

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Инженерная школа информационных технологий и робототехники  
Направление подготовки 15.03.04 «Автоматизация технологических процессов и производств»  
Отделение автоматизации и робототехники

**БАКАЛАВРСКАЯ РАБОТА**

Тема работы
<b>Система дистанционного изменения параметров управления электрического двигателя с применением веб интерфейса</b>

УДК 004.896:004.777.621.313.13

Студент

Группа	ФИО	Подпись	Дата
8Т4Б	Звонарев Александр Андреевич		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Фадеев Александр Сергеевич	к.т.н.		

**КОНСУЛЬТАНТЫ:**

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ст. преподаватель ОСГН	Хаперская Алена Васильевна			

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ИШХБМТ	Невский Егор Сергеевич			

**ДОПУСТИТЬ К ЗАЩИТЕ:**

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР	Громаков Евгений Иванович	к.т.н.		

Томск – 2018 г.

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код результата	Результат обучения (Выпускник должен быть готов)
<i><b>Профессиональные компетенции</b></i>	
P1	Демонстрировать базовые естественнонаучные и математические знания для решения научных и инженерных задач в области анализа, синтеза, проектирования, производства и эксплуатации систем автоматизации технологических процессов и производств. Уметь сочетать теорию, практику и методы для решения инженерных задач, и понимать область их применения.
P2	Иметь осведомленность о передовом отечественном и зарубежном опыте в области теории, проектирования, производства и эксплуатации систем автоматизации технологических процессов и производств.
P3	Применять полученные знания для определения, формулирования и решения инженерных задач при разработке, производстве и эксплуатации современных систем автоматизации технологических процессов и производств с использованием передовых научно-технических знаний и достижений мирового уровня, современных инструментальных и программных средств.
P4	Уметь выбирать и применять соответствующие аналитические методы и методы проектирования систем автоматизации технологических процессов и обосновывать экономическую целесообразность решений.
P5	Уметь находить необходимую литературу, базы данных и другие источники информации для автоматизации технологических процессов и производств.
P6	Уметь планировать и проводить эксперимент, интерпретировать данные и их использовать для ведения инновационной инженерной деятельности в области автоматизации технологических процессов и производств.
P7	Уметь выбирать и использовать подходящее программно-техническое оборудование, оснащение и инструменты для решения задач автоматизации технологических процессов и производств.
<i><b>Универсальные компетенции</b></i>	
P8	Владеть иностранным языком на уровне, позволяющем работать в интернациональной среде с пониманием культурных, языковых и социально-экономических различий.
P9	Эффективно работать индивидуально, в качестве члена и руководителя группы с ответственностью за риски и работу коллектива при решении инновационных инженерных задач в области автоматизации технологических процессов и производств, демонстрировать при этом готовность следовать профессиональной этике и нормам.
P10	Иметь широкую эрудицию, в том числе знание и понимание современных общественных и политических проблем, вопросов безопасности и охраны здоровья сотрудников, юридических аспектов, ответственности за инженерную деятельность, влияния инженерных решений на социальный контекст и окружающую среду.
P11	Понимать необходимость и уметь самостоятельно учиться и повышать квалификацию в течение всего периода профессиональной деятельности.

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Школа Инженерная школа информационных технологий и робототехники  
Направление подготовки 15.03.04 «Автоматизация технологических процессов и производств»  
Отделение школы (НОЦ) Отделение автоматизации и робототехники

УТВЕРЖДАЮ:  
Руководитель ООП  
\_\_\_\_\_ Громаков Е. И.  
(Подпись) (Дата) (Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

бакалаврской работы
---------------------

Студенту:

Группа	ФИО
8Т4Б	Звонареву Александру Андреевичу

Тема работы:

Система дистанционного изменения параметров управления электрического двигателя с применением веб интерфейса	
Утверждена приказом директора (дата, номер)	№2183/с от 28.03.2018 г.

Срок сдачи студентом выполненной работы:	
--	--

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<b>Исходные данные к работе</b>	<ul style="list-style-type: none"><li>– Техническая документация по Wi-Fi модулю ESP12-E на базе микроконтроллера ESP8266</li><li>– Техническая документация по микроконтроллеру STM32F103C8T6</li><li>– Интернет-ресурсы, включающие описание технологии веб и UART</li><li>– Интернет-ресурсы по проектированию и разработке устройств в программах ES-Plorer, STM32CubeMX, Keil uVision</li></ul>
---------------------------------	--

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p>	<p>Компоненты системы дистанционного изменения технологических параметров</p> <ul style="list-style-type: none"> <li>• Оборудование для реализации системы</li> <li>• Программное обеспечение</li> <li>• Протоколы обмена данными</li> </ul> <p>Реализация системы дистанционного изменения технологических параметров</p> <ul style="list-style-type: none"> <li>• Создание структурной схемы</li> <li>• Реализация веб сервера</li> <li>• Создание HTML страницы</li> <li>• Реализация обмена данными по UART протоколу</li> </ul>
<p><b>Перечень графического материал</b></p>	<p>Презентация в формате *.ppt</p>
<p><b>Консультанты по разделам выпускной квалификационной работы</b></p>	
<p><b>Раздел</b></p>	<p><b>Консультант</b></p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Хаперская Алена Васильевна</p>
<p>Социальная ответственность</p>	<p>Невский Егор Сергеевич</p>
<p><b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b></p>	
<p style="text-align: center;">—</p>	

<p><b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b></p>	
--	--

**Задание выдал руководитель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Фадеев Александр Сергеевич	к.т.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8Т4Б	Звонарев Александр Андреевич		

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Инженерная школа информационных технологий и робототехники  
Направление подготовки 15.03.04 «Автоматизация технологических процессов и производств»  
Уровень образования Бакалавриат  
Отделение автоматизации и робототехники  
Период выполнения осенний / весенний семестр 2017/2018 учебного года

Форма представления работы:

бакалаврская работа

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН**  
**выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
14.05.2018	Основная часть	75
20.05.2018	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
28.05.2018	Социальная ответственность	10

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Фадеев Александр Сергеевич	К.Т.Н.		

