментов. Для пациента нужно, чтобы такая система записывала данные о его приеме медикаментов и оповещала его о необходимости приема медикаментов. Таким образом в системе должно быть устройство, которое будет отслеживать открытие хранилища лекарств и сигнализировать о необходимости приема медикаментов, а также программное обеспечение, которое вело бы запись приема медикаментов, а также хранило и позволяло настраивать режим прием медикаментов

Важным критерием такого устройства является минимизация его стоимости, особенно при планировании серийного производства и возможность выхода в Интернет, поскольку данное устройство должно обладать возможностью обмена данными с сервером для отслеживания врачом прохождения курса лечения пациентом. Аппаратная часть должна иметь универсальные размеры для того, чтобы медикаменты любого размера могли поместиться в устройстве, кроме медикаментов жидкого типа.

Для программного обеспечения так же выдвигаются свои функциональные требования. Все данные о приеме должны записаться в файл для того, чтобы врач мог с легкостью отследить курс лечения, а также уточнить дату и время принятия медикамента. Специалист так же должен иметь возможность легко изменять график приема медикаментов.

Далее проанализируем существующие решения в этой области и попытаемся определить те, которые удовлетворяют заданным требованиям.

1.2 Сравнительный анализ существующих решений

На данный момент система здравоохранения в Соединенных Штатах Америки является ведущей в мире. По расходам США так же занимает первое место как в процентах ВВП, которое составляет 16%, так и в цифрах – 2,26 триллиона долларов, что составляет 7439 долларов на одного человека [24]. Также в США широко распространена цифровизация системы здравоохранения. Вводятся в эксплуатацию такие подсистемы как Electronic health record, что по сути является электронной медицинской картой пациента. Для того, чтобы такая система набирала популярность среди специалистов там используется система поощрений. Государство выплачивает 44 000 долларов тому, кто будет использовать такую систему, со временем сумма становится меньше. Эта подсистема входит в состав системы Медицинской Информационной Системы, которая в свою очередь используется для поддержки врачебных решений, инфообмен с другими медицинскими учреждениями, для того чтобы пациент мог продолжать лечение в любой доступной больнице для него и со страховыми компаниями, электронному хранению медицинских данных пациента, электронного рецепта а также получение лабораторных данных. Рассматриваемый нами тип устройств также представлен на рынке несколькими типами устройств.

1.2.1 Система GlowCap



Рисунок 1.1 - Система GlowCap

Компания NantHealth расширяет возможности создания интегрированных сетей оказания медицинской помощи. Разрабатываемые ими системы обеспечивают эффективность, персонализированное лечение и сотрудничество в сфере здравоохранения. NantHealth признана поставщиками программного обеспечения и высокотехнологичных устройств [6].

Устройство GlowCap представляет собой специализированную крышку, которая одевается на емкость с лекарствами. Ко времени приема медикаментов это устройство начинает мигать оранжевым цветом. Если пациент так и не принял лекарство после включения световой сигнализации, то через час устройство начинает проигрывать мелодию. Затем, по истечению двух часов с начала включения световой индикации на крышке, специальное устройство отправляет сообщение на телефон в случае пропуска приема. Так же для пациентов, проживающих в Соединенных Штатах Америки предусмотрена функция дозаказа медикаментов. Пациенту достаточно нажать на кнопку в основании крышки и через мобильную сеть оператора AT&T посылается автоматический сигнал в ближайшую аптеку с просьбой восполнения запасов медикаментов. В таком случае предусмотрен обязательный звонок владельцу устройства для подтверждения заказа. В случае если заказ был подтвержден, пациенту достаточно зайти в ближайшую аптеку и забрать свой заказ.

Система состоит из двух частей:

- Крышка которая отвечает за сбор информации о приеме медикаментов, световой и звуковой сигнализации о необходимости приёма медикаментов.

- Ночник, который дублирует функции крышки, а так же обеспечивает обмен данными крышки и приложения и заказ медикаментов в аптеке.

Возможности системы:

- Отслеживание приема медикаментов.
- Звуковая и световая сигнализация о необходимости приема медикаментов.
- Отправка сообщений о необходимости приема медикаментов на смартфон если пациент не произвел прием после световой и звуковой сигнализации.
- Заказ в ближайшей аптеке.

Достоинства системы:

- Разнообразие методов оповещения.
- Функция дозаказа медикаментов.

Отметим, что данное устройство не удовлетворяет требованиям выдвинутым нами ранее, в частности :

- Система работает исключительно на территории Соединенных Штатов Америки.
- Высокая стоимость устройства – 100 \$.
- Отсутствие возможности сбора данных для лечащего врача.

1.2.2 Система AdhereTech



Рисунок 1.2 - Система AdhereTech

Компания AdhereTech специализируется на медицинском оборудовании. Разрабатываемые системы соответствуют нормативам и обеспечивают безопасность пациентов, так как соответствуют стандартам FDA, HIPAA, CE, USP, ISO, FCC и другие [7].

Устройство AdhereTech выполняет функцию уведомления пациента о необходимости приема медикаментов, а также отслеживает регулярность потребления и дозировки. Данное устройство подключается к облачному сервису и обменивается информацией для сбора данных о приеме. Уведомление происходит путем звонка или текстового сообщения на телефон и миганием светового индикатора на корпусе баночки. Благодаря чипу беспроводной связи CDMA баночка в состоянии отправлять небольшое количество данных о дате и времени открытия, а также о количестве принятых таблеток или жидкого лекарства. Настройка курса лечения происходит заранее и помещается в специальное облако. При выдаче баночки все настройки по определенному курсу загружаются в память устройства. Срок службы такой баночки составляет 45 дней после чего баночка разряжается и выдается новая с обновленным запасом медикаментов.

Достоинства системы:

- Возможность отслеживать дату и время приема.
- Возможность отслеживать дозировку медикаментов.
- Предварительная настройка устройства.
- Низкая стоимость – 60\$.

Однако, система также не удовлетворяет сформулированным нами требованиям, в частности:

- Не соответствует стандартам, действующие на территории Российской Федерации.

- Распространяется только в медицинских учреждениях Соединенных Штатов Америки. Эти устройства используются только в больницах, которые заключили контракт с компанией Adheretech и находятся в Соединенных Штатах Америки.

1.2.3 Система MedReady 1700



Рисунок 1.3 - Система MedReady 1700

С 2010 года Assisted Living Technologies предоставляет технологические решения, которые способствуют независимости, безопасности и улучшению качества жизни пожилых людей и людей с особыми потребностями. Ассортимент решений включает в себя продукты для предотвращения падений, аварийного реагирования, дистанционного мониторинга, управления лекарственными препаратами и домашней безопасности [8].

Устройство MedReady 1700 представляет собой дозатор или систему распределения таблеток, которое состоит из 28 ячеек, и предназначенное для недельного приема медикаментов 3 раза в день. В каждый лоток заранее помещаются медикаменты лечащим врачом, которые пациент обязан

принять, что гарантирует правильную дозировку лекарств. В назначенное время лоток внутри проворачивается и нужная ячейка оказывается под небольшой крышкой, а также раздается световой и звуковой сигнал. Система оснащена модулем мобильной связи, который позволяет связаться с центральным сервером, содержащим информацию о правилах приема медикаментов и их исполнении, а также информацию о контактном лице, которым является врач или ухаживающий за пациентом медицинский персонал или родственник, чтобы отправить сообщение о том, что прием медикаментов был пропущен.

Достоинства системы:

- Гарантированная правильная дозировка медикаментов.
- Набор ячеек, рассчитанный на 7 дней.
- Звуковая и световая сигнализация.

Данная система не отвечает нашим требованиям в следующей части:

- Не соответствует стандартам, действующие на территории Российской Федерации.
- Отсутствует связь с лечащим врачом.

1.2.4 Система MedCenter System



Рисунок 1.4 - Система MedCenter System

MedCenter Systems была создана в 2007 году и представляет собой совокупность специальных контейнеров для медикаментов и главное устройство для отслеживания приема медикаментов. Устройство состоит из 31 коробочки, разделенных на 4 ячейки для утреннего, дневного, вечернего и ночного приема. На передней части располагается блок с датой и временем, а также с набором управляющих кнопок.

Устройство работает следующим образом: в начале месяца лечащий врач заполняет ячейки всех контейнеров системы и помещает их на свои места зеленой стороной вверх. В назначенный день утром пациент слышит звуковое сообщение о номере контейнера, которое следует поместить в отсек для сегодняшних медикаментов. После этого пациент обязан нажать на соответствующую кнопку, которая отвечает за подтверждение выполнения действия. В назначенное время в течении суток пациент будет получать звуковые сообщения о необходимости приема медикаментов из определенной ячейки и после каждого приема должен обязательно подтверждать прием путем нажатия на специальную кнопку. После последнего суточного приема пациента помещает контейнер на свое место красной стороной вверх, что сигнализирует о завершении приема этого суточного курса.

Достоинства системы:

- Система покрывает четырехразовый прием медикаментов в течении 31 дня.
- Звуковые сообщения о необходимости выполнения действий.

Данная система также не удовлетворяет требованиям, так как:

- Отсутствует связь с лечащим врачом.
- Отсутствует автоматическое подтверждение приема медикаментов.

1.2.5 Система Xiaomi Zayata Portable Smart Pill Dispenser



Рисунок 1.5 - Система Xiaomi Zayata Portable Smart Pill Dispenser

Устройство Xiaomi Zayata Portable Smart Pill Dispenser [10] представляет собой круговой контейнер с ячейками для медикаментов. Контейнер предназначен на семидневный курс. Устройство имеет возможность для установки шести будильников. Так же имеет возможность синхронизировать устройство с мобильным телефоном через канал связи Bluetooth. В приложении на телефоне пользователь будет иметь возможность выставлять будильники, следить за своей статистикой и настроить уведомления при срабатывании будильника.

Когда происходит срабатывание будильника светодиод на устройстве сигнализирует об этом и включается звуковое уведомление на устройстве, а также отправляет уведомление на телефон с помощью предварительно установленного приложения. Сигнализирование о необходимости приема медикаментов длится в течении 5 минут и по истечению этого времени записывает пропуск, в случае если пациент так и не принял медикаменты. В случае если пациент собирается принять медикаменты, то ему необходимо

расположить специальную крышку над нужной ячейкой и открыть ее после чего перевернуть устройство для извлечения медикамента и закрыть крышку. В момент переворота устройства система записывает принятие медикамента и отправляет соответствующие данные на смартфон.

Достоинства системы:

- Очень низкая стоимость – 14\$.
- Устройство изготовлено из антиаллергенного пластика.

К сожалению, данное устройство также не удовлетворяет нашим требованиям, так как:

- Врач или ухаживающее лицо не имеет доступа к информации о приеме.

1.3 Итоги

Подводя итоги можно сказать, что некоторые системы подходят исключительно для контейнеров с медикаментами определенных типов, то есть распространенные только в определенной стране. Так же не все системы поддерживают связь с врачом или пользователей и представляют собой обособленную систему способную лишь сигнализировать пользователю о необходимости принять медикаменты, но никак не отслеживают факт принятия медикамента пациентом. Существуют системы, которые связаны по смартфоном по каналу Bluetooth что позволяет синхронизировать систему с пользователем и на небольшом расстоянии и невозможность синхронизировать такую систему с врачом.

Проведенный обзор показывает, что в сложившихся условиях необходимо провести разработку собственной системы, которая удовлетворяла бы сформулированным требованиям.

2 Разработка аппаратного обеспечения системы

Для обработки, сбора и передачи информации в проектах умного города или умного дома используются микроконтроллеры или микрокомпьютеры и наборы датчиков [11]. В данной главе мы опишем реализацию аппаратной части нашего устройства, которое в дальнейшем мы будем называть “умная банка”.

2.1 Описание процессов в системе

Прежде всего, опишем процессы системы и определим необходимые её компоненты. Для описания будут использоваться нотации методологии структурного анализа SADT (Structured Analysis and Design Technique). Для построения наиболее общей схемы воспользуемся уровнем А0 методологии IDEF0 [25], позволяющей отобразить функции устройства. На рисунке 2.1. показана общая постановка задач для проектирования и разработки серверной части информационной системы “Умная банка”.

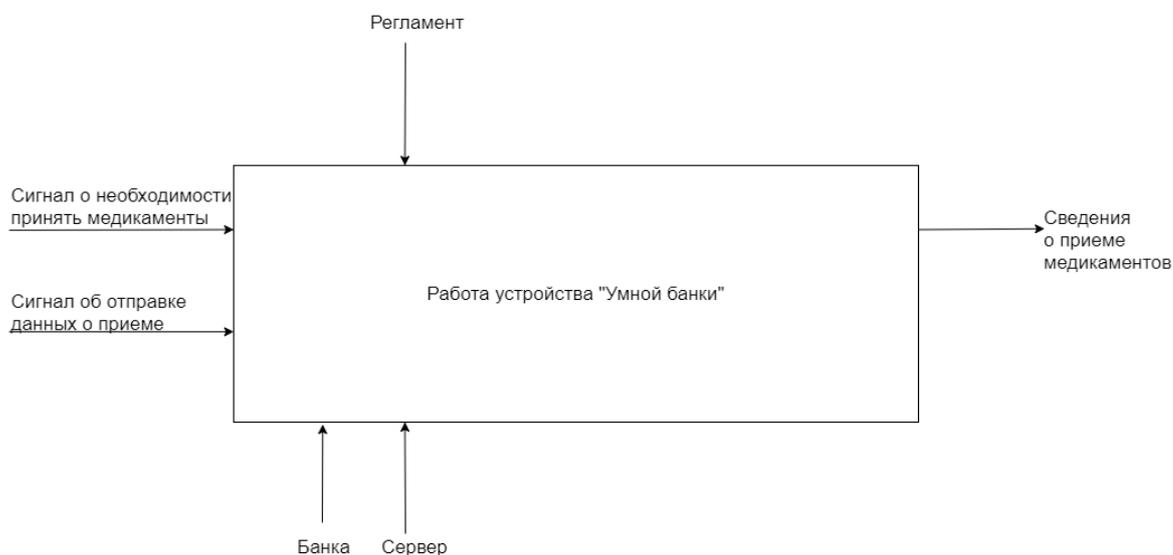


Рисунок 2.1 - Диаграмма уровня А0 для работы устройства «Умная банка»

На диаграмме показана общая постановка задачи: «Работа «Умной банки»». Диаграмма типа IDEF0 состоит из четырех типов стрелок: вход – это объекты, которые используются и преобразуются в процессе работы;

управление – информация, которая используется для управления действиями работы; выход – это объекты, в которые преобразуются входы и механизм – это ресурсы, которые выполняют работу. Для работы устройства на него должны поступить входящие данные, которыми являются сигнал о необходимости принять медикаменты, а также сигнал об отправке данных о приеме. В данном случае механизмом управления является регламент. Именно регламент управляет действиями работы устройства. Выполнение функций, которые необходимы для умной банки будут выполняться как непосредственно устройством, так и сервером. Выходными данными с умной банки являются сведения о приеме медикаментов.

