

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное автономное образовательное учреждение
высшего образования

**«Национальный исследовательский Нижегородский
государственный университет им. Н.И. Лобачевского» (ННГУ)**

Институт информационных технологий, математики и механики

Кафедра: Математическое моделирование и суперкомпьютерные
технологии

Направление подготовки: «Фундаментальная информатика и
информационные технологии»

Профиль подготовки: «Инженерия программного обеспечения»

ОТЧЕТ

по научно-исследовательской работе на тему:

**«Разработка системы медицинского интернета вещей
для умного дома»**

Выполнил:

студент группы 3823Б1ФИЗ

_____ / Сахаров А. В. /

(подпись)

Научный руководитель:

старший преподаватель

_____ / Карчков Д. А. /

(подпись)

Нижний Новгород

2025

Оглавление

ВВЕДЕНИЕ.....	3
1. ЦЕЛИ И ЗАДАЧИ ПРАКТИКИ	4
2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ	5
2.1. Обзор технологий IoT для мониторинга здоровья	5
2.2. Протокол Bluetooth Low Energy (BLE)	5
2.3. Методы сбора данных с АЦП на ESP32	6
3. ПРАКТИЧЕСКАЯ ЧАСТЬ.....	8
3.1. Архитектура системы	8
3.2. Разработка монитора артериального давления.....	9
3.3. Разработка центрального хаба	10
3.4. Инструменты анализа данных	11
4. РЕЗУЛЬТАТЫ РАБОТЫ.....	13
ЗАКЛЮЧЕНИЕ	15
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	16
Приложение	17
1. Исходный код счетчика измерений артериального давления	17
2. Исходный код хаба	17

ВВЕДЕНИЕ

В современном мире системы мониторинга здоровья человека приобретают всё большую актуальность. Развитие технологий Интернета вещей (IoT) и миниатюризация электронных компонентов позволяют создавать компактные, энергоэффективные устройства для непрерывного контроля физиологических показателей.

Данная работа посвящена разработке адаптируемой системы контроля над здоровьем человека на базе микроконтроллера ESP32. Система представляет собой распределённую архитектуру, состоящую из периферийных устройств-датчиков, центрального хаба для сбора данных и программных инструментов для анализа полученной информации.

Основной особенностью разрабатываемой системы является её модульность и расширяемость: к центральному хабу могут подключаться различные датчики для измерения разных физиологических показателей (артериальное давление, пульс, температура тела и др.). Передача данных осуществляется по беспроводному протоколу Bluetooth Low Energy (BLE), что обеспечивает низкое энергопотребление и достаточную дальность связи.

В рамках практики был реализован первый модуль системы — монитор артериального давления, а также центральный хаб для приёма и хранения данных.

1. ЦЕЛИ И ЗАДАЧИ ПРАКТИКИ

Цель практики: разработка прототипа адаптируемой системы мониторинга здоровья человека на базе микроконтроллера ESP32 с использованием беспроводной передачи данных по протоколу BLE.

Задачи практики:

1. Изучить современные методы мониторинга физиологических показателей человека и существующие IoT-решения в области здравоохранения.
2. Исследовать принципы работы датчиков давления и методы обработки аналоговых сигналов на микроконтроллере ESP32.
3. Изучить протокол Bluetooth Low Energy (BLE) и его применение для передачи данных между устройствами.
4. Спроектировать архитектуру распределённой системы «датчик → хаб → хранилище».
5. Разработать периферийное устройство мониторинга артериального давления с передачей данных по BLE.
6. Реализовать центральный хаб для приёма данных от нескольких устройств и сохранения на SD-карту.
7. Создать программные инструменты для анализа и визуализации полученных данных.
8. Провести тестирование системы и документирование результатов.

2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

2.1. Обзор технологий IoT для мониторинга здоровья

Интернет вещей (IoT) в сфере здравоохранения представляет собой сеть взаимосвязанных устройств, которые собирают, передают и анализируют данные о состоянии здоровья человека. Современные носимые устройства (wearables) позволяют осуществлять непрерывный мониторинг таких показателей, как частота сердечных сокращений, артериальное давление, уровень кислорода в крови, температура тела и физическая активность.