**СОГЛАСОВАНО:**

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР	Громаков Евгений Иванович	К.Т.Н.		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕ-  
СУРСОСБЕРЕЖЕНИЕ»**

Студенту:

<b>Группа</b>	<b>ФИО</b>
8Т4Б	Звонареву Александру Андреевичу

<b>Школа</b>	<b>Инженерная школа информационных технологий и робототехники</b>	<b>Отделение</b>	<b>Автоматизации и робототехники</b>
<b>Уровень образования</b>	Бакалавр	<b>Направление/специальность</b>	15.03.04 «Автоматизация технологических процессов и производств»

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	<i>Оклады ТПУ, стоимость оборудования</i>
2. <i>Нормы и нормативы расходования ресурсов</i>	<i>Расчетно-аналитические</i>
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	<i>Общая система налогообложения с учетом льгот для учебных учреждений (в том числе 27.1% отчислений во внебюджетные фонды)</i>

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	<ol style="list-style-type: none"> <li>1. <i>Потенциальные потребители результатов исследования</i></li> <li>2. <i>Анализ конкурентных решений</i></li> <li>3. <i>Технология QuaD</i></li> <li>4. <i>Оценки перспективности проекта по технологии SWOT</i></li> <li>5. <i>Определение возможных альтернатив проведения научных исследований</i></li> </ol>
2. <i>Планирование и формирование бюджета научных исследований</i>	<ol style="list-style-type: none"> <li>1. <i>Расчет материальных затрат НИИ.</i></li> <li>2. <i>Расчет затрат на специальное оборудование для научных (экспериментальных) работ.</i></li> <li>3. <i>Основная заработная плата исполнителей темы.</i></li> <li>4. <i>Дополнительная заработная плата исполнителей темы.</i></li> <li>5. <i>Отчисление во внебюджетные фонды</i></li> <li>6. <i>Прочие расходы.</i></li> </ol>
1. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	<i>Анализ интегральных показателей эффективности.</i>

**Перечень графического материала**

1. *Оценка конкурентоспособности технических решений*
2. *Оценочная карта сравнения конкурентных решений*
3. *Матрица SWOT*
4. *Диаграмма Ганта*
5. *Материальные затраты*
6. *Бюджет затрат по каждому исполнителю НИИ*
7. *Оценка ресурсной, финансовой и экономической эффективности НИ*

Дата выдачи задания для раздела по линейному графику	
--	--

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ст. преподаватель ОСГН	Хаперская Алена Васильевна			

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8Т4Б	Звонарев Александр Андреевич		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

<b>Группа</b>	<b>ФИО</b>
8Т4Б	Звонареву Александру Андреевичу

<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение</b>	<b>Автоматизации и робототехники</b>
<b>Уровень образования</b>	Бакалавр	<b>Направление/специальность</b>	15.03.04 Автоматизация технологических процессов и производств

Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	«Система дистанционного изменения параметров управления электрического двигателя с применением веб интерфейса» Объектом исследовательской работы является система дистанционного изменения технологических параметров объекта управления (электрического двигателя)
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
– Состав системы	В данном разделе указывается то, что используется в разработанной системе.
– Принцип работы системы	Описывается о том, как взаимодействуют все элементы системы.
– Методы защиты данных	Данный раздел сообщает о том, какие факторы влияют на повреждение информации в ходе передачи, о методах защиты от повреждения. Также рассказывается о защите системы от несанкционированного доступа.
– Факторы влияющие на человека	В данном разделе сообщается о факторах, возможно имеющих влияние на человека.
– Удобство в использовании	Анализ системы с точки зрения удобства использования.
– Надежность системы	Производится анализ надежности разработанной части системы.
– Возможности развития системы	В данном разделе изложены идеи развития системы с целью расширения функционала и улучшения удобства в использовании.

<b>Дата выдачи задания для раздела по линейному графику</b>	
---	--

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Ассистент ИШХБМТ	Невский Е.С.			

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8Т4Б	Звонарев Александр Андреевич		



## Реферат

Пояснительная записка содержит 75 страниц, 36 рисунков, 16 таблиц, 22 источника и 1 приложение.

Ключевые слова: дистанционное изменение технологических параметров, веб интерфейс, веб сервер, веб страница, Wi-Fi модуль, STM32.

Данная работа посвящена разработке системы дистанционного изменения технологических параметров с использованием веб интерфейса и html страницы удовлетворяющих поставленной задаче.

Объектом исследования является разработка системы на базе микроконтроллеров STM32 и ESP8266.

Цель работы – разработка системы дистанционного изменения технологических параметров с применением веб интерфейса и реализация на базе микроконтроллеров STM32 и ESP8266.

Результатом исследования является рабочий прототип системы.

Значимость работы заключается в создании системы, которая упростит наблюдение за технологическим процессом за счет отображения необходимых данных на экране на экранах смартфона (либо ноутбука) подсоединённых к сети Wi-Fi предприятия.

В будущем планируется модификация системы: реализация отображения информации о действующих значениях технологических параметров, оптимизация работы кода.

## Обозначения и сокращения

Веб сервер — сервер, принимающий HTTP-запросы от клиентов, обычно веб-браузеров, и выдающий им HTTP-ответы, как правило, вместе с HTML-страницей, изображением, файлом, медиа-поток или другими данными [1].

HTTP (англ. HyperText Transfer Protocol — «протокол передачи гипертекста») — сетевой протокол прикладного уровня передачи данных [2].