Для более полного понимания работы устройства воспользуемся диаграммой уровня A1 модели IDEF0. На рисунке 2.2 представлена контекстная диаграмма IDEF0 модели [5].



Рисунок 2.2 - Диаграмма уровня A1 для работы устройства «Умная банка»

На диаграмме уровня A1 расписаны процессы, происходящие во время работы устройства. Работа устройства начинается после того, как сервер отправляет сигнал о необходимости принять медикаменты на устройство. Приняв эту информацию, устройство ее обрабатывает и включает на своей стороне световую сигнализацию о необходимости

приема медикаментов из банки. Сразу после включения световой индикации устройство запускает процесс сбора информации с датчика открытия/закрытия. Далее приходит управляющий сигнал об отправке данных о приеме с сервера, после чего устройство собирает полученную с датчика информацию и выдает управляющий сигнал на отключение световой сигнализации о приеме медикаментов. Далее устройство отключает световую индикацию на банке и формирует ответ из информации с датчика в котором содержится информация о статусе приема. Заканчивается работа устройства отправкой сформированного ответа на сервер.

IDEF3 - методология моделирования и стандарт документирования процессов, происходящих в системе. IDEF3 широко применяется при разработке информационных систем. При этом используется инструмент визуального моделирования бизнес-процессов.

Для процесса работы «Умной банки», которая принимает управляющий сигнал, включает сигнализацию на выбранных банках, собирает и отправляют информацию о приеме медикаментов построена диаграмма семейства IDEF3 и представлена на рисунке 2.3.

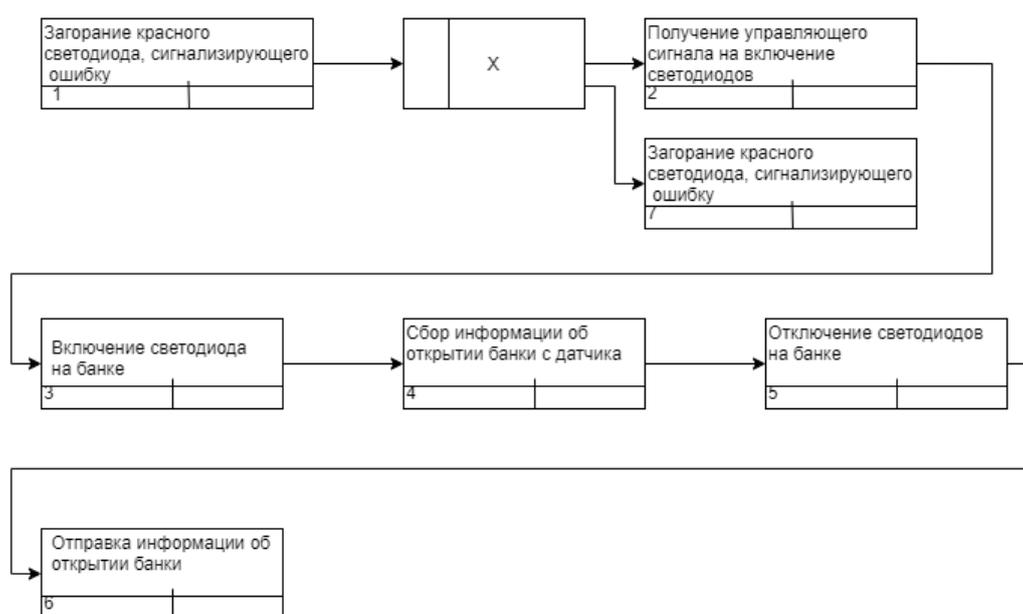


Рисунок 2.3 - Диаграмма IDEF3 для «Умной банки»

На диаграмме представлено описание технологических процессов аппаратной системы, с указанием того, что происходит на каждом этапе процесса. На диаграмме присутствуют разворачивающиеся потоки данных соединения, предназначенные для завершения одного действия и вызова начала выполнения других [15]. По данной диаграмме можно составить определенную последовательность действий устройства. В случае ошибки загорается красный светодиод. Далее, если ошибка не была устранена, то красный светодиод остается гореть. В случае, если ошибка была устранена, то идет получение управляющего сигнала на включение светодиодов, после чего идет включение светодиода на банке. После включения световой индикации начинается сбор данных об открытии банки. После окончания сбора данных с датчиков об открытии банки идет отключение светодиодов на устройстве и завершается отправкой информации об открытии банок.

На основе проведенного анализа работы умной банки можно сделать вывод что устройство должно состоять из нескольких компонентов:

- Платы для управления устройством, обработки сигналов с датчиков и формирования ответа.
- Устройства для соединения платы управления с сетью Ethernet для обмена данными с разрабатываемым устройством.
- Датчик для определения открытия контейнера с медикаментами, чтобы отправить эту информацию на сервер.
- Светодиод для световой сигнализации о необходимости приема медикаментов.

В следующих разделах мы осуществим выбор этих компонент и опишем формирование итогового устройства из них.

2.1 Сравнение микроконтроллера и микрокомпьютера

Для выбора подходящей платы проведем анализ существующих плат и выберем наиболее подходящую. На рынке представлены два типа плат для управления устройствами: микроконтроллеры и микрокомпьютеры. Для микроконтроллеров характерно выполнять только одну операцию одновременно, но с высокой производительностью, в то время как одноплатные компьютеры выполняют программы в рамках операционной системы, обладают обширными мультимедийными возможностями и большой производительностью. Чтобы выбрать наиболее подходящий контроллер, на основе которого будет основываться разрабатываемое устройство, нужно сравнить микроконтроллер и микрокомпьютер.

2.1.1 Плата на микроконтроллере

На основании среднестатистических данных можно выявить основные технические характеристики микроконтроллера.

- Производительность: 1 ядро, десятки-сотни МГц, десятки КБ оперативной памяти, десятки-сотни КБ постоянной памяти.
- Многозадачность: отсутствует, но есть возможность эмулировать.
- Простота подключения к сети интернет: Нужны дополнительные модули и знание протоколов.
- Длительность работы от внешнего устройства питания: Потребляет единицы-десятки мА. Возможна автономная работа в течении недели.
- Контроль своевременности передачи сигналов: 100% контроль над временем и длительностью сигнала.
- Язык программирования: С подобный, С, С++.
- Возможность для работы с видео и компьютерным зрением: Отсутствует.

- Возможность работы со звуком: На мощных микроконтроллерах возможен синтез звука, но для работы с файлами формата MP3, OGG, WAV необходимы дополнительные модули.

2.1.2 Плата на микрокомпьютере

На основании среднестатистических данных можно выявить основные технические характеристики микрокомпьютера.

- Производительность: 1 или более ядер, сотни-тысячи МГц, сотни МБ оперативной памяти, единицы ГБ постоянной памяти.
- Многозадачность: присутствует и управляется операционной системой.
- Простота подключения к сети интернет: В большинстве случаев сетевой модуль установлен в микрокомпьютере и не требует дополнительных настроек.
- Длительность работы от внешнего устройства питания: Потребляет сотни-тысячи мА. Возможна автономная работа в течении нескольких часов.
- Контроль своевременности передачи сигналов: Существует возможность пропуска выполнения критического процесса из-за многозадачности.
- Язык программирования: Python, JavaScript, Bash и другие языки, доступные для операционной системы.
- Возможность для работы с видео и компьютерным зрением: Присутствует, HDMI-выход, OpenCV.
- Возможность работы со звуком: Поддерживает файлы формата MP3, OGG, WAV. Так же имеются аудиовыход HDMI или разъем 3,5 мм.

Оба типа этих контроллеров обладают разными преимуществами, а так же подходят для разработки в сфере интернета вещей для систем умного дома

или умного города. Для устройства “умная банка” существуют свои приоритетные технические характеристики, которыми являются:

- Время автономной работы, так как предполагается, что устройство будет использоваться длительное время и для удобства пользователя должно обладать малым энергопотреблением, что означает малое количество ядер и отсутствие многопоточности чтобы минимизировать затраты электроэнергии на проведение вычислений, сбор и обмена данными.
- Контроль своевременности передачи сигналов. Устройство рассчитано на точную работу по сбору и передачи данных о приеме медикаментов из контейнера. В случае если процесс не будет запущен своевременно может произойти потеря данных о приеме и в следствии отсутствие соответствующей информации в журнале приема у лечащего врача.
- Простота подключения к сети Интернет. Пользователь должен будет подключить устройство для того, чтобы устройство могло производить обмен данными с сервером и записывать данные о курсе приема медикаментов в журнал приема.

На основе технических характеристик контроллеров и приоритетных аспектов работы каждого типа платы было принято решение использовать микроконтроллер, так как он содержит в себе большинство приоритетных технических характеристик.

2.2 Сравнение технических характеристик плат на микроконтроллере

2.2.1 Плата Arduino Uno



Рисунок 2.4 - Плата Arduino Uno

Семейство плат Arduino дало огромный скачок в развитии электротехники. На данный момент большинство датчиков, сенсоров или плат расширения выпускаются под Arduino или делаются совместимыми с этим типом плат. Самой распространенной платой в этом семействе плат является плата Arduino Uno, ведь именно под нее и выпускается абсолютное большинство датчиков, плат расширений или сенсоров [12].

Основные технические характеристики платы Arduino Uno:

- Процессор на 16 МГц.
- 32 КБ постоянной памяти.
- 2 КБ оперативной памяти.
- 20 портов ввода-вывода.
- 6 аналоговых входов.
- 6 каналов ШИМ.
- 2 аппаратных прерывания.

Преимущества платы Arduino Uno

- Большое количество документации.
- Простая в освоении среда разработки Arduino IDE.
- Простой в освоении язык Arduino C++.
- Огромное количество готовых библиотек.
- Возможность подключить большое количество периферии, сенсоров и исполнительных устройств.
- Невысокая стоимость.

Недостатки платы Arduino Uno

- Отсутствие встроенного средства связи с сетью Интернет.

2.2.2 Плата Arduino Leonardo

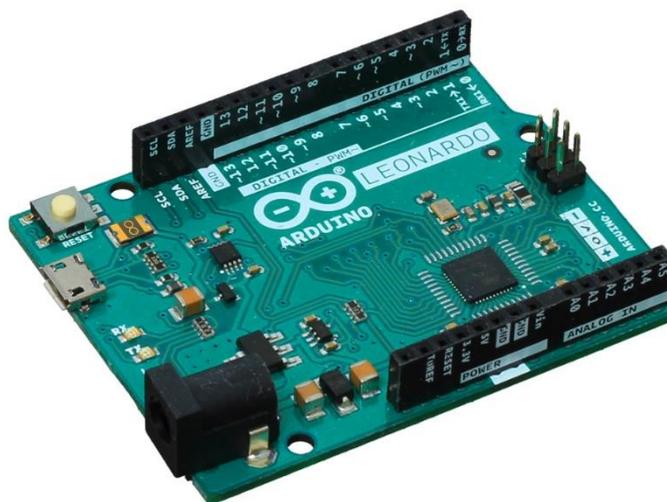


Рисунок 2.5 - Плата Arduino Leonardo

По существу, плата Arduino Leonardo – это модифицированная плата Arduino Uno, но в лучшую сторону. На этой плате используется микроконтроллер ATmega32u4 и именно он используется как преобразователь для прошивки в качестве USB-UART. Это и есть главное отличие платы Arduino Leonardo от остальных плат семейства Arduino. По сравнению с Arduino Uno данная плата имеет идентичные габариты и допустимый вольтаж [13].

Основные технические характеристики платы Arduino Leonardo:

- Процессор на 16 МГц.
- 32 КБ постоянной памяти.
- 2,5 КБ оперативной памяти.
- 20 портов ввода-вывода.
- 12 аналоговых входов.
- 7 каналов ШИМ.
- 5 аппаратных прерывания.

Преимущества платы Arduino Leonardo:

- Имеет возможность эмулировать HID-устройство.
- Простая в освоении среда разработки Arduino IDE.
- Простой в освоении язык Arduino C++.
- Имеет большее количество аналоговых выходов и каналов ШИМ.

Недостатки платы Arduino Leonardo

- Из-за отличия в распиновке с платой Arduino Uno некоторые датчики, сенсоры, платы расширения или управляющие устройства могут не подходить.
- Труднее в освоении. Большинство документации и библиотек для платы Arduino Uno.
- Отсутствие встроенного средства связи с сетью Интернет.

2.2.3 Плата Iskra Neo

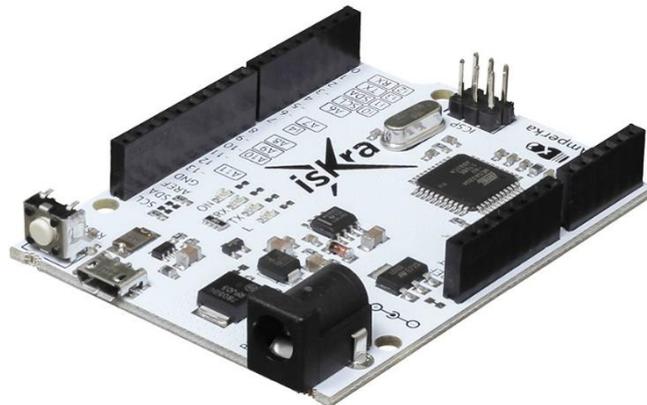


Рисунок 2.6 - Плата Iskra Neo

Плата Iskra Neo – это российский аналог платы Arduino Leonardo. Плата, как и Arduino Leonardo, имеет на борту микроконтроллер ATmega32U4. Для программирования этой платы используется упрощенная версия C++, известная как Wiring. Разработка может производиться из Arduino IDE, при указании того, что разработка ведется под плату Arduino Leonardo или графического языка программирования XOD IDE. Поддерживаемые операционные системы Windows, MacOS X и Linux, как и у Arduino Uno [14].