Микроконтроллер ESP32 является одним из наиболее популярных решений для создания IoT-устройств благодаря следующим преимуществам:

- Встроенные модули Wi-Fi и Bluetooth (включая BLE);
- Двухъядерный процессор с частотой до 240 МГц;
- Низкое энергопотребление с поддержкой режимов глубокого сна;
- Богатый набор периферии (АЦП, ЦАП, SPI, I2C, UART);
- Доступная цена и широкая поддержка сообщества разработчиков.

Для измерения артериального давления в данном проекте используется датчик давления XGZP6847A — пьезорезистивный датчик с аналоговым выходом, обеспечивающий линейную зависимость выходного напряжения от приложенного давления.

2.2. Протокол Bluetooth Low Energy (BLE)

Bluetooth Low Energy (BLE) — это технология беспроводной связи, разработанная для приложений с низким энергопотреблением. В отличие от классического Bluetooth, BLE оптимизирован для передачи небольших объёмов данных с минимальным расходом энергии.

Основные концепции BLE, используемые в проекте:

GATT (Generic Attribute Profile) — определяет способ обмена данными между устройствами. Данные организованы в виде иерархии: профили содержат сервисы, сервисы содержат характеристики.

Сервер и клиент — в архитектуре BLE устройство-датчик выступает в роли GATT-сервера, предоставляющего данные, а хаб — в роли GATT-клиента, запрашивающего и получающего данные.

Уведомления (Notifications) — механизм, позволяющий серверу асинхронно отправлять данные клиенту без явного запроса. Это обеспечивает эффективную передачу потоковых данных.

В разработанной системе используются следующие UUID:
- Service UUID: 4fafc201-1fb5-459e-8fcc-c5c9c331914b
- Characteristic UUID: beb5483e-36e1-4688-b7f5-ea07361b26a8

2.3. Методы сбора данных с АЦП на ESP32

ESP32 оснащён двумя 12-битными АЦП (SAR ADC), которые могут использоваться для оцифровки аналоговых сигналов. Существует несколько методов работы с АЦП:

Прямое чтение (analogRead) — простейший метод, при котором процессор блокируется на время преобразования. Подходит для низкочастотных измерений.

I2S DMA — продвинутый метод, использующий периферию I2S в режиме АЦП с прямым доступом к памяти (DMA). Позволяет осуществлять высокочастотную оцифровку без нагрузки на процессор.

В данном проекте используется метод I2S DMA со следующими параметрами:

- Частота дискретизации: 1000 Гц (1000 выборок в секунду);

- Разрядность: 12 бит (диапазон значений 0-4095);
- Размер DMA-буфера: 256 выборок;
- Количество буферов: 4.

Такая конфигурация обеспечивает непрерывный сбор данных с датчика давления с последующим усреднением за 1 секунду для получения стабильных показаний.

3. ПРАКТИЧЕСКАЯ ЧАСТЬ

3.1. Архитектура системы

Разработанная система имеет трёхуровневую архитектуру:

Уровень 1 — Периферийные устройства (датчики): Устройства на базе ESP32, оснащённые различными датчиками для измерения физиологических показателей. Каждое устройство работает автономно, собирает данные и передаёт их по BLE. В текущей реализации создан монитор артериального давления.

Уровень 2 — Центральный хаб: Устройство на базе ESP32, выполняющее роль BLE-клиента. Хаб сканирует эфир, обнаруживает периферийные устройства по UUID сервиса или имени, подключается к ним и принимает данные. Полученные данные сохраняются на SD-карту для последующего анализа.

Уровень 3 — Программные инструменты: Python-скрипты для чтения данных с хаба через Serial-порт, анализа и визуализации данных, экспорта в формат Excel.