HTML (англ. HyperText Markup Language) — язык гипертекстовой разметки страниц, представляет собой набор символов разметки и кодов.

Lua — скриптовый язык программирования. Он сочетает в себе простой процедурный синтаксис с гибкими конструкциями описания данных, основанных на ассоциативных массивах и расширяемой семантике. Lua является динамически типизированным и интерпретируемым [3].

Язык Си (англ. C) — компилируемый статически типизированный язык программирования общего назначения, разработанный в 1969—1973 годах сотрудником Bell Labs Деннисом Ритчи как развитие языка Би [4].

## Оглавление

Введение .....	13
1. Проектирование системы дистанционного изменения параметров с применением веб интерфейса .....	15
1.1. Разработка структурной схемы .....	15
1.2. Компоненты системы .....	15
1.2.1. Микроконтроллер STM32 .....	16
1.2.2. WiFi модуль ESP12-E .....	18
1.2.3. JTAG – программатор .....	18
1.2.4. TTL-USB преобразователь.....	19
1.3. Выбор программного обеспечения .....	20
1.3.1. Программная среда Keil uVision для микроконтроллеров.....	20
1.3.2. Визуальный редактор настроек STM32CubeMX.....	22
1.3.3. Терминальное приложение Terminal v1.9b by Br@y++..	24
1.3.4. Терминальное приложения для микроконтроллера ESP8266 ESPlorer V0.2.0.....	25
1.3.5. ESP8266Flasher (NodeMCU flasher) .....	26
1.4. Используемые протоколы передачи данных .....	26
1.4.1. Протокол передачи данных UART .....	26
1.4.2. Технология Wi-Fi.....	27
1.5. Заключение по разделу 1.....	29
2. Техническая реализация системы дистанционного управления.	30
2.1. Подключение оборудования.....	30
2.1.1. Запуск модуля ESP12-E.....	30
2.1.2. Запуск микроконтроллера STM32F103 .....	31
2.1.3. Подключение TTL-USB преобразователя.....	32
2.2. Проверка оборудования .....	32
2.3. Реализация обмена данных между ESP и STM32 .....	35
2.4. Установка прошивки NodeMCU .....	37
2.5. Реализация веб сервера на модуле ESP12-E .....	38
2.6. Обмен данными с STM32 по UART .....	42
2.7. Заключение по разделу 2.....	43
3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....	45
3.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения.....	45
3.1.1. Потенциальные потребители продукта .....	45
3.1.2. Анализ конкурентных технических решений.....	46
3.1.3. Технология QuaD .....	48
3.1.4. SWOT-анализ .....	49
3.2. Определение возможных альтернатив проведения исследований .....	50
3.3. Планирование научно-исследовательских работ .....	51

3.3.1.	Структура работ в рамках научного исследования.....	51
3.3.2.	Определение трудоемкости выполнения работ.....	52
3.3.3.	Разработка графика проведения научного исследования .....	53
3.4.	Бюджет научно-технического исследования .....	56
3.4.1.	Основная заработная плата исполнителей темы.....	56
3.4.2.	Дополнительная заработная плата исполнительской.....	57
3.4.3.	Отчисление во внебюджетные фонды (страховые отчисления) .....	58
3.4.4.	Прочие расходы .....	59
3.4.5.	Формирование бюджета затрат научно-исследовательского проекта.....	60
3.4.6.	Определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования.....	61
3.5.	Заключение по разделу 3.....	63
4.	Социальная ответственность.....	64
4.1.	Аннотация.....	64
4.2.	Введение .....	64
4.3.	Состав системы .....	64
4.4.	Принцип работы системы .....	66
4.5.	Методы защиты данных.....	67
4.5.1.	Wi-Fi.....	67
4.5.2.	UART.....	68
4.6.	Факторы, влияющие на человека .....	68
4.7.	Удобство в использовании.....	69
4.8.	Надежность системы .....	69
4.9.	Возможности развития системы .....	69
4.10.	Заключение по разделу 4.....	70
	Заключение .....	71
	Список литературы .....	72
	Приложение А. Обозначение выводов модуля ESP12-E .....	75

## Введение

В настоящее время велика роль систем автоматизированного управления технологическими процессами в производстве. В некоторых случаях человека исключают из системы управления технологическим процессом, в то время как за управлением параметрами процесса отвечает система автоматического управления. Большинство систем автоматического управления используют микроконтроллеры в качестве устройства сбора данных о процессе, контроле состояний внутренних переменных системы и выдаче управляющих воздействий.

Задачи микроконтроллеров состоят в следующем: управление и регулирование; сбор и обработка технологических параметров; отображение и контроль данных; сигнализация и защита от аварийных ситуаций [5].

Особо можно выделить роль контроллеров в управлении электрическими двигателями. Для оператора основные задачи в управлении электродвигателем это: изменение параметров закона изменения скорости вращения вала двигателя, изменение уставки скорости, получение действительной информации о состоянии двигателя.

Для оперативного изменения параметров (устава, закон регулирования и т.д.), наблюдения и контроля текущих параметров используется связь с оператором (например, при помощи SCADA системы). Как правило, для реализации таких связей используются сложные и мало распространенные в Интернете протоколы, что приводит к усложнению сопряжения систем на базе микроконтроллеров со стандартными Интернет-решениями, использующимися на персональных компьютерах и смартфонах. Наиболее популярными для большинства современных веб-приложений являются веб-технологии. На базе веб-технологий возможно также реализовать связь «оператор-контроллер», тогда любому пользователю, обладающему современным устройством, имеющим выход в Интернет (например, с использованием модуля связи Wi-Fi), можно удаленно изменять значения технологического параметра и следить за этими параметрами из окна Интернет-браузера.

Целью работы является разработка системы дистанционного изменения параметров технологического процесса с применением веб-интерфейса.