Основные технические характеристики платы Iskra Neo:

- Процессор на 16 МГц.
- 32 КБ постоянной памяти.
- 2,5 КБ оперативной памяти.
- 20 портов ввода-вывода.
- 12 аналоговых входов.
- 7 каналов ШИМ.
- 5 аппаратных прерывания.

Преимущества платы Iskra Neo:

- Имеет такие же преимущества, как и Arduino Leonardo.
- Имеет низкую цену.

Недостатки платы Iskra Neo:

- Имеет такие же недостатки, как и Arduino Leonardo.
- Имеет меньшую популярность и из-за этого для нее меньше документации, хотя большинство библиотек для Arduino Leonardo подходят.
- Отсутствие встроенного средства связи с сетью Интернет.

2.2.4 Плата Arduino Mini

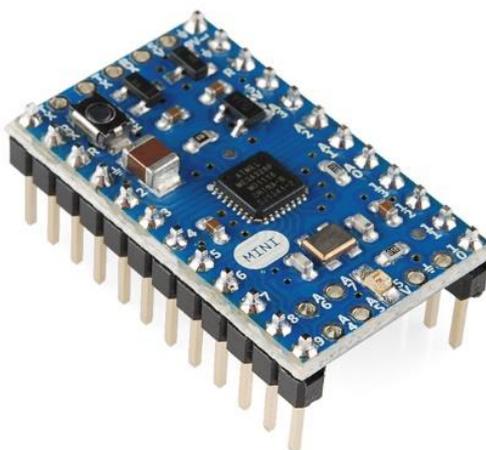


Рисунок 2.7 - Плата Arduino Mini

Arduino Mini – это та же самая плата Arduino Uno, только разработанная на более компактной плате. Это плата является самой компактной в семействе Arduino, ее размеры всего 30x18 мм. В качестве чипа здесь используется ATmega328p. Такая плата используется только там, где размер используемого аппаратного обеспечения критичен [15].

Основные технические характеристики платы Arduino Mini:

- Процессор на 16 МГц.
- 16 КБ постоянной памяти.

- 1 КБ оперативной памяти.
- 8 портов ввода-вывода.
- 8 аналоговых входов.
- 6 каналов ШИМ.

Преимущества платы Arduino Mini:

- Самая компактная плата в семействе.
- Легко начать работу, используя макетную плату.
- Подавать ток можно на один из контактов на выбор.

Недостатки платы Arduino Mini:

- Отсутствие собственного USB-порта, нужно использовать USB-UAR преобразователь для прошивки микроконтроллера.
- Дополнительные контакты на коротких сторонах не распаяны. Если необходимо их использовать, то следует припаять к ним провода или разъемы самостоятельно.
- Отсутствие встроенного средства связи с сетью Интернет.

2.2.5 Плата Arduino Micro



Рисунок 2.8 - Плата Arduino Micro

Arduino Micro – это та же самая плата Arduino Leonardo, только разработанная на более компактной плате. Эта плата является одной из самых компактных в семействе Arduino, ее размеры всего 48x17,7 мм. В качестве чипа здесь используется ATmega32u4. Прошивка такого микроконтроллера в этом случае происходит через micro-USB. Такая плата используется только там, где размер используемого аппаратного обеспечения критичен, а возможности должны быть более обширны чем у Arduino Mini [16].

Основные технические характеристики платы Arduino Micro:

- Процессор на 16 МГц.
- 32 КБ постоянной памяти.
- 2,5 КБ оперативной памяти.
- 20 портов ввода-вывода.
- 12 аналоговых входов.
- 7 каналов ШИМ.

Преимущества платы Arduino Micro:

- Одна из самых компактных плат в семействе Arduino.
- Легко начать работу, используя макетную плату.

Недостатки платы Arduino Micro:

- Для простой работы с этой платой нужно иметь макетную плату.
- Чтобы подключить устройство внешнего питания его выводы должны подключаться к выводам Gnd и Vin.
- При подаче напряжения меньше 7 В на выход с платы может подаваться менее 5 В, что может привести к некорректной работе платы.
- Отсутствие встроенного средства связи с сетью Интернет.

2.2.6 Плата Arduino Mega



Рисунок 2.9 - Плата Arduino Mega

Arduino Mega – это та же самая плата Arduino UNO, только разработанная на более мощном контроллере ATmega2560. Это плата является одной из самых больших в семействе Arduino, ее размеры 101x53 мм. Такой микроконтроллер и такие габариты позволили вместить больший функционал, что позволило увеличить количество портов, оперативную и постоянную память. Левая часть платы полностью соответствует плате Arduino Uno, что позволяет использовать платы расширения последней. Это значит, что при недостатке возможностей Arduino Uno ее можно быстро и легко подменить на более мощную плату Arduino Mega [17].

Основные технические характеристики платы Arduino Mega:

- Процессор на 16 МГц.
- 256 КБ постоянной памяти.
- 8 КБ оперативной памяти.
- 70 портов ввода-вывода.
- 16 аналоговых входов.
- 15 каналов ШИМ.

Преимущества платы Arduino Mega:

- Обладает большей памятью.
- Подходят платы расширения, что и для Arduino Uno.

- Возможность подключения большого количества сенсоров и датчиков.

Недостатки платы Arduino Mega:

- Большие габариты платы.
- Отсутствие встроенного средства связи с сетью Интернет.
- Высокая стоимость.

2.2.7 Плата Arduino Due



Рисунок 2.10 - Плата Arduino Due

Arduino Due является одной из самых производительных плат в семействе Arduino за счет микроконтроллера Cortex-M3, на котором и разработана плата. По форм фактору эта плата аналогична плате Arduino Mega, а это означает, что эта плата так же имеет возможность подключить модули расширения, доступные для Arduino Uno. У этой платы встроен контроллер шины CAN, который позволяет создавать сеть из плат Arduino Due, а также возможно организовать взаимодействие с автомобильной электроникой. Также плата имеет в своем распоряжении два канала ЦАП, которые позволяют синтезировать стереозвук с расширением 4,88 Гц [18].

Основные технические характеристики платы Arduino Due:

- Процессор на 84 МГц.
- 512 КБ постоянной памяти.

- 96 КБ оперативной памяти.
- 54 портов ввода-вывода.
- 12 аналоговых входов.
- 12 каналов ШИМ.

Преимущества платы Arduino Due:

- Возможность создавать сеть из плат Arduino Due.
- Возможность подключиться к автомобильной электронике.
- Обладает большей памятью.
- Подходят платы расширения, что и для Arduino Uno.
- Возможность синтезировать стереозвук.
- Возможность подключения большого количества сенсоров и датчиков.

Недостатки платы Arduino Due:

- Большие габариты платы.
- Отсутствие встроенного средства связи с сетью Интернет.
- Высокая стоимость.
- Родное напряжение платы 3,3 В, а не 5 В. Это означает, что не любая периферия может быть подключена в плате.

2.2.8 Плата Iskra JS

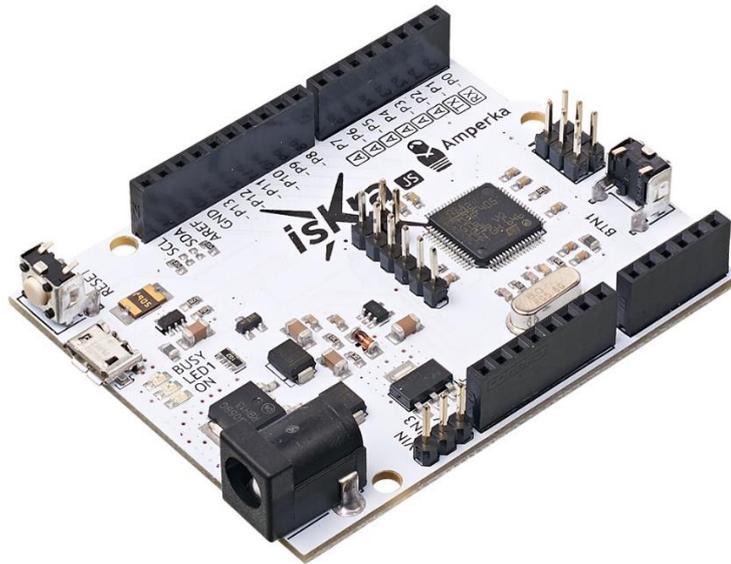


Рисунок 2.11 - Плата Iskra JS

Iskra JS – одна из самых удобных для написания программ плата. Разработка программ для нее ведется на языке JavaScript, который является языком высокого уровня, но он обладает своими особенностями использования. Такое возможно из-за того, что плата разработана на ядре Espruino. Разработка программ ведется в специальной среде Espruino Web IDE. Для обработки программ, написанных на JavaScript плата Iskra JS обладает одним из лучших аппаратных обеспечений среди плат разработанных на микроконтроллере [19].

Основные технические характеристики платы Iskra JS:

- Процессор на 168 МГц.
- 1 МБ постоянной памяти.
- 192 КБ оперативной памяти.
- 26 портов ввода-вывода.
- 12 аналоговых входов.
- 22 каналов могут генерировать ШИМ.

Преимущества платы Iskra JS:

- Возможность разработки на высоком языке программирования JavaScript.
- Максимальная совместимость с платами расширения, сенсорами и другими электронными модулями платы Arduino Uno.
- Снабжена предохранителем для защиты USB-порта компьютера для защиты от коротких замыканий или перенапряжения.

Недостатки платы Iskra JS:

- Из-за того, что среда программирования отличается от Arduino IDE, а также отличается экосистема может возникнуть проблема, связанная с отсутствием готовой библиотеки для подключаемой периферии, которую необходимо будет разработать самому.
- Отсутствие встроенного средства связи с сетью Интернет.

2.2.9 Плата Strela

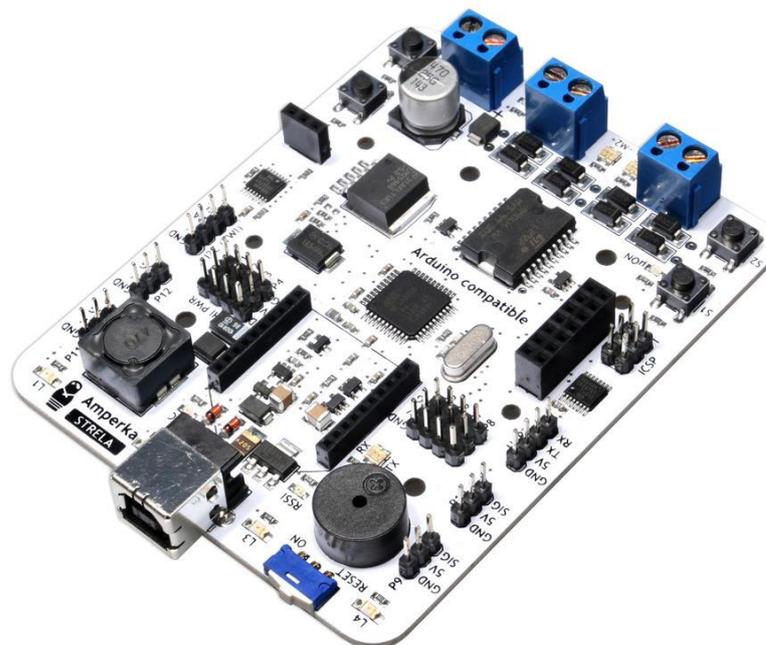


Рисунок 2.12 - Плата Strela

Strela – платформа, которая предназначена для создания легкого мобильного робота. Данная платформа имеет 4 разъема под сервоприводы, 4 кнопки и светодиода свободного назначения, встроенный драйвер для двух двигателей. Регулятор питания позволяет использовать множество источников питания. В основе этой платы находится тот же микроконтроллер, что и в Arduino Leonardo, что означает доступность возможности разработки в Arduino IDE, предварительно выбрав разработку под Arduino Leonardo [20].

Основные технические характеристики платы Strela:

- Процессор на 168 МГц.
- 32 КБ постоянной памяти.
- 2,5 КБ оперативной памяти.
- 12 портов ввода-вывода.
- 8 аналоговых входов.
- 4 каналов ШИМ.

Преимущества платы Strela:

- Хорошо подходит для робототехники.
- Разработка программ можно проводить в Arduino IDE или XOD IDE.
- Большинство сенсоров, которые совместимы с Arduino Uno, подходят для Strela.

Недостатки платы Strela:

- Невозможно установить платы расширения Arduino Uno.
- Нумерация контактов отличается от Arduino Leonardo, поэтому для работы с пинами платы используются другие функции.
- Высокая стоимость.
- Отсутствие встроенного средства связи с сетью Интернет.

2.2.10 Плата Arduino Yún



Рисунок 2.13 - Плата Arduino Yún

Arduino Yún – это гибрид, разработанный на основе платы Arduino Leonardo и микроконтроллера OpenWRT Linux. Часть с Arduino разработана на основе микроконтроллера ATmega32u4, распиновка совпадает с Arduino Leonardo. Встроенный модуль Wifi позволяет прошивать плату без ее физического подключения к компьютеру. Для разработки нужно использовать Arduino IDE или XOD IDE [21].

Основные технические характеристики платы Arduino Yún:

Микроконтроллер Arduino:

- Процессор на 16 МГц.
- 32 КБ постоянной памяти.
- 2,5 КБ оперативной памяти.
- 20 портов ввода-вывода.
- 12 аналоговых входов.
- 7 каналов ШИМ.

Микропроцессор Linux

- Процессор Atheros AR9331.