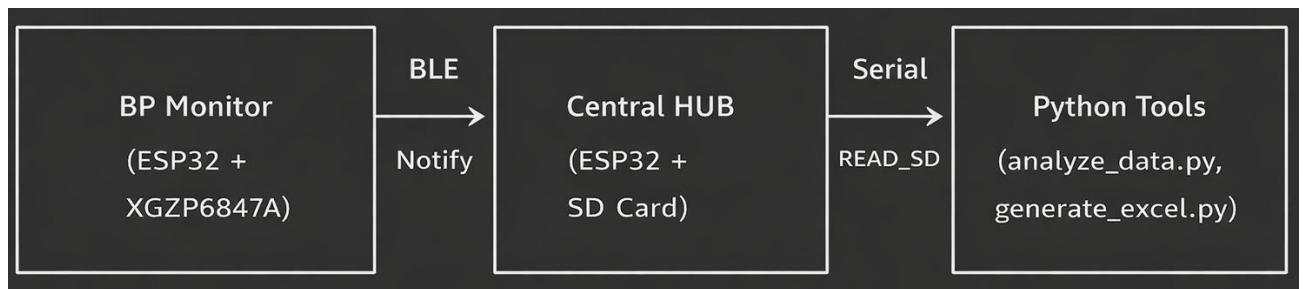


Рис. 1 — Схема взаимодействия компонентов

3.2. Разработка монитора артериального давления

Монитор артериального давления представляет собой устройство на базе ESP32 с подключённым датчиком давления XGZP6847A. Устройство автоматически определяет момент начала измерения, записывает данные и передаёт их на хаб по BLE.

Аппаратная конфигурация:

- Микроконтроллер: ESP32 DevKit
- Датчик давления: XGZP6847A (подключён к GPIO 34)
- Питание: 3.3В

Алгоритм работы устройства:

1. Инициализация: настройка BLE-сервера, I2S DMA для АЦП, запуск рекламы (advertising).
2. Мониторинг: непрерывное чтение данных с датчика, усреднение за 1 секунду, отслеживание максимального давления.
3. Определение начала измерения: когда давление превышает порог (70 мм рт.ст.) и начинает падать более чем на 15 мм рт.ст., запускается 4-секундный период проверки.
4. Подтверждение падения: если в течение 4 секунд давление продолжает падать, начинается запись данных в буфер.
5. Запись данных: значения давления сохраняются в вектор pressureHistory с частотой 1 Гц.
6. Завершение измерения: когда давление падает ниже 40 мм рт.ст., запись прекращается и данные передаются по BLE.

Протокол передачи данных по BLE:

Данные передаются в виде последовательности уведомлений (notifications):

1. Маркер начала: "START"
2. Данные: "D:XXX.XX" (где XXX.XX — значение давления в мм рт.ст.)
3. Маркер конца: "END"

Пример передаваемых данных:

START

D:120.55

D:118.30

D:115.20

...

D:42.10

END

Ключевые параметры конфигурации:

- SENSOR_KOEF = 0.14 — коэффициент преобразования АЦП в мм рт.ст.
- SENSOR_ZERO_OFFSET = 271 — смещение нуля датчика
- PRESSURE_THRESHOLD = 70.0 — порог начала измерения (мм рт.ст.)
- PRESSURE_DROP_THRESHOLD = 15.0 — минимальное падение для фиксации

3.3. Разработка центрального хаба

Центральный хаб выполняет функции сбора данных от периферийных устройств и их хранения. Устройство построено на базе ESP32 с подключённым модулем SD-карты.

Аппаратная конфигурация:

- Микроконтроллер: ESP32 DevKit
- SD-карта: подключена по SPI (CS=GPIO5, MOSI=GPIO23, MISO=GPIO19, SCK=GPIO18)

Алгоритм работы хаба:

1. Инициализация: настройка SD-карты, инициализация BLE-клиента, запуск сканирования.
2. Поиск устройств: сканирование BLE-эфира, поиск устройств по UUID сервиса или имени "ESP32_BP_Monitor".
3. Подключение: при обнаружении устройства хаб подключается к нему и регистрируется на получение уведомлений.
4. Приём данных: при получении уведомления данные парсятся и сохраняются в вектор receivedData.
5. Сохранение: по завершении передачи (маркер "END") данные записываются в файл /received_data.txt на SD-карте.
6. Интерфейс чтения: по команде "READ_SD" через Serial-порт хаб отправляет содержимое файла с маркерами ---START_FILE--- и ---END_FILE---.