Для достижения поставленной цели были сформулированы следующие задачи:

- разработать структурную схему системы дистанционного изменения технологических параметров;
- подобрать программные среды для разработки программного кода;
- смонтировать и отладить систему;
- изучить веб технологию и реализовать веб-сервер;
- обеспечить возможность обмена данными между элементами системы.

# 1. Проектирование системы дистанционного изменения параметров с применением веб интерфейса

## 1.1. Разработка структурной схемы

В ходе работы была спроектирована структурная схема, показанная на рисунке 1

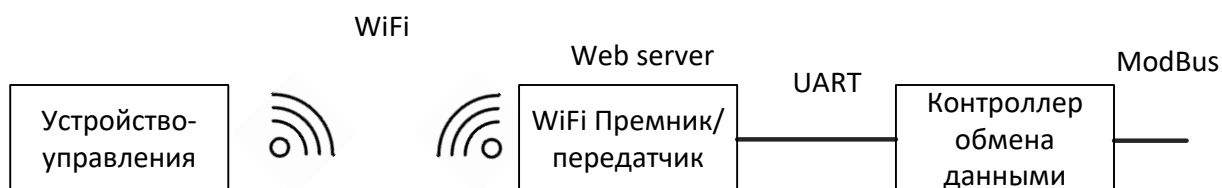


Рисунок 1 – Структурная схема системы дистанционного изменения технологических параметров

Принцип работы: устройство управления оператора (смартфон или ноутбук с установленным интернет браузером и Wi-Fi) подключается к локальной сети и формирует запрос к веб-серверу, где расположена HTML-страница. На данной странице расположена форма, в которую можно занести установочное значение скорости вращения вала двигателя. При нажатии кнопки на форме, это значение методом GET передается на веб-сервер, с сервера отсылается в контроллер обмена данными, а оттуда по протоколу ModBus передается далее в систему управления электрическим двигателем (например частотный регулятор на микроконтроллере).

Аналогично данные о скорости вращения вала двигателя приходят на контроллер обмена данными, оттуда передаются на веб сервер, с сервера – на веб страницу, а затем отображаются на устройстве управления оператора.

## 1.2. Компоненты системы

После разработки структурной схемы системы начался этап подбора оборудования. Оборудование выбиралось исходя из его наличия. В качестве

контроллера обмена данными выступил микроконтроллер STM32F103B8C6. Для размещения локального веб сервера был выбран модуль ESP12-E с микроконтроллером ESP8266. Для этапа отладки оборудования и программирования модуля ESP12-E необходимо было приобрести USB – TTL преобразователь. Также, для программирования микроконтроллера STM32 необходим внутрисхемный программатор/отладчик JTAG.

### **1.2.1. Микроконтроллер STM32**

STM32 — это микроконтроллер, построен на ядре ARM Cortex-M3 [6]. Данное ядро имеет много преимуществ, которые будут перечислены ниже, но его основное преимущество на сегодняшний день — универсальность. За два года Cortex-M3 стал индустриальным стандартом. Об этом говорит количество производителей, присоединившихся к данной архитектуре. Все основные производители микроконтроллеров, которые присутствуют в России, кроме Microchip, имеют или развивают решения на основе этой архитектуры: STMicroelectronics, Texas Instrument, NXP, ATMEL, Analog Devices, Renesas и т.д. Компания ST одна из первых выпустила свои микроконтроллеры Cortex-M3 (2007 г.) и быстро стала доминирующим игроком на этом рынке [7].

Преимущества STM32:

- «Pin-to-pin совместимость» – означает, что для одного размера корпуса все сигналы сохраняются на тех же самых вводах/выводах для разных вариантов микроконтроллеров семейства.
- Семейство STM32 отличается большим выбором бесплатных библиотек.
- Имеет также много (до 80 в варианте корпуса LQFP100) вводов/выводов общего назначения (GPIO). Эти выводы имеют высокую степень гибкости, их можно не только сконфигурировать в разные стандартные режимы (двухтактная схема, открытый коллектор, pull-up, pull-down и т.д.),



но и переназначить входами или выходами для периферии (remapping). Скорость передачи данных регулируется в целях снижения электромагнитных помех.

- Семейство STM32 отличается от конкурентов стабильностью работы в температурном диапазоне от  $-40$  до  $85^{\circ}\text{C}$ . Производительность ядра и периферии сохраняется полностью. В семействе STM32 есть ряд изделий, сертифицированных на расширенный температурный диапазон от  $-40$  до  $105^{\circ}\text{C}$ .

- Ядро Cortex-M3 имеет механизм перехода в спящий режим, и этот механизм был дополнен собственными режимами семейства STM32F.

- встроенный DMA-контроллер, позволяет разгрузить ядро от обработки и передачи данных.

- Невысокая цена изделия (210руб [8]).

- специальные библиотеки для модуля Simulink в ПО Matlab от компании Mathworks, предназначенные для генерирования из Simulink готового C-кода под микроконтроллеры STM32 — возможность разработки программного кода в графическом формате [9];

- Низкое энергопотребление (36 мА на частоте 72 МГц, 2мкА в режиме ожидания);

Основные параметры микроконтроллера:

- Максимальная частота ядра 72 МГц;

- Объем памяти программ: 64 кБ;

- Тип памяти программ: Flash;

- Объем оперативной памяти: 20 кБ;

- Количество входов/выходов: 37;

- Интерфейсы: CAN, I<sup>2</sup>C, IrDA, LIN, SPI, UART/USART, USB;

- Периферия: DMA, Motor Control PWM, PDR, POR, PVD, PWM, Temp Sensor, WDT.