- Архитектура MIPS @400MHz.
- Рабочее напряжение 3,3 В.
- Флеш-память 16 Мб.
- Оперативная память 64 МБ DDR2.
- Ethernet IEEE 802.3 10/100Mbit/s.
- Wifi IEEE 802.11b/g/n.

Преимущества платы Arduino Yún:

- Встроенного средство связи с сетью Интернет через Wifi или Ethernet.
- Походит как головное устройство в системе умный дом.
- Легко интегрируется с веб-сервисами и социальными сетями.
- Подходят платы расширения, что и для Arduino Uno.
- Возможность работать с мультимедией.

Недостатки платы Arduino Yún:

- Очень высокая стоимость.
- Чувствительно к напряжению в 5 В, так как отсутствует встроенный регулятор напряжения, что может привести к выходу устройства из эксплуатации.

2.2.11 Плата Netduino 2

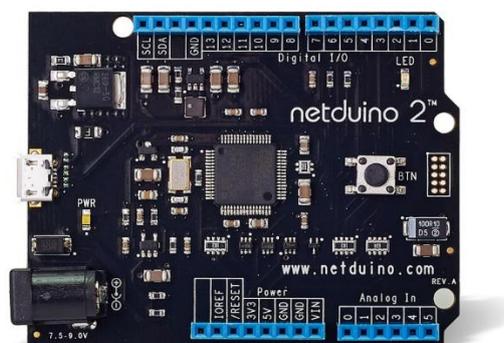


Рисунок 2.14 - Плата Netduino 2

Netduino 2 по форм фактору похож на Arduino Uno, что позволяет использовать те же платы расширения. Главным отличием от множества плат с микроконтроллером является возможность разработки на платформе .NET. Благодаря мощному ARM-микроконтроллеру STM32F205RF с ядром Cortex-M3 плата поддерживает разработку в среде программирования Microsoft Visual Studio с использованием стандартной библиотеки .NET Micro Framework [22].

Основные технические характеристики платы Netduino 2:

- Процессор на 120 МГц.
- 192 КБ постоянной памяти.
- 60 КБ оперативной памяти.
- 22 портов ввода-вывода.
- 6 аналоговых входов.
- 6 каналов ШИМ.

Преимущества платы Netduino 2:

- Возможность разработки в среде Microsoft Visual Studio.
- Большинство плат расширения Arduino Uno подходят.

Недостатки платы Netduino 2:

- Отсутствие встроенного средства связи с сетью Интернет.
- Очень высокая стоимость.
- Малое количество документации и библиотек.

2.2.12 Плата Netduino Plus 2

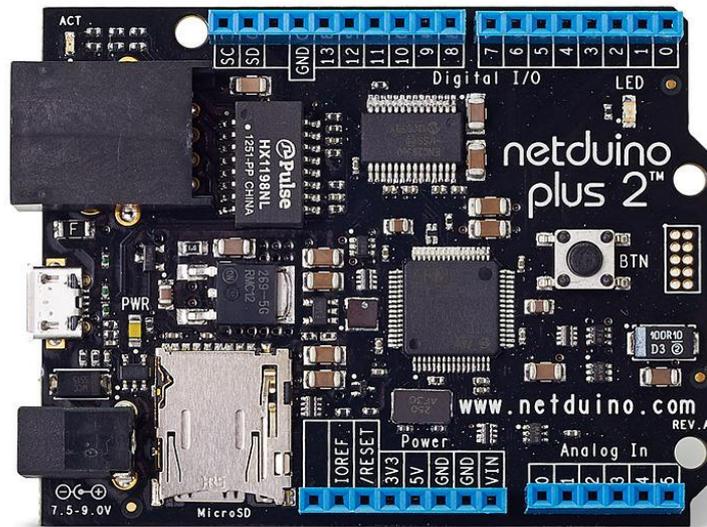


Рисунок 2.15 - Плата Netduino Plus 2

Netduino Plus 2 очень похожа на Netduino 2, но главной отличительной чертой является наличие встроенного 10-мегабитного Ethernet-порта. Для работы с таким дополнением необходимо использовать более мощный микроконтроллер STM32F405RG с ядром Cortex-M4. Такое дополнение позволит создавать веб-сервисы с помощью .NET Micro Framework. Так же как и Netduino 2 эта плата совместима с большинством модулей расширения для плат семейства Arduino [23].

Основные технические характеристики платы Netduino 2:

- Процессор на 168 МГц.
- 384 КБ постоянной памяти.
- 100 КБ оперативной памяти.
- 22 портов ввода-вывода.
- 6 аналоговых входов.
- 6 каналов ШИМ.

Преимущества платы Netduino 2:

- Возможность разработки в среде Microsoft Visual Studio.

- Большинство плат расширения Arduino Uno подходят.
- Имеет возможность для подключения к сети Интернет.

Недостатки платы Netduino 2:

- Очень высокая стоимость.
- Малое количество документации и библиотек.

Подводя итоги сравнения плат на микроконтроллерах было принято решение вести разработку на платформе Arduino Uno в связи с тем, что эта плата имеет наибольшее количество библиотек для работы с датчиками, сенсорами или платами расширений. Так же Arduino Uno обладает наибольшим количеством библиотек, что позволяет с легкостью начать работу практически с любой периферией. Наибольшее количество таких сенсоров и датчиков, модулей расширения и другой периферии или подходит или разрабатывается под стандарты Arduino Uno. Разработка программного обеспечения для такой платы проходит в удобной IDE и на «С» подобном языке программирования, что позволяет быстро адаптироваться к написанию скетчей. Одним из самых важных аспектов выбора данной платы – это одна из самых низких стоимостей на рынке, что позволяет снизить стоимость разрабатываемого устройства.

2.3 Выбор способа передачи данных

Для организации обмена информацией между платой на микроконтроллере Arduino Uno и сервером необходимо выбрать канал связи. На данный момент возможно обмениваться информацией между Arduino Uno и сервером тремя способами: сеть Bluetooth, сеть Интернет и мобильная сеть. Далее мы рассмотрим каждую из них.

2.3.1 Связь по Мобильной сети

Способ связи по стандарту GSM очень распространен с настоящее время. Он позволяет обмениваться информацией серверу и плате с микроконтроллером посредством отправки текстовых сообщений SMS или посредством входящих и исходящих звонков. Для работы в мобильной сети необходимо иметь SIM-карту оператора связи, так как для работы по GSM-модулю необходимо иметь и SIM-карту, с помощью которой модуль будет обмениваться информацией между устройством и сервером [26]. Достоинствами связи по мобильной сети являются низкая стоимость, возможность работы практически на всей поверхности Земли, где есть покрытие сотовым оператором, а также простая установка и работа. Достаточно подключить расширения с SIM-картой. Недостатками способа являются необходимость оплачивать услуги сотового оператора для работы устройства и отсутствие возможности управления устройством, если на телефоне пользователя не будет средств.

2.3.2 Связь по сети Bluetooth

Одним из способов обмена информации является канал Bluetooth, так как он является одним из самых известных и часто используемых протоколов безопасности. Bluetooth в Ардуино позволяет объединить сразу несколько устройств по беспроводному каналу связи. Такое подключение позволяет передавать информацию с датчиков, подключенных к Ардуино и наоборот, получать информацию с сервера на плате Ардуино [27]. К достоинствам данного способа связи можно отнести высокую помехоустойчивость, простоту использования и низкое энергопотребление. К недостаткам можно отнести малое расстояние, в котором возможна работа между платой и сервером.

2.3.3 Связь по сети Интернет

Так же существует способ обмена информации через сеть Интернет. На данный момент практически любой человек имеет в своем доме сеть Интернет, что показывает популярность обмена данными по этой сети. По сети Интернет возможно объединить неограниченное количество устройств. Скорость передачи данных по таким каналам возможна до 100 Мбит/с. Для работы с такой сетью достаточно подключить плату расширения к устройству и подсоединить кабель Ethernet в стандартный разъем RJ45 [28]. Достоинствами этого способа связи являются простота установки, наименьшее количество помех среди всех способов связи, высокая скорость обмена данными, а также она наиболее защищена от постороннего вмешательства. Главным недостатком способа является уменьшение мобильности устройства.

В итоге можно сделать выводы по каждому способу связи и определить самый подходящий способ связи платы Arduino Uno с сервером.

Изначально предполагается, что врач отслеживает курс лечения пациента удаленно, что означает, что канал связи Bluetooth не подходит, потому что по такому каналу связи возможно передавать сообщение только на небольших расстояниях.

Мобильная сеть позволяет передавать информацию на большие расстояния, так как сотовые вышки, через которые передается информация покрывает практически всю поверхность земного шара. Однако в этом случае придется использовать SIM-карту оператора, чьи вышки и покрывают площадь Земли. Если использовать SIM-карту, то придется платить за ее использование по тарифу, подключенному оператором связи. Такой способ нам не подходит, так как мы стремимся минимизировать стоимость производства и эксплуатации.

Самой подходящей сетью передачи данных остается сеть Интернет. Сеть Интернет покрывает большую часть поверхности Земли, а так же большая часть населения планеты ей пользуется. При использовании этой сети пользователю не нужно будет доплачивать за использование этой сети, через которую устройство будет обмениваться информацией с сервером. Так же самым надежным подключением, которое будет гарантировать минимизацию помех при передаче данных и безопасность соединений является проводное подключение к сети Интернет. Так же установка и настройка маршрутизаторов беспроводной сети и подключение устройств к такой сети имеет большую сложность.

2.4 Сравнение Ethernet-модулей для Arduino Uno

В настоящее время существует несколько способов проводного подключения платы Arduino Uno с сетью Интернет. На рынке в настоящее время представлены платы расширения и модули подключения. Плата расширения – плата, по форм-фактору совпадающая с платой, к которой она подключается и обеспечивает дополнительный функционал, то есть расширяет функционал платы. Модуль подключения – специальный модуль, который подключается с помощью проводов к соответствующим выходам на плате с микроконтроллером и обеспечивает дополнительный функционал. Далее мы сравним наиболее распространенные модели каждого типа проводного подключения.

2.4.1 Плата расширения Ethernet Shield W5100

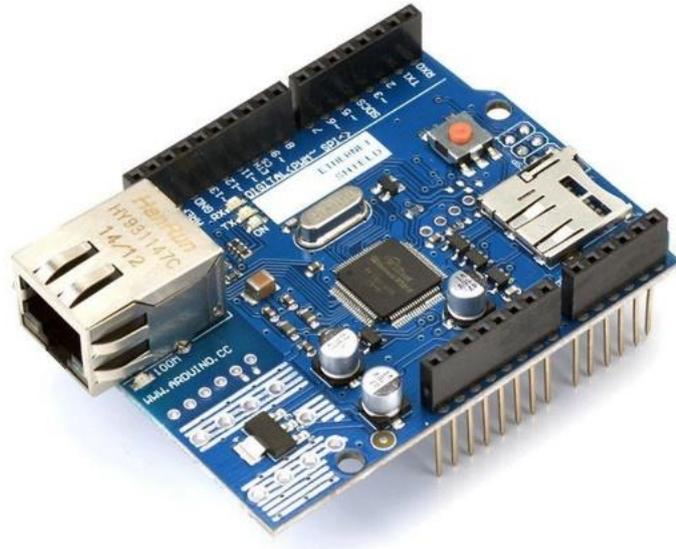


Рисунок 2.16 - Плата расширения Ethernet Shield W5100

Плата расширения Ethernet Shield позволяет осуществить соединение и реализовать обмен данными с любыми устройствами Ардуино в сети интернет. Достаточно подключить модуль к Ардуино с помощью кабеля RJ-45 и с помощью встроенной библиотеки настроить соединение с сервером [29]. Эта плата расширения собрана на базе Ethernet-контроллера Wiznet W5100, который поддерживает стек сетевых протоколов и позволяет работать с TCP, а так же и с UDP-протоколами. Так же плата расширения Ethernet Shield позволяет использовать Ардуино как веб-клиента, так и веб-сервер. Представленный модуль имеет напряжение питания в 5В, контроллер W5100, скорость подключения варьируется от 10 до 100 Мбит/с, имеется разъем для карты памяти micro-SD и размер буфера в 16 Кб. Главными достоинствами модуля являются высокая скорость обмена данными, защита от явлений перезагрузки и короткого замыкания, возможность эффективно преобразовывать напряжение, а так же изоляция

в 1500 Вольт на точке между «вход-выход». Главным недостатком модуля являются большие габариты (74 x 54 мм).

2.4.2 Модуль подключения ENC28J60

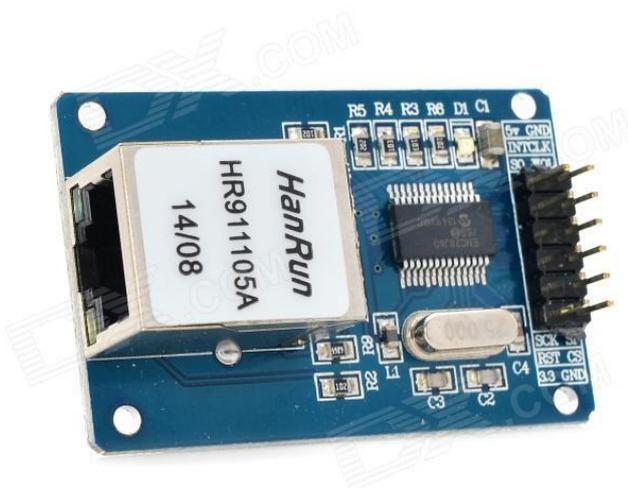


Рисунок 2.17 - Модуль подключения ENC28J60

Модуль подключения ENC28J60 позволяет подключить плату Arduino Uno к сети Интернет. Этот чип имеет всего 28 контактов, работает через интерфейс SPI, который доступен на любом контроллере. Он подключается с помощью проводов к определенным выходам на плате Arduino. Этот модуль так же позволит плате Arduino Uno осуществлять прием и передачу информацию по сети Интернет, а так же использует разъем RJ45 для подключения Ethernet кабеля [30]. Данный модуль имеет напряжение питания в 3,3В, скорость подключения 10 Мбит/с, протокол обмена данными SPI и размер буфера в 8 Кб. Главными достоинствами модуля являются маленькие размеры (30 x 24 мм) и низкая стоимость. Недостатками модуля являются меньшая скорость передачи данных из-за используемого протокола SPI, маленькое количество информации о работе

с модулем, а так же необходимость в установке дополнительных библиотек для работы с модулем.