Обработка уведомлений реализована в callback-функции notifyCallback, которая анализирует полученные данные и выполняет соответствующие действия в зависимости от типа сообщения (START, END или данные).

3.4. Инструменты анализа данных

Для работы с полученными данными разработаны два Python-скрипта:

1. generate_excel.py — скрипт для чтения данных с хаба и экспорта в Excel:
 - Автоматическое определение COM-порта ESP32;
 - Отправка команды READ_SD для получения данных;
 - Парсинг полученных значений;
 - Создание Excel-файла с таблицей данных и графиком.

Скрипт использует библиотеки: pandas, xlsxwriter, pyserial.

2. analyze_data.py — скрипт для анализа и визуализации данных измерений:

- Чтение CSV-файлов с данными измерений;
- Расчёт статистических показателей (среднее, минимум, максимум);
- Построение графика давления с отметкой пикового значения;
- Сохранение графика в PNG-файл.

Скрипт использует библиотеки: pandas, matplotlib.

Пример использования:

```
python generate_excel.py # Интерактивный выбор COM-порта  
python analyze_data.py bp_data.txt # Анализ файла измерений
```

4. РЕЗУЛЬТАТЫ РАБОТЫ

В ходе выполнения практики были достигнуты следующие результаты:

1. Изучены технологии IoT для мониторинга здоровья, протокол BLE и методы работы с АЦП на ESP32.
2. Спроектирована трёхуровневая архитектура системы мониторинга здоровья с возможностью расширения.
3. Разработан монитор артериального давления на базе ESP32:
 - Реализован сбор данных с датчика XGZP6847A через I2S DMA;
 - Реализован алгоритм автоматического определения начала и конца измерения;
 - Реализована передача данных по BLE с использованием механизма уведомлений.
4. Разработан центральный хаб на базе ESP32:
 - Реализовано сканирование и подключение к BLE-устройствам;
 - Реализован приём и парсинг данных от периферийных устройств;
 - Реализовано сохранение данных на SD-карту;
 - Реализован интерфейс чтения данных через Serial-порт.
5. Созданы программные инструменты на Python:
 - Скрипт для экспорта данных в Excel с построением графиков;