### **1.2.2. WiFi модуль ESP12-E**

WiFi модуль ESP-12E разработан компанией Ai-thinker и построен на базе процессора с ядром ESP8266, отличительной особенностью которого является наличие радиointерфейса WiFi. Ядро ESP8266 интегрировано в 32-битный микроконтроллер с ультранизким энергопотреблением Tensilica L106. Поддерживает работу на тактовых частотах 80 и 160 МГц, имеет возможность установки RTOS (операционной системы реального времени). Имеет микрополосковую антенну на плате модуля. [10].

Характеристики:

- 80 MHz 32-bit процессор Tensilica Xtensa L106.
- Соответствует набору стандартов связи для коммуникации в беспроводной локальной сетевой зоне IEEE 802.11 b/g/n Wi-Fi. Поддерживается WEP и WPA/WPA2 шифрование.
- 14 портов ввода-вывода(из них возможно использовать 11), интерфейсы SPI, I<sup>2</sup>C, I<sup>2</sup>S, UART, 10-битовый АЦП.
- Питание 2,2...3,6 В. Потребление до 215 мА в режиме передачи, 100 мА в режиме приема, 70 мА в режиме ожидания. Поддерживаются три режима пониженного потребления, все без сохранения соединения с точкой доступа: Modem sleep (15 мА), Light sleep (0.4 мА), Deep sleep (15 мкА)].

### **1.2.3. JTAG – программатор**

ST-LINK V2 – это внутрисхемный программатор и отладчик для микроконтроллеров серии STM8 и STM32. С помощью ST-LINK можно программировать и выполнять отладку по интерфейсам SWD и JTAG (для микроконтроллеров STM32). Все коммуникационные интерфейсы программатора (SWIM, SWD, JTAG) доступны для использования и выведены на 10-ти контактный разъем программатора. [11].

Программатор ST-LINK подключается к компьютеру через USB и может работать с различным программным обеспечением: для микроконтроллеров STM32 – это интегрированные среды разработки Atollic, IAR, KEIL, Tasking, Coocox.

Характеристики:

- Питание программатора ST-LINK непосредственно от USB.
- Поддержка подключения Full Speed USB 2.0.
- Не требует дополнительных проводов для подключения к USB (тип подключения USB-A).
- Два светодиода отображающих режим работы программатора, чтение и запись данных.
- Поддержка SWD, SWIM и JTAG интерфейсов с напряжением 3,3 вольта.
- Стабилизированное напряжение 3,3 вольта выведено на разъем программатора и позволяет питать программируемое устройство непосредственно от ST-LINK.

#### **1.2.4. TTL-USB преобразователь**

Модуль предназначен для подключения различных микроконтроллерных устройств к современному компьютеру. Со стороны компьютера используется USB-интерфейс, а со стороны контроллера - стандартный асинхронный приемо-передатчик "UART" без инверсии полярности и с TTL уровнями сигналов [12].

Характеристики:

- Преобразование интерфейса USB\_2.0 в UART (TTL).
- UART линии RxD и TxD без управления потоком.
- Скорость обмена до 2Mbps.
- Питание 5V от USB-шины.
- Выбор уровней сигналов RxD, TxD 3,3V/5V.
- Обеспечение питания микроконтроллерной системы.
- светодиодная индикация питания и активности RxD, TxD.
- драйверы для большинства популярных операционных систем: WIN98/ME/2000/XP/Server2003/2008/VISTA/Win7/8(32/64)/MAC(OS32/OS64), LINUX.

### 1.3. Выбор программного обеспечения

Для работы с оборудованием необходимо следующее программное обеспечение:

- Keil uVision – В данной среде происходит программирование контроллера STM32.
- STM32CubeMX – В данной программе происходит конфигурирование микроконтроллера, автоматически генерируются необходимые файлы на языке программирования C.
- Терминал «Terminal v1.9b by Br@y++» – данное терминальное приложение использовалось для отладки.
- ESPlorer V0.2.0 данный терминал написан специально для модуля с чипом ESP8266. Поддерживает стандартные AT (модемные) команды, также позволяет загружать файлы для прошивки NodeMCU и писать код для нее на языках MicroPython и Lua. Поддерживает подсветку синтаксиса и загрузку файлов на модуль. Имеет терминал ввода. Автоматическое дописывание управляющих символов (сдвиг каретки и переход на другую строку).
- ESP8266Flasher – программа для прошивки микроконтроллера модуля ESP8266. В ходе работы с ней была полностью удалена родная прошивка, работающая на AT командах, и установлена NodeMCU version 2.1.0 и использующая язык скриптов LUA версии 5.1.

#### 1.3.1. Программная среда Keil uVision для микроконтроллеров

Keil uVision – среда разработки, представляющая собой набор утилит для выполнения полного комплекса мероприятий по разработке программного обеспечения для микроконтроллеров [13].

Keil uVision позволяет работать с проектами любой степени сложности, начиная с введения и правки исходных текстов и заканчивая внутрисхемной отладкой кода и программированием ПЗУ микроконтроллера. От разработчика скрыта большая часть второстепенных функций, что сильно разгружает интерфейс и делает управление интуитивно понятным. Однако при возрастании сложности реализуемых задач, всегда можно задействовать весь

потенциал модулей, функционирующих под управлением единой оболочки. Среди основных программных средств Keil uVision можно отметить.

- Базу данных микроконтроллеров, содержащую подробную информацию обо всех поддерживаемых устройствах. Здесь хранятся их конфигурационные данные и ссылки на источники информации с дополнительными техническими описаниями. При добавлении нового устройства в проект все его уникальные опции устанавливаются автоматически.

- Менеджер проектов, служащий для объединения отдельных текстов программных модулей и файлов в группы, обрабатываемые по единым правилам. Подобная группировка позволяет намного лучше ориентироваться среди множества файлов.