Подводя итоги сравнения модулей подключения к сети Интернет мы остановились на выборе Ethernet Shield W5100, так как этот модуль более защищен от изменения напряжений в сети, которые могут повлечь за собой выход из строя всего устройства, имеет отличную изоляцию в 1500В в точках «вход-выход», в среде разработки Arduino IDE предустановлены библиотеки для работы с этой платой расширения, а так же имеется больше информации о работе и подключении Ethernet Shield. Так же ENC28J60 имеет меньшую скорость передачи данных по сети Интернет из-за поддерживаемого стандарта IEEE 802.3i . В этом стандарте физическим уровнем является 10BASE-T, что и дает всего 10 Мбит/с по витой паре. На плате Ethernet Shield аппаратно реализованы TCP/IP и Ethernet протоколы, а во второй - лишь Ethernet.

2.5 Выбор датчика открытия банки

В качестве датчика открытия мы будем использовать датчик магнитного поля. Для плат Arduino предоставляется несколько видов таких датчиков, в основе работы которых лежит геркон (сокращение от герметизированный магнитоуправляемый контакт) с нормально разомкнутым контактом [31]. Использование такого контакта означает, что при приближении магнита к герметичной стеклянной колбе под действием силовых линий магнитного поля два проводника в трубке намагничиваются и притягиваются друг к другу. В случае, если линии магнитного поля не действуют на проводники, то они размыкаются.

К наиболее доступным модулям можно отнести KY-021 и KY-025. Заметим, что наиболее популярным можно назвать модуль KY-021 (рис. 2.18).

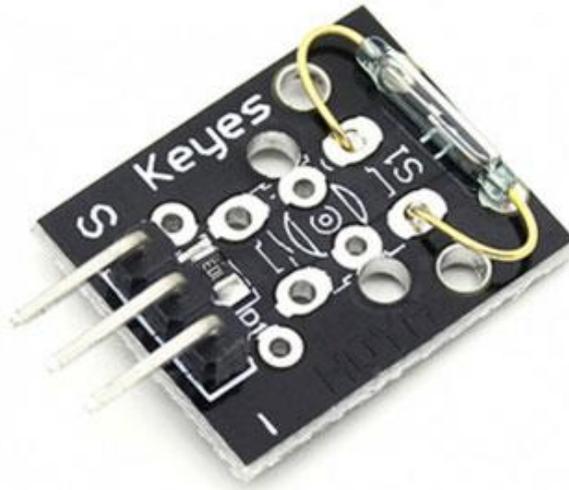


Рисунок 2.18 - Модуль KY-021

На данной плате установлен резистор 10 кОм между контактами «+5» и «S». Датчик необходимо закрепить на том элементе конструкции, который будет неподвижен, а на подвижной надо разместить магнит. При отсутствии магнита рядом с датчиком на цифровой выход подается «1», если же магнит близко, то на цифровом выходе будет «0» [32]. Основным достоинством данного модуля является его компактность (15 x 19 мм) при оптимальной дальности срабатывания (10 мм). Однако же, он дороже модуля KY-025.

Модуль KY-025 (рисунок 2.19) похож на модуль KY-021, на нем так же установлены компаратор, два светодиода и подстроечный резистор. Один из светодиодов отвечает за сигнализацию о подаче питания на датчик, другой за наличие магнитного поля в радиусе действия датчика.

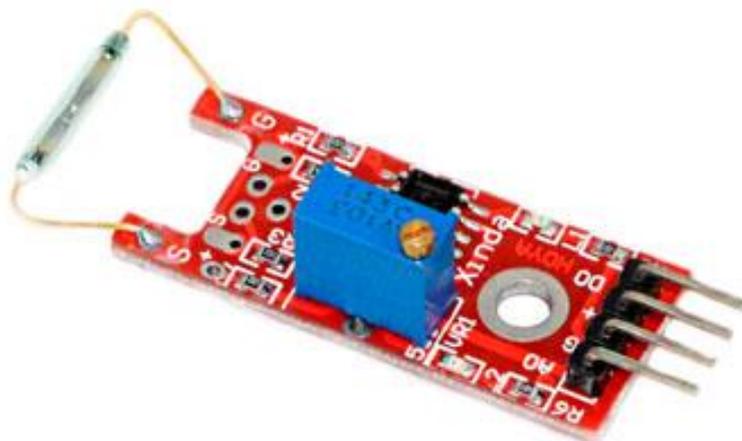


Рисунок 2.19 - Модуль KY-025

Однако, в отличии от КУ-021, у данного датчика имеется два выхода: аналоговый и цифровой. Если магнитного поля нет, то цифровой выход выдает логический 0, при наличие магнитного поля – логический 1. Аналоговый выход выдает 0 при наличии магнитного поля и 5В при его отсутствии [33]. Дальность срабатывания у данного модуля выше (15 мм), но в нашем случае это скорее недостаток, так как при открытии крышки датчик должен сработать на открытие без сбоев, а если дальность срабатывания будет слишком большой, то открытие может быть не засчитано из-за того, что магнит на крышке датчика будет на меньшем расстоянии, чем 15 мм и датчик не сработает. Кроме того, он заметно больше чем КУ-021 (45 x 20 мм).

Таким образом, мы считаем верным выбрать КУ-021, так как он дает нам наилучшую комбинацию характеристик относительно его размеров и дальности срабатывания, при этом для опытного устройства не так важна разница в стоимости модулей. Так же в устройстве не будет использоваться аналоговый выход, так как достаточно информации с цифрового выхода датчика.

2.6 Результат сравнительного анализа аппаратной части

По результатам из вышеприведенного анализа была выбрана как самая подходящая плата – микроконтроллер. Данное устройство было выбрано из-за того, что оно обладает большим временем автономной работы из-за более низкого потребления электроэнергии, а также из-за гарантии своевременности передачи сигнала. Так как выбор был сделан в пользу микроконтроллера, то самая подходящая плата – это Arduino Uno по нескольким причинам. Под Arduino Uno написано огромное количество библиотек, которые позволяют работать практически с любой периферией. Большое количество модулей подключения, плат расширения, датчиков и

сенсоров разрабатывается и производится под стандарты выбранной платы. Для разработки программного обеспечения под Arduino Uno используется очень удобная среда разработки – Arduino IDE, а сам язык разработки – C подобный, что позволяет быстро адаптироваться под написание скетчей.

Для обеспечения связи был выбран способ обмена через сеть интернет по Ethernet-кабелю в связи с тем, что интернетом пользуются практически на всей поверхности земного шара, что означает отсутствие доплаты за использование данного способа связи в отличие от связи по мобильной сети. Обмен данными через сеть Интернет является самым надежным подключением, которое минимизирует помехи и потери данных при передаче. Для реализации данного способа была выбрана плата расширения Ethernet Shield W5100, так как она защищена от изменений напряжения в сети, которые могли бы повлечь за собой полный выход из строя устройства. Для работы с этой платой расширения в среде разработки Arduino Uno уже имеются библиотеки и примеры. Также Ethernet Shield обладает большей скоростью передачи данных.

Для отслеживания взаимодействия с устройством был выбран датчик магнитного поля KY-021 по нескольким причинам, таким как маленький размер, относительно других магнитных датчиков, а также данный модуль имеет меньшую дальность срабатывания, что минимизирует погрешности работы и точность собираемых данных.

2.7 Реализация аппаратной части

После проведенного анализа и выбора подходящих комплектующих необходимо разработать схему подключения. Такую схему подключения возможно реализовать с помощью эмуляторов для Arduino. Самым популярным сервисом для реализации схематичной сборки является

TinkerCAD. Это онлайн сервис, который входит в состав программного обеспечения от Autodesk, которая является самой известной CAD-системой в мире. TinkerCAD одна из самых простых и бесплатных сред для обучения 3D-моделированию с помощью которой легко создавать и отправлять на 3D-печать. Это программа – онлайн платформа, то есть для работы в ней достаточно иметь браузер. В сервисе имеются отсортированные наиболее популярные электронные компоненты, что позволяет быстро и удобно начать разработку. Данный сервис предлагает встроенный редактор Arduino с монитором порта и возможность пошаговой отладки. Так же сервис имеет визуальный редактор Arduino. TinkerCAD так же имеет огромное количество готовых проектов, что позволяет сразу иметь схему подключения датчиков с микроконтроллером. Так же сервис позволяет включить режим симуляции и протестировать работу собранной системы, изменять показания датчиков, чтобы следить за тем как реагирует система [34].

В результате работы в онлайн-сервисе TinkerCAD была спроектирована схема подключения микроконтроллера Arduino Uno к плате расширения Ethernet Shield, а также подключение красного светодиода для сигнализации ошибки, желтого светодиода для сигнализации о необходимости закрыть банку и зеленый светодиод для сигнализации о необходимости принять медикаменты. Был подключен датчик магнитного поля для отслеживания состояния открытия умной банки. Данная схема представлена на рисунке 2.20

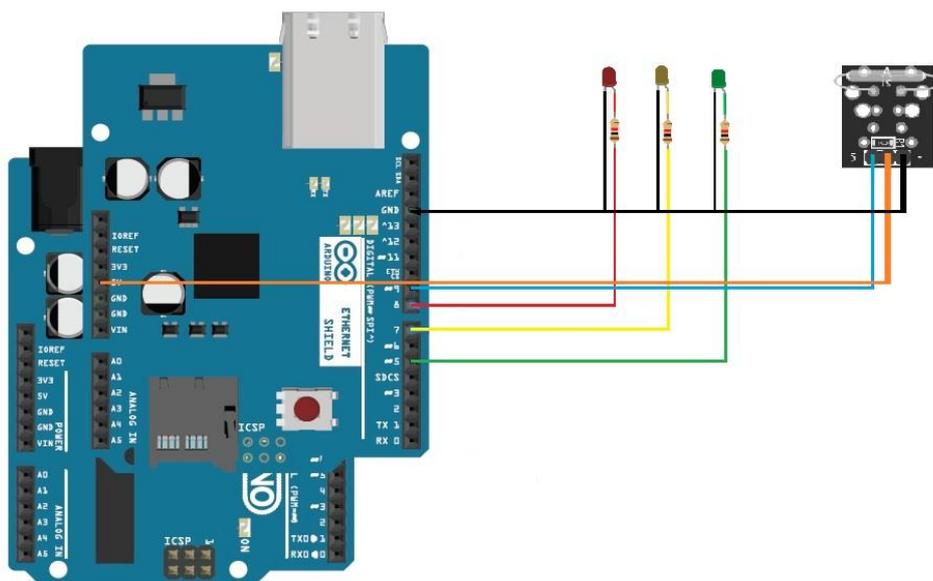


Рисунок 2.20 - Схема прототипа умной банки

Начинать работу необходимо с подключения платы расширения Ethernet Shield W5100 к плате с микроконтроллером Arduino Uno. Из-за того, что форм-фактор подключаемого модуля совпадает с форм-фактором платы, то для подключения достаточно разместить Ethernet Shield над платой так, чтобы разъем для подключения кабеля Ethernet платы расширения находился над разъемом для подключения кабеля USB платы Arduino Uno. После подключения контакты, расположенные на Ethernet Shield'e начинают работать как контакты платы Arduino Uno, а это значит, что в дальнейшем необходимо подключать все провода именно к контактам платы расширения.

Для организации работы светодиода на оптимальной яркости необходимо к анодам всех светодиодов подключить резистор на 220 Ом. Далее соединить резистор и цифровой выход как показано на схеме. Красный светодиод, сигнализирующий ошибку подключения к сети Интернет, необходимо подключить к цифровому выходу номер 8. Желтый светодиод, сигнализирующий о необходимости закрыть крышку устройства, подключается к цифровому выходу 7. Зеленый светодиод, сигнализирующий о необходимости принять медикаменты, входит в

цифровой выход 5. К катодам всех светодиодов подключаются провода для заземления. На плате Arduino Uno имеется ограниченное количество разъемов для подключения заземления из-за которого необходимо объединить заземление для всех светодиодов и подключить получившийся провод к свободному заземлению на плате.

Заканчивается сборка подключением модуля магнитного поля. Для его корректного функционирования необходимо в правильной последовательности подключить контакты модуля к контактам платы. Контакт, расположенный около надписи на плате модуля «-» - является заземлением. Его необходимо подключить к контакту заземления на плате с микроконтроллером. Следующий контакт необходимо подключить к свободному разъему на плате с надписью «5V». Последний контакт необходим для считывания информации с датчика и его необходимо подключить к цифровому выходу 9. Схема подключения модуля представлена на рисунке 2.21.

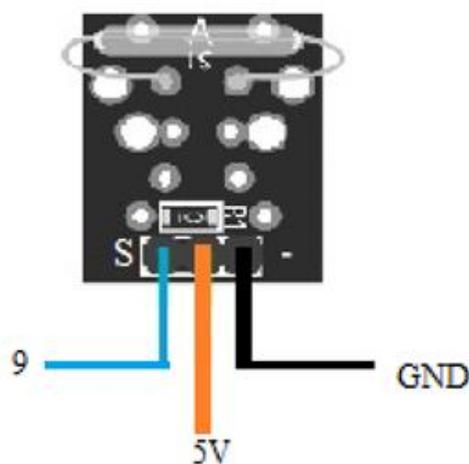


Рисунок 2.21 - Схема подключения датчика магнитного поля

Последним шагом является закрепление магнита на крышке банки, который будет взаимодействовать с сенсором открытия/закрытия. При поднесении, магнит будет замыкать два разомкнутых контакта в вакуумной колбе, расположенных на датчике магнитного поля.