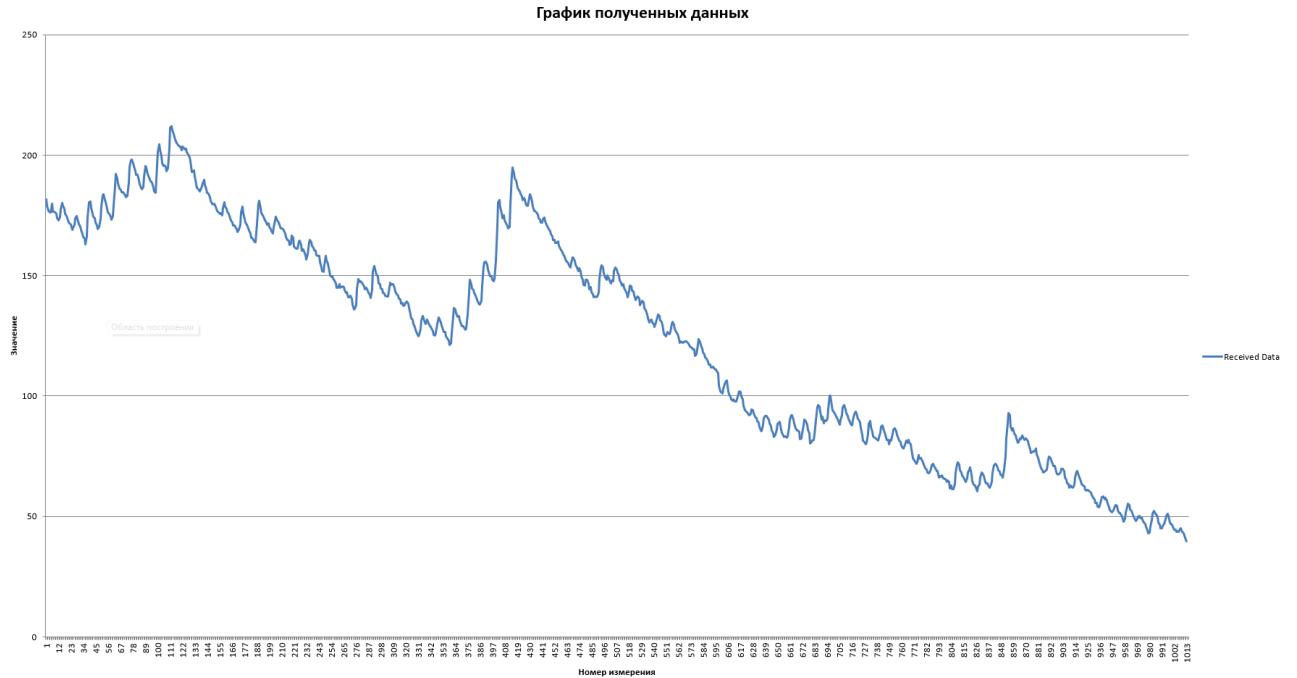


Рис. 2 — График показаний счетчика давления

6. Проведено тестирование системы, подтвердившее корректность работы всех компонентов.

Система успешно демонстрирует возможность беспроводного сбора данных о физиологических показателях человека с последующим хранением и анализом.

ЗАКЛЮЧЕНИЕ

В ходе проектно-технологической практики была разработана адаптируемая система контроля над здоровьем человека на базе микроконтроллера ESP32. Система представляет собой модульную архитектуру, позволяющую подключать различные датчики для мониторинга физиологических показателей.

Основные достижения практики:

- Освоены технологии разработки IoT-устройств на платформе ESP32;
- Изучен и применён протокол Bluetooth Low Energy для беспроводной передачи данных;
- Получен опыт работы с аналого-цифровым преобразованием и DMA;
- Разработан рабочий прототип системы мониторинга артериального давления;
- Созданы инструменты для анализа и визуализации медицинских данных.

Разработанная система имеет потенциал для дальнейшего развития:

- Добавление новых типов датчиков (пульсоксиметр, термометр, ЭКГ);
- Реализация серверной части для централизованного хранения данных;
- Разработка веб-приложения для отображения данных и рекомендаций;
- Интеграция с системами искусственного интеллекта для анализа состояния здоровья.

Все поставленные задачи практики выполнены в полном объёме. Полученные знания и навыки будут использованы для дальнейшего развития проекта.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ESP32 Technical Reference Manual. — Espressif Systems, 2024.
2. Bluetooth Core Specification v5.3. — Bluetooth SIG, 2023.
3. XGZP6847A Pressure Sensor Module Datasheet. — CFSensor, 2020.
4. . Kolban N. Kolban's Book on ESP32. — 2023.
5. Официальная документация PlatformIO. — URL:
<https://docs.platformio.org/>
6. Официальная документация Arduino-ESP32. — URL:
<https://docs.espressif.com/projects/arduino-esp32/>
7. Документация библиотеки pandas. — URL: <https://pandas.pydata.org/docs/>

Приложение

1. Исходный код счетчика измерений артериального давления

Код находится на репозитории github Esp32_HealthSaver по ветке Esp32_blood_pressure_monitor:

https://github.com/AleksndrSakharov/Esp32_HealthSaver/tree/Esp32_blood_pressure_monitor/blood_pressure_monitor

2. Исходный код хаба

Код находится на репозитории github Esp32_HealthSaver по ветке Esp32_hub:

https://github.com/AleksndrSakharov/Esp32_HealthSaver/tree/Esp32_hub/Esp32_hub