- Встроенный редактор, облегчающий работу с исходным текстом за счет использования многооконного интерфейса, выделения синтаксических элементов шрифтом и цветом. Существует опция настройки в соответствии со вкусами разработчика. Редактирование остается доступным и во время отладки программы, что позволяет сразу исправлять ошибки или отмечать проблемные участки кода.

- Средства автоматической компиляции, ассемблирования и компоновки проекта, которые предназначены для создания исполняемого (загрузочного) модуля программы. При этом между файлами автоматически генерируются новые ассемблерные и компиляторные связи, которые в дальнейшем позволяют обрабатывать только те файлы, в которых произошли изменения или файлы, находящиеся в зависимости от изменённых.

Снимок экрана программы показан на рисунке 2

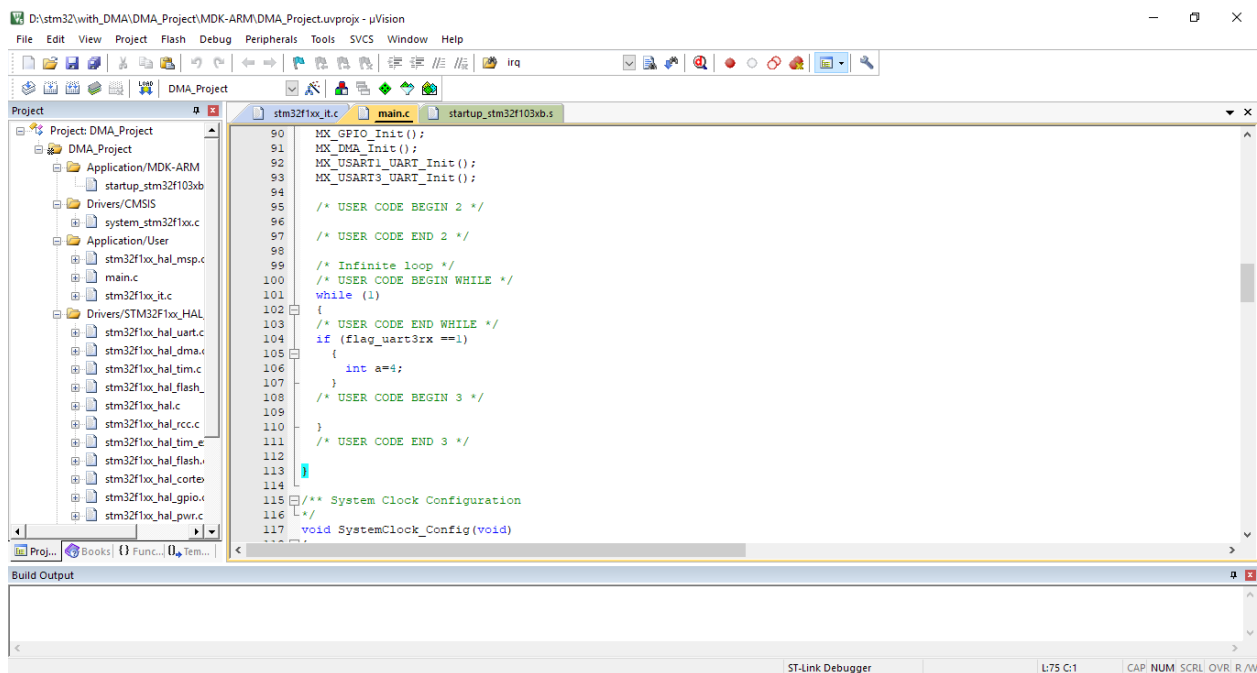


Рисунок 2 – Снимок экрана Keil uVision

### 1.3.2. Визуальный редактор настроек STM32CubeMX

Визуальный редактор настроек STM32CubeMX является составной частью инфраструктуры STM32Cube, которая также включает специализированные библиотеки для работы с различными линейками микроконтроллеров STM32. Стоит отметить, что интеграция между составными частями выполнена удачно, поскольку устанавливать необходимые библиотеки можно прямо из редактора. Для этого необходимо подключение к Интернету, поскольку загрузка производится с сайта компании STMicroelectronics. Также есть возможность отслеживать обновления версий уже загруженных библиотек. Кроме того, библиотеки могут использоваться самостоятельно при работе с другим программным обеспечением [14].

Пакет программного обеспечения, который предлагается для использования, включает в себя следующие основные компоненты:

- библиотеку Cortex Microcontroller Software Interface Standard (CMSIS) – это независимый от производителя уровень аппаратной абстракции для серии ядер Cortex-M, а также интерфейс отладчика (англ. debugger). CMSIS предоставляет последовательные и простые интерфейсы для ядра, его периферии и операционных систем реального времени.;

- библиотеку для работы с периферийными устройствами (Hardware Abstraction Layer – HAL);
- библиотеки, упрощающие использование возможностей популярных плат (Board Support Package – BSP);
- библиотеки более высокого прикладного уровня, позволяющие реализовать специализированные возможности (система реального времени FreeRTOS, поддержка файловой системы FAT, работа со стеком протоколов TCP/IP благодаря библиотеке lwIP, использование возможностей USB и др.). Не все библиотеки данного уровня могут быть доступны для подключения при использовании конкретного микроконтроллера.
- Библиотека HAL, которую можно назвать продолжением развития известной библиотеки Standard Peripheral Library (SPL), с которой она имеет много общего в концепции организации программного кода. Также архив с библиотеками содержит документацию, примеры кода и полезные дополнения.

Основными функциями, которые предоставляет визуальный редактор конфигурации, являются:

- настройка использования выводов периферийными устройствами в составе микроконтроллера;
- настройка тактирования;
- настройка периферийных устройств;
- расчет потребления энергии в соответствии с планируемыми режимами работы.