После реализации сборки всех комплектующих в правильном порядке, опираясь на схему, сконфигурированную в эмуляторе TinkerCAD был собран прототип устройства «умная банка», изображенный на рисунке 2.22

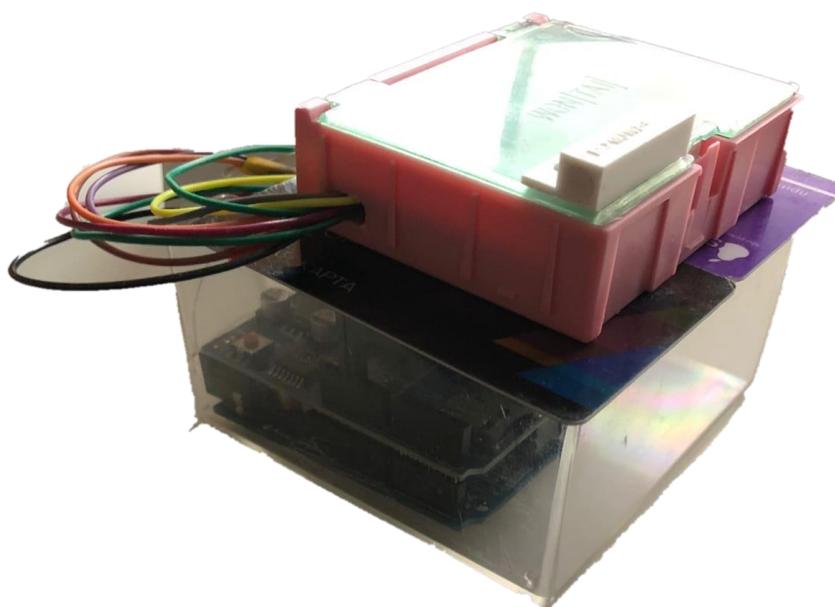


Рисунок 2.22 - Прототип устройства системы «умная банка»

Исходный код, реализующий логику работы описанного устройства представлен в Приложении А.

2.8 Процесс подключения и эксплуатация устройства

После сборки устройства для ввода его в эксплуатацию необходимо подключить питание и Ethernet-кабель. Для подключения питания можно воспользоваться двумя способами: первый – необходимо взять щелочной элемент питания формата MN1604 и специально предназначенный провод питания для Arduino, подсоединить элемент питания к одному концу провода, а другой конец подсоединить к плате Arduino в специальный разъем расположенный с торцевой части устройства; второй – необходимо взять шнур USB Defender AM-BM, подсоединить концом с разъемом USB

2.0 к источнику питания, а другим концом к плате с торцевой части(см. рис. 2.23).

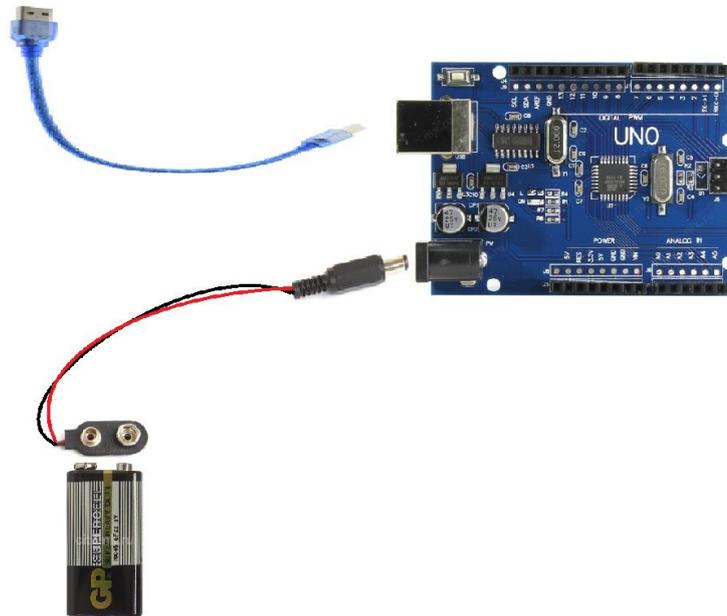


Рисунок 2.23 - Подключение питания к устройству

После подключения питания к плате Arduino и Ethernet-кабеля устройство начинает загружать программу, которая хранится в памяти и работать по описанному там алгоритму. С этого момента начинается эксплуатация устройства. В случае, если произойдет ошибка соединения с сервером системы внутри банки, в которой хранятся медикаменты загорится красный светодиод.

Если красная лампочка не горит, то это означает, что устройство соединилось с сервером и готово к работе. Устройство находится в режиме ожидания и не записывает никакие данные пока не придет сигнал с сервера о необходимости принять медикаменты. Как только такой сигнал приходит устройство переходит в режим чтения данных с датчика магнитного моля, который в нашем случае является датчиком открытия/закрытия крышки банки. Также включается световая индикация, а в частности загорается

зеленый светодиод, который означает, что время приема медикаментов наступило и необходимо их принять.

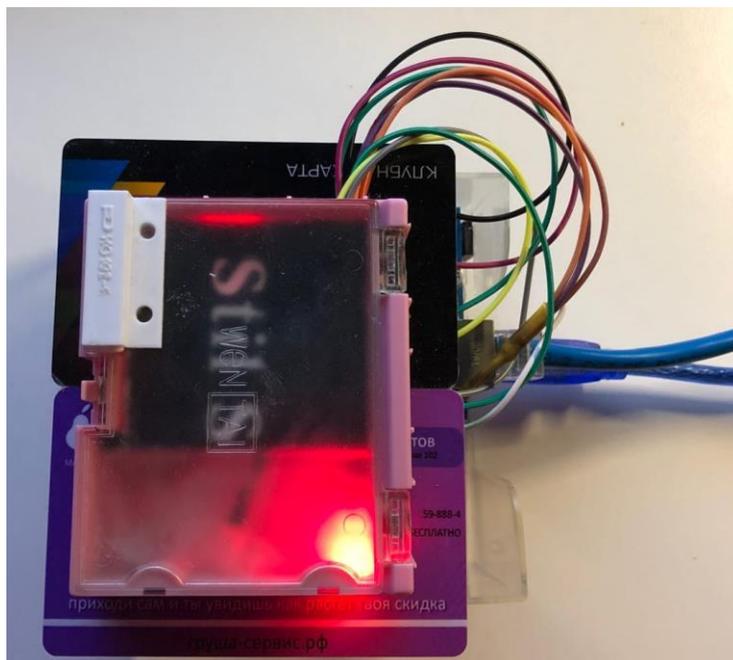


Рисунок 2.24 - Ошибка соединения с сервером

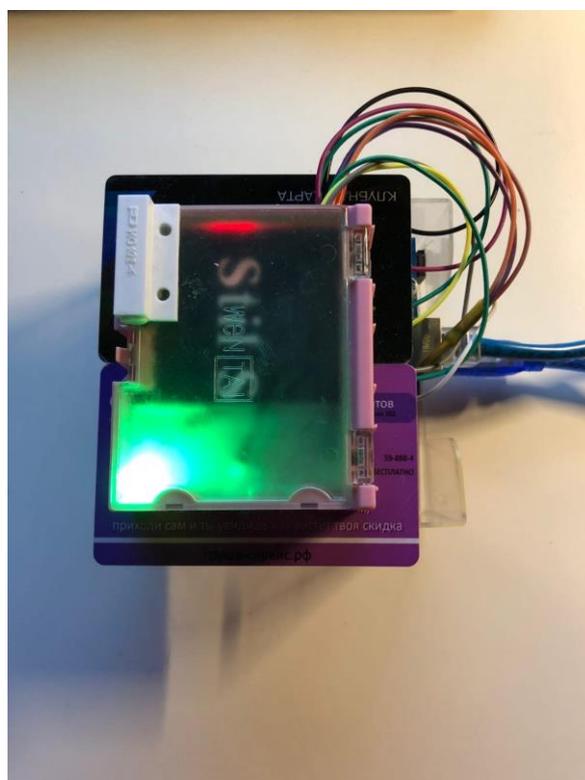


Рисунок 2.25 - Сигнализация о необходимости приема медикаментов

После того, как пользователь увидит световую индикацию ему необходимо открыть крышку, чтобы достать медикаменты из устройства. Когда крышка откроется, то магнит отойдет на достаточное расстояние от датчика, чтобы он сработал и записал данные об открытии банки. Также включается дополнительная световая индикация, а в частности желтый светодиод и пользователь увидит горящие зеленый и желтый светодиод. Такая световая индикация так же сигнализирует о необходимости закрыть крышку банки, чтобы устройство считало прием успешным.

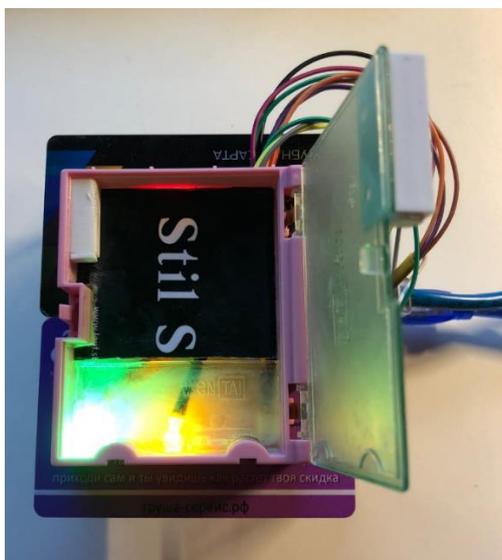


Рисунок 2.26 - Сигнализация о необходимости закрыть крышку банки

После того, как пользователь закроет крышку вся световая сигнализация погаснет, что означает успешность приема медикаментов. В случае, если пользователь не уложился в отведенный интервал для приема световая сигнализация так же погаснет (см. рис 2.27). Далее устройство отправляет информацию с датчика на сервер, где происходит обработка данных, и информация записывается как локально, так и отправляется на внешний веб-сервис.



Рисунок 2.27 - Выключенная сигнализация

Описанный процесс может быть использован как инструкция по использованию устройства.

3 Разработка программного обеспечения серверной части системы

Для управления устройствами на основе микроконтроллеров или микропроцессоров через интернет необходимо иметь программное обеспечение, которое позволило бы работать с этим устройством.

В дальнейшем мы будем называть наше программное обеспечение “сервер”.

Для реализации такого программного обеспечения необходимо реализовать набор некоторого функционала, такого как обмен информацией с устройством, которое позволило бы устройству отправлять сведения о своей работе и сведения, полученные с датчиков, а также сообщать о неисправности. Сервер должен иметь возможность записывать и хранить все данные, полученные с умной банки, а также отображать эту информацию. Для корректной работы умной банки необходимо отсылать информацию о включении световой сигнализации о необходимости принять медикаменты, а также отсылать информацию о необходимости отправить полученные данные с датчиков, и такой функционал об отправке информации должен реализовываться в себе сервер.

Чтобы полностью разобраться с функционалом и путями работы сервера необходимо воспользоваться уже известной методологией моделирования IDEF0. С помощью диаграммы уровня A0 диаграммы методологии IDEF0 можно полностью разобраться со способами взаимодействия разработанного прототипа умной банки с сервером, а также обмен информации сервера с веб-приложением.

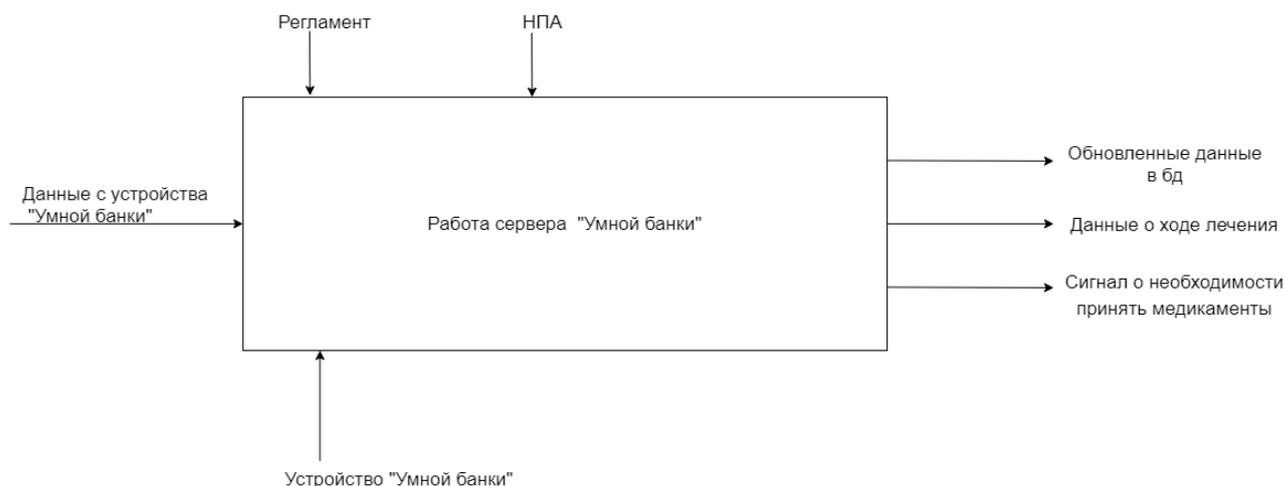


Рисунок 3.1 - Диаграмма уровня А0 для работы сервера «Умной банки»

На диаграмме показана общая постановка задачи: «Работа сервера «Умной банки»». Ранее было описано, что диаграмма IDEF0 состоит из четырех типов стрелок: вход, управление, механизм и выход. Для работы сервера на него должны поступить входные данные, которыми являются данные с устройства о приеме медикаментов. Механизмами управления для работы сервера выступают регламент и нормативно-правовые акты. Эти документы регулируют работу сервера. Для того, чтобы произвести необходимую работу используется устройство, которое присылает вышеупомянутые данные. В результате работы сервера должны быть выходные данные, которыми являются обновленные данные в базе данных, данные о ходе лечения для внешнего веб-приложения, которое будет работать с данными, а также сигнал о необходимости принять медикаменты, отправка которого предназначена для устройства «умной банки».

3.1 Язык программирования PHP

PHP – скриптовый язык общего назначения, использующийся для веб-разработки. Имеет высокую популярность среди других языков, применяющихся для создания динамических веб-сайтов. В 1994 году Расмус

Лерддорф разработал на языке программирования С набор скриптов, который позволял отслеживать количество посетителей веб-сайта с помощью обработки шаблонов HTML-документов. Данный набор скриптов был назван Personal Home Page Tools. Первый публичный релиз состоялся 8 июня 1995 года. С появлением PHP 2 язык стал использоваться как язык скриптов на стороне сервера. С тех пор язык набирает популярность среди разработчиков и занимает лидирующую долю на рынке среди языков программирования для написания веб-сервисов [35]. Особенности языка программирования PHP выступают открытость исходного кода, который означает бесплатное использование. Язык имеет несколько predefined констант для уведомлений об ошибках или предупреждениях. Поддерживает множество баз данных. Предлагает регистрацию доступа, которая позволяет создавать сводку последних событий. В качестве преимуществ языка программирования PHP [36] называют работоспособность на различных платформах, таких как Microsoft Windows, Unix, Linux, MacOS, простота в изучении, бесплатное использование, эффективная работа на стороне сервера, поддержка абсолютным большинством веб-серверов и наибольшее количество документации. К недостаткам языка программирования PHP относятся отсутствие статической типизации, отсутствие возможности компилировать ни в исходный код, ни в промежуточный байт-код, не подходит для разработки настольных приложений, ограниченный функционал для отслеживания ошибок.

3.2 Серверная платформа

Для развертывания сервера приложения было выбрано программное обеспечение OpenServer. Это программное обеспечение является профессиональным инструментом для веб-разработки под операционную систему Windows. Данное ПО – это переносимый локальный WAMP/WNMP

сервер, который имеет многофункциональную управляющую программу и большой выбор подключаемых компонент. Главная идея OpenServer – независимость от рабочего места.[37]

Разработка данного программного обеспечения началась 1 августа 2010 года. Старт разработки связан с многолетним ожиданием выхода обновления для другой известной WAMP системы – Denwer. Разработка велась для Microsoft Windows и называлась PHP Devel Studio, но из-за особенностей среды разработки программа была написана сначала на языке Delphi начиная с версии 4.0. На данный момент программа находится в фазе активной разработки.

Поскольку данное ПО не требует установки, то есть является портативным, имеет возможность работы с USB накопителями, позволяет работать как на локальном, так и на сетевом, и на внешнем IP адресе, дает возможность легкого создания домена путем создания папки, Поддерживает псевдонимы, которые являются доменными указателями, что позволит задать доменный указатель для разрабатываемого сервера, IP адрес которого указывается в программном обеспечении умной банки, а так же содержит в себе планировщик заданий, который реализует сценарии, которые позволят осуществить полную автономную работу сервера по взаимодействию с аппаратной частью, то выбор был сделан именно в его пользу.

3.3 Детали реализации

В результате разработки были реализованы php-скрипты для обмена данными с устройством посредством GET-запросов. Ниже описана структура системы.

- activateLED.php – данный скрипт написан для реализации отправки POST-запроса, который посылает управляющий сигнал для включения световой сигнализации о необходимости принять медикаменты.
- readINFO.php – скрипт, разработанный для реализации логики отправки управляющего сигнала о необходимости отправить данные о приеме с умной банки на сервер и выключить световую сигнализацию.
- sendInfoToDB.php – этот скрипт был написан для реализации обмена данными со сторонним веб-приложением мед-учреждения, которое будет отслеживать и анализировать прием медикаментов, а так же для ведения журнала приема на стороне сервера.
- Controller.php – предназначен для ручного управления запросами сервера к умной банке.
- fullInfo.php – разработан для выведения журнала приема на сервере.
- shortInfo.php – необходим для отображения последнего статуса приема медикаментов.

Были реализованы Cron-сценарии, которые изображены на рисунке 3.2, для реализации алгоритма, организующего автономную работу сервера.

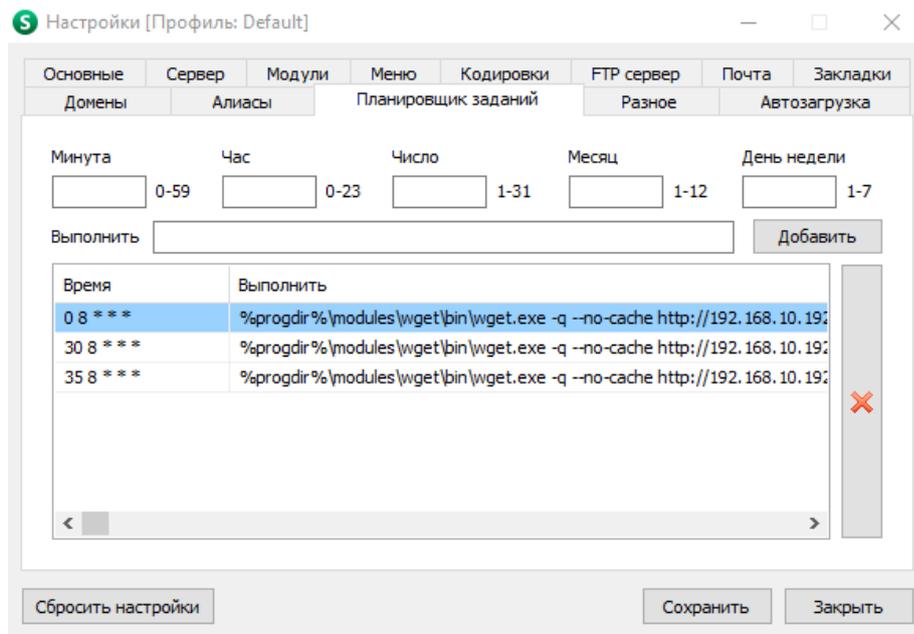


Рисунок 3.2 - Разработанные cron-сценарии

Алгоритм состоит из трех сценариев:

- Первый сценарий запускает на выполнение скрипт activateLED.php в специально назначенное время для включения световой сигнализации и чтения информации с датчика на стороне умной банки;

- Второй сценарий реализует исполнение скрипта readINFO.php в определенное время, которое реализует отправку управляющего сигнала на аппаратное обеспечение, которое на стороне умной банки приостанавливает считывание данных с датчика, выключает световую сигнализацию и формирует GET-запрос с данными о приеме медикаментов;

- Последний сценарий реализует отправку данных на веб-приложения для анализа учета врачом приема медикаментов пациентом, а также добавляет информацию в журнал приема и обновляет статус о последнем приеме медикаментов.

После написания скриптов для работы сервера, создания последовательности сценариев в планировщике задач для передачи управляющих сигналов на устройство, а также отладки работы сервера и

устройства умной банки получается сервис для обмена информацией с устройством и внешним веб-приложением. Экранная форма получившегося сервера представлена на рисунке 3.3

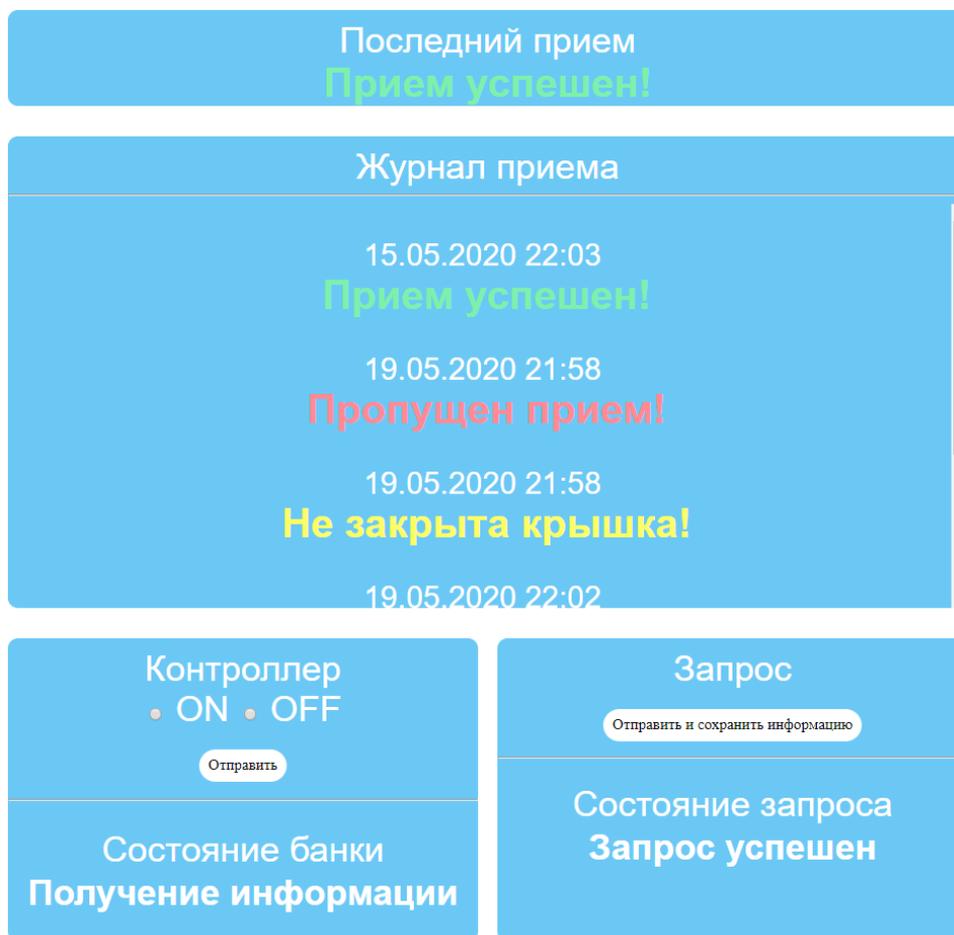


Рисунок 3.3 - Экранная форма сервера

ЗАКЛЮЧЕНИЕ

В результате проделанной работы были проанализированы аналоги систем «Умная банка». Были проанализированы типы плат, модели плат, разработанных на микроконтроллере. Были проанализированы доступные способы связи для плат на микроконтроллере. На основе этого анализа было произведено сравнение существующих способов реализации обмена данными с платой по выбранному способу связи. Так же были проанализированы существующие датчики магнитного поля для выбора наиболее подходящего в качестве сенсора, который отслеживает состояние разработанного прототипа. Было спроектировано схематичное представление работы прототипа устройства «умная банка». Была произведена сборка в соответствии со спроектированной схемой устройства умная банка, а также успешно протестирована работоспособность разработанного устройства. Было разработано программное обеспечение для реализации связи между собранным прототипом и веб-приложением. Данное ПО реализует набор сценариев для обеспечения автономной работы умной банки, а также реализована функциональность ручного управления. Сервер реализует отправку управляющего сигнала, который включает световую индикацию на прототипе, сигнализирующую о необходимости принятия медикаментов. В случае, когда крышка умной банки не закрыта включается дополнительная световая индикация, которая сообщает пользователю о необходимости закрыть крышку умной банки. После успешного приема вся световая индикация выключается. Далее сервер реализует отправку управляющего сигнала о необходимости отправить данные о приеме с умной банки на сервер. Когда прототип принимает сигнал, то световая индикация выключается в случае, если пользователь так и не закрыл крышку. Прототип формирует данные и отправляет их на сторону сервера. После произведенных действий сервер генерирует запрос

с данными для веб-приложения, а так же вносит данные в журнал приема и изменяет актуальный статус последнего приема медикаментов. Сервер имеет возможность отображать журнал приема медикаментов со статусом приема, датой и временем, статус последнего приема медикаментов пользователем, а так же элементы ручного управления.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Соблюдение режима приема лекарственных препаратов [Электронный ресурс]: Режим доступа: <https://www.msmanuals.com/ru/дома/лекарственные-препараты/факторы,-определяющие-ответ-организма-на-воздействие-лекарственных-препаратов/соблюдение-режима-приема-лекарственных-препаратов> (дата обращения: 15.03.2020).
2. Побочные эффекты лекарственных средств [Электронный ресурс]: Режим доступа: <https://cyberleninka.ru/article/n/pobochnye-effekty-lekarstvennyh-sredstv> (дата обращения: 21.03.2020).
3. Смертность от болезней системы кровообращения и продолжительность жизни в России [Электронный ресурс]: Режим доступа: <https://cyberleninka.ru/article/n/smertnost-ot-bolezney-sistemy-krovoobrascheniya-i-prodolzhitelnost-zhizni-v-rossii> (дата обращения: 14.04.2020).
4. Бактерии устойчивы к антибиотикам. Что делать? Не принимать антибиотики [Электронный ресурс]: Режим доступа: <https://www.bbc.com/russian/features-46065466> (Дата Обращения: 16.04.2020).
5. Вирт, Никлаус Построение компиляторов / Никлаус Вирт. - М.: ДМК Пресс, 2017. - 696 с.
6. About Nanhealth [Электронный ресурс]: Режим доступа: <https://nanthealth.com/about/> (Дата обращения 22.03.2020)
7. Our mission: to build software and devices that seamlessly connect patients to care. [Электронный ресурс]: Режим доступа: https://adheretech.com/company_ (Дата обращения 22.03.2020)

8. About us | Learn About Assisted Living Technologies Today [Электронный ресурс]: Режим доступа: <https://www.assistedlivingtechnologies.com/about-us> (Дата обращения 22.03.2020)
9. About us [Электронный ресурс]: Режим доступа: <https://www.medcentersystems.com/aboutus.asp> (Дата обращения 22.03.2020)
10. О нас | Xiaomi Россия | Mi.com [Электронный ресурс]: Режим доступа: <https://www.mi.com/ru/about/> (Дата обращения 22.03.2020)
11. В.А. Петин. Arduino и Raspberry Pi в проектах Internet of Things. – СПб.: БХВ-Петербург, 2016. – 320 с.
12. Д.И. Мамичев. Программирование на Ардуино. От простого к сложному. – М.: Солон-Пресс, 2018. – 244 с.
13. У.Соммер. Программирование микроконтроллерных плат Arduino/Freduino. – СПб.: БХВ-Петербург, 2012. – 256 с.
14. Iskra Neo [Электронный ресурс]: Режим доступа: <https://amperka.ru/product/iskra-neo> (Дата обращения 22.03.2020)
15. Ю.Ревич. Албука электроники. Изучаем Arduino. – М.: АСТ, Кладезь, 2017. – 224 с.
16. С.Монк. Програмуем Arduino. Основы работы со скетчами. – СПб.: Питер, 2016. – 176 с.
17. Джон Бокселл. Изучаем Arduino. 65 проектов своими руками. – СПб.: Питер, 2017. – 400 с.
18. В.А. Петин, А.А. Биняковский. Практическая энциклопедия Arduino. – М.: ДМК Пресс, 2017. – 152 с.
19. Iskra JS [Электронный ресурс]: Режим доступа: <https://amperka.ru/product/iskra-js> (Дата обращения 22.03.2020)
20. Strela [Электронный ресурс]: Режим доступа: <https://amperka.ru/product/strela> (Дата обращения 22.03.2020)