Снимок экрана окна программы STM32CubeMX показан на рисунке 3

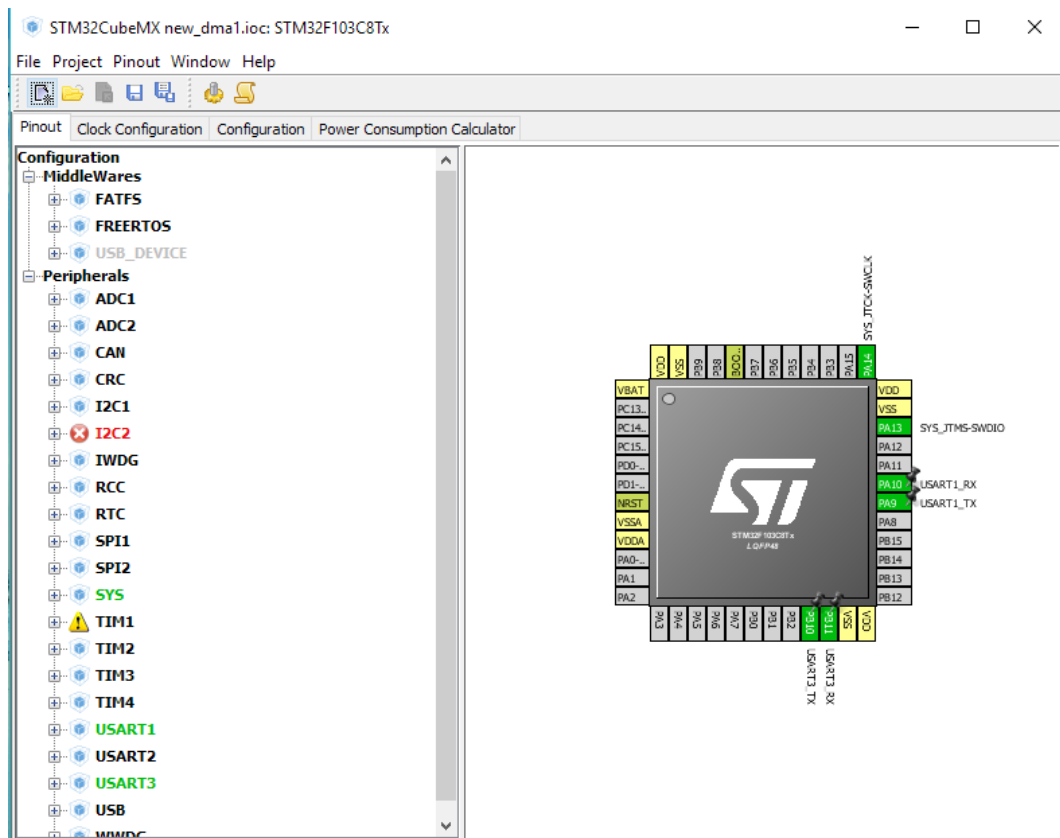


Рисунок 3 – Снимок экрана окна программы STM32CubeMX

### 1.3.3. Терминальное приложение Terminal v1.9b by Br@y++

Для связи контроллера и компьютера с помощью UART использовался терминал, снимок экрана которого представлен на рисунке 4.

Основные возможности Terminal 1.9b:

- Работает без установки. Вся программа — один .exe-файл размером около 300Кб.
- Счетчик переданных и принятых байтов.
- Возможность отправки файлов.
- Возможность установить нестандартную скорость передачи данных.
- Поддерживает до 64 COM-портов.
- Возможность ведения журнала работы в виде файла.
- Возможность назначения до 24 макросов.