21. А.В. Белов. Arduino. От азов программирования до создания практических устройств. – М.: Наука и техника, 2018. – 480 с.
22. Netduino 2 [Электронный ресурс]: Режим доступа: <https://amperka.ru/product/netduino-2> (Дата обращения 22.03.2020)
23. Netduino 2 Plus [Электронный ресурс]: Режим доступа: <https://amperka.ru/product/netduino-plus-2> (Дата обращения 22.03.2020)
24. А. Бронштейн, В. Ривкин, И. Левин Частная медицина в России и за рубежом. - М.: Кворум, 2013. - 204 с.
25. Зелковиц, М. Принципы разработки программного обеспечения / М. Зелковиц, А. Шоу, Дж. Гэннон. - М.: Мир, 2018. - 364 с.
26. Т.Иго. Arduino, датчики и сети для связи устройств. – СПб.: БХВ-Петербург, 2015. – 544 с.
27. В.А. Петин. Создание умного дома на базе Arduino. – М.: ДМК Пресс, 2017. – 180 с.
28. Торо Карвинен, Киммо Карвинен, Вилле Валтокари. Делаем сенсоры. Проекты сенсорных устройств на базе Arduino и Raspberry Pi. – М.: Вильямс, 2015. – 448 с.
29. Ярнольд Стюарт. Arduino для начинающих. Самый простой пошаговый самоучитель. – М.: Эксмо-Пресс, 2017. – 256 с.
30. Джереми Блум. Изучаем Arduino. Инструменты и методы технического волшебства. – СПб.: БХВ-Петербург, 2015. – 336 с.
31. Чарльз Платт. Электроника. Логические микросхемы, усилители и датчики для начинающих. – М.: Торгово-издательское Бюро ВHV, 2018. – 448 с.
32. Датчик магнитного поля (геркон) модуль KY-021 для Ардуино. Обзор [Электронный ресурс]: Режим доступа: <http://funnydiy.net/1002> (Дата обращения 15.04.2020)

33. Цифровой датчик магнитного поля. Модуль KY-025 для Ардуино. Обзор [Электронный ресурс]: Режим доступа: <http://funnydiy.net/1011> (Дата обращения 15.04.2020)

34. Tinkercad – простой веб-инструмент для 3D-проектирования и 3D-печати [Электронный ресурс]: Режим доступа: <http://www.proghouse.ru/article-box/115-tinkercad> (Дата обращения 20.04.2020)

35. Кузнецов, М. Объектно-ориентированное программирование на PHP / М. Кузнецов, И. Симдянов. - М.: БХВ-Петербург, 2008. - 608 с.

36. Локхарт, Джош Современный PHP. Новые возможности и передовой опыт / Джош Локхарт. - М.: ДМК Пресс, 2015. - 208 с.

37. Встречайте: Open Server! [Электронный ресурс]: Режим доступа: <https://ospanel.io/> (Дата обращения 22.04.2020)

ПРИЛОЖЕНИЕ А

Исходный код устройства «Умная банка»

Листинг А.1 – Исходный код программного обеспечения умной банки на С подобном языке

```
#include <SPI.h>
#include <Ethernet.h>

byte mac[] = { 0x54, 0x34, 0x41, 0x30, 0x30, 0x31 };

EthernetClient client;
char server[] = "192.168.10.192";
int buff=0;
const int led=5;
const int errorLed=8;
const int openLed=7;
const int gercon=9;
int ledState = 0;
int gerconState = 0;
int gerconCount = 0;
int forRequest = 0;

void setup()
{
  Ethernet.begin(mac);
  pinMode(led, OUTPUT);
  pinMode(errorLed, OUTPUT);
  pinMode(openLed, OUTPUT);
  digitalWrite(led, LOW);
  digitalWrite(errorLed, LOW);
  digitalWrite(openLed, LOW);
  pinMode(gercon, INPUT);
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ А

```
Serial.begin(9600);

}

void loop()
{
  if (client.connect(server, 80)){
    digitalWrite(errorLed,LOW);
    sendRequest();
    Serial.print("data from server:");
    Serial.print(buff);
    Serial.print("\n cane status:");
    Serial.print(gerconCount);
    Serial.print("\n");
    while(client.available()){
      char c = client.read();
      if ( c=='1')
      {
        buff=1;
      }
      if ( c=='0')
      {
        buff=0;
      }
    }
    client.stop();
    client.flush();
    delay(100);
  }
  else
  {
    Serial.print("Error happened \n");
  }
}
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ А

```
if(digitalRead(errorLed == LOW)){
    digitalWrite (errorLed, HIGH);
}

client.stop();
delay(1000);
digitalWrite (errorLed, HIGH);
client.connect(server, 80);
}

if(buff==1){
    startListen();
}

if ( buff==0){
    digitalWrite(led, LOW);
    digitalWrite (openLed, LOW);
    ledState = 0;
    gerconCount = 0;
}

delay(500);
}

void startListen(){
    if(gerconCount!=2){
        digitalWrite (led, HIGH);
        ledState = 1;
        gerconState = !digitalRead(gercon);
        if(gerconState ==0 && gerconCount==0){
            gerconCount++;
        }
    }
}
```

ОКОНЧАНИЕ ПРИЛОЖЕНИЯ А

```
digitalWrite (openLed, HIGH);
}
if(gerconState==1 && gerconCount==1){
    gerconCount++;
    digitalWrite (openLed, LOW);
    digitalWrite (led, LOW);

}
forRequest = gerconCount;
}
}

void sendRequest(){
    client.print( "GET /add_data.php?");
    client.print("gercon=");
    if(buff!=0){
        client.print( "0" );
    }else{
        client.print(forRequest);
    }
    client.println( " HTTP/1.1");
    client.print( "Host: " );
    client.println(server);
    client.println( "Connection: close" );
    client.println();
    client.println();
    delay(200);
}
```

ПРИЛОЖЕНИЕ Б

Исходный код сервера «Умная банка»

Листинг Б.1 – Исходный код работы сервера для активации световой индикации

```
<?php
$S2      = $_POST[1];
$myFile2 = "../txt/out-1.txt";
$fh2 = fopen($myFile2, 'w') or die("can't open file");
fwrite($fh2, 1);
fclose($fh2);
echo $S2;
?>
```

Листинг Б.2 – Исходный код работы сервера для отправки данных с устройства «Умная банка»

```
<?php
$S2      = $_POST[0];
$myFile2 = "../txt/out-1.txt";
$fh2 = fopen($myFile2, 'w') or die("can't open file");
fwrite($fh2, 0);
fclose($fh2);
echo $S2;
?>
```

Листинг Б.3 – Исходный код работы сервера для обработки, сохранения и отправки во внешнее веб-приложение полученных с устройства «Умная банка» данных

```
<?php
$myFile = "../txt/in-2.txt";
$fh     = fopen($myFile, 'r');
$theData = fread($fh, filesize($myFile));
fclose($fh);
$request = 'https://webapplication820200604105620.azurewebsites.net/home/adbrecordtodb?id=1&status=';
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
$request .= $theData;
$request .= '&data=';
$date = date("d.m.Y");
$request .= $date;
$request .= '&time=';
$time = date('H:i');
$request .= $time;
$contents = @file_get_contents($request);

$response = "../txt/out-2.txt";
$fh3 = fopen($response, 'w') or die("can't open file");
if ($contents !== false) {
    fwrite($fh3, '1');
} else {
    fwrite($fh3, '0');
}
fclose($fh3);

//Save info local
$shortInfo = "../txt/in-2.txt";
$si = fopen($shortInfo, 'r');
$S3 = fread($si, filesize($shortInfo));
fclose($si);

$myFile4 = "../txt/in-4.txt";
$fh3 = fopen($myFile4, 'w') or die("can't open file");
$message = "<p class='off'><b>Пропущен прием!</b></p>";
if ($S3 == 1) {
    $message = "<p class='wait'><b>Не закрыта крышка!</b></p>";
    fwrite($fh3, $message);
}
if ($S3 == 2) {
    $message = "<p class='on'><b>Прием успешен!</b></p>";
    fwrite($fh3, $message);
}
if ($S3 == 0) {
    fwrite($fh3, $message);
}
fclose($fh3);
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
$myFile3 = "../txt/in-3.txt";
$fh4 = fopen($myFile3, 'a') or die("can't open file");
$info = date("d.m.Y H:i ");
$info .= $message;
fwrite($fh4, $info . "\r\n");
fclose($fh4);

header('Location: ../index.php');
exit;
?>
```

Листинг Б.4 – Исходный код работы сервера для обработки полученных данных

```
<?php
$S3      = $_GET['gercon'];
$myFile3 = "txt/in-2.txt";
$fh3 = fopen($myFile3, 'w') or die("can't open file");
fwrite($fh3, $S3);
fclose($fh3);

$myFile  = "txt/out-1.txt";
$fh      = fopen($myFile, 'r');
$theData = fread($fh, filesize($myFile));
fclose($fh);
echo $theData;
?>
```

Листинг Б.5 – Исходный код работы сервера для ручного управления устройством «Умная банка»

```
<?php
$S2      = $_POST['status'];
$myFile2 = "../txt/out-1.txt";
$fh2 = fopen($myFile2, 'w') or die("can't open file");
fwrite($fh2, $S2);
fclose($fh2);
echo $S2;
?>
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

Листинг Б.6 – Исходный код работы сервера для отображения журнала приема медикаментов

```
<?php
$myFile  = "../txt/in-3.txt";
$fh      = fopen($myFile, 'r');
$theData1 = fread($fh, filesize($myFile));
fclose($fh);
echo $theData1;
?>
```

Листинг Б.7 – Исходный код работы сервера для отображения состояния устройства «Умная банка»

```
<?php
$myFile  = "../txt/out-1.txt";
$fh      = fopen($myFile, 'r');
$theData = fread($fh, filesize($myFile));
fclose($fh);
if ($theData == 1) {
    echo " <p><b>Включение индикации</b></p>";
}

if ($theData == 0) {
    echo " <p><b>Получение информации</b></p>";
}
?>
```

Листинг Б.8 – Исходный код работы сервера для отображения состояния запроса во внешнее веб-приложение и записи в журнал приема полученной с устройства «Умная банка» информации

```
<?php
$myFile  = "../txt/out-2.txt";
$fh      = fopen($myFile, 'r');
$theData = fread($fh, filesize($myFile));
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
fclose($fh);
if ($theData == 1) {
    echo "<p><b>Запрос успешен</b></p>";
}

if ($theData == 0) {
    echo "<p><b>Ошибка отправки</b></p>";
}

?>
```

Листинг Б.9 – Исходный код работы сервера для отображения состояния последнего приема медикаментов пациентом

```
<?php
$myFile  = "../txt/in-4.txt";
$fh      = fopen($myFile, 'r');
$theData1 = fread($fh, filesize($myFile));
fclose($fh);
echo $theData1;
?>
```

Листинг Б.10 – Исходный код главной страницы для администрирования устройством «Умная банка».

```
<html>

<head>
    <title>HOME</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <script src="//code.jquery.com/jquery-1.11.0.min.js"></script>

    <link rel="stylesheet" type="text/css" href="style.css">
    <script>
        function show() {
            $.ajax({
                url: "transfer/shortInfo.php",
                cache: false,
                success: function(html) {
                    $("#content").html(html);
                }
            });
        }
    </script>
</head>
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
    }
  });
  $.ajax({
    url: "transfer/fullInfo.php",
    cache: false,
    success: function(html) {
      $("#content-1").html(html);
    }
  });
  $.ajax({
    url: "transfer/ledstate.php",
    cache: false,
    success: function(html) {
      $("#content-3").html(html);
    }
  });
  $.ajax({
    url: "transfer/requestState.php",
    cache: false,
    success: function(html) {
      $("#content-4").html(html);
    }
  });
}
$(document).ready(function() {
  show();
  setInterval('show()', 500);
});

function AjaxFormRequest(result_id, controller, url) {
  jQuery.ajax({
    url: url,
    type: "POST",
    dataType: "html",
    data: jQuery("#" + controller).serialize(),
  });
}
</script>

</head>

<body>
  <div class="r">
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
<p class="r1">Последний прием</p>
<div class="r2" style="display:inline-block;">
  <div class="r3" id="content"></div>
  <!--<div class="r3"> C<sup>o</sup></div>-->
</div>
</div>
<div class="r">
  <p class="r1">Журнал приема</p>
  <hr>
  <div class="r5">
    <div class="r4" id="content-1"></div>
  </div>
</div>
<div class="r">
  <div class="r1">
    <p class="r1">Контроллер</p>
    <div class="r2" style="font-size:35px">
      <form id="controller" action="" method="post">
        <label>
          <input type="radio" name="status" value="1"> ON </label>
        <label>
          <input type="radio" name="status" value="0"> OFF </label>
        <br>
        <input class="submitButton" type="submit" value="Отправить"
onclick="AjaxFormRequest('messegeResult', 'controller',
'transfer/controller.php')">
      </form>
      <hr>
      <p class="r1">Состояние банки</p>
      <div class="r2" style="font-size:35px">
        <div class="r3" id="content-3"></div>
      </div>
    </div>
  </div>
</div>
<div class="rr">
  <p class="r1">Запрос</p>
  <div class="r2" style="font-size:35px">
    <form method="post" action="./transfer/sendInfoToDB.php" />
    <input class="submitButton" type='submit' value='Отправить и сохранить
информацию' />
  </form>
  </div>
</div>
<hr>
```

ОКОНЧАНИЕ ПРИЛОЖЕНИЯ Б

```
<p class="r1">Состояние запроса</p>
<div class="r2" style="font-size:35px">
  <div class="r3" id="content-4"></div>
</div>
</div>
</body>

</html>
```