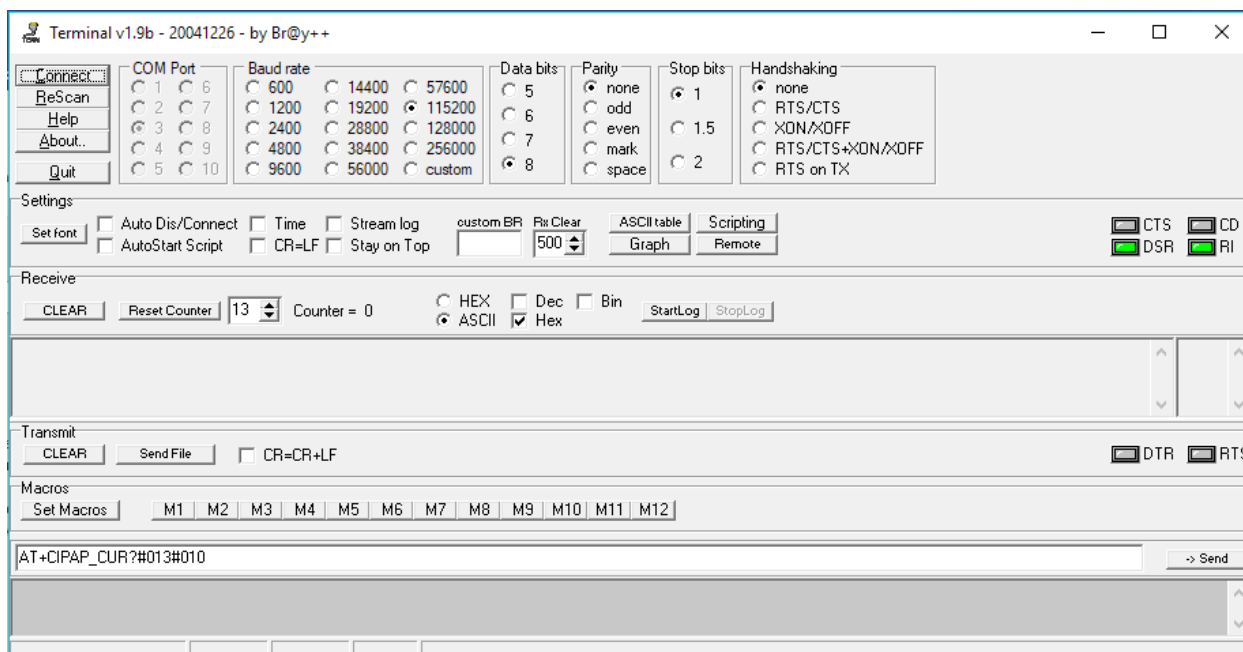


Рисунок 4 – Снимок экрана окна программы Terminal v1.9b by Br@y++

### 1.3.4. Терминальные приложения для микроконтроллера ESP8266 ESPlorer V0.2.0

ESPlorer V0.2.0 – Основной мультиплатформенный инструмент для работы с ESP8266 от авторов языка Lua, включая поддержку языков LUA и MicroPython для прошивок NodeMCU. Также имеется поддержка AT команд (модемные команд) [15].

Поддержка платформ [16]:

- Windows(x86, x86-64);
- Linux(x86, x86-64, ARM soft & hard float);
- Solaris(x86, x86-64);
- Mac OS X(x86, x86-64, PPC, PPC64).

Отличие от других программ для ESP8266:

- Работает на множестве платформ (NodeMCU, ESP01 и т.д.);
- Поддержка нескольких открытых файлов;
- Подсветка кода LUA, Python;
- Отмена последних изменений;
- Цветовые темы редактора: dark, Eclipse, IDEA, Visual Studio;
- Автозавершение кода по CTRL+Space;

- «Умная» отправка файлов, с ожиданием ответа (вместо отправки с фиксированной паузой между строками);

### 1.3.5. ESP8266Flasher (NodeMCU flasher)

NodeMCU flasher – Загрузчик программного обеспечения (прошивок) в микроконтроллер ESP8266. Снимок экрана окна программы показан на рисунке

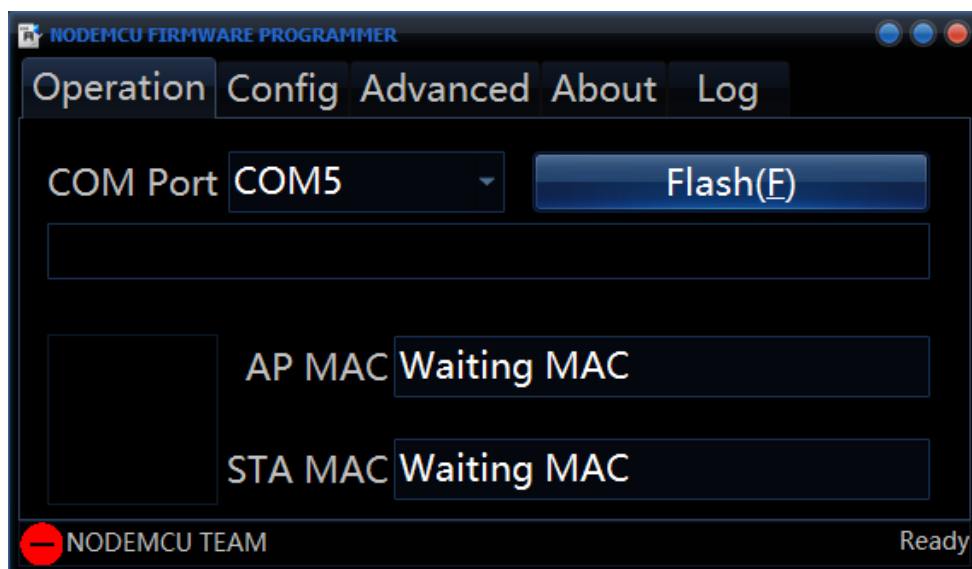


Рисунок 5 – Снимок экрана окна программы NodeMCU flasher

## 1.4.Используемые протоколы передачи данных

Для работы были выбраны следующие протоколы и технологии передачи данных: протокол UART и технология Wi-Fi.

### 1.4.1. Протокол передачи данных UART

UART (Universal Asynchronous Receiver-Transmitter) универсальный асинхронный приёмопередатчик (УАПП) — узел вычислительных устройств, предназначенный для организации связи с другими цифровыми устройствами. Преобразует передаваемые данные в последовательный вид так, чтобы было возможно передать их по цифровой линии другому аналогичному устройству.

Метод преобразования хорошо стандартизован и широко применяется в компьютерной технике. Для передачи данных используется две линии - RxD и TxD [17].

Принцип работы:

2 устройства подключаются перекрестно, то есть линия TxD первого устройства подключается к RxD и наоборот, RxD первого – к TxD второго. Передача данных осуществляется от передатчика (TxD) к приемнику (RxD).

Передача осуществляется побитово за равные промежутки времени. Временной промежуток определяется скоростью работы UART и задается для каждого соединения. Общепринятый ряд стандартных скоростей: 300; 600; 1200; 2400; 4800; 9600; 19200; 38400; 57600; 115200; 230400; 460800; 921600 бод (бит/с).

Принято соглашение, что пассивным (в отсутствие потока данных) состоянием входа и выхода UART является логическая 1. Стартовый бит всегда логический 0, поэтому приёмник UART ждёт перепада из 1 в 0 и отсчитывает от него временной промежуток в половину длительности бита (середина передачи стартового бита). Если в этот момент на входе всё ещё 0, то запускается процесс приёма минимальной посылки. Для этого приёмник отсчитывает 9 битовых длительностей подряд (для 8-битных данных) и в каждый момент фиксирует состояние входа. Первые 8 значений являются принятыми данными, последнее значение проверочное (стоп-бит). Значение стоп-бита всегда 1, если реально принятое значение иное, UART фиксирует ошибку.

Формат передающихся данных (1 старт-бит, 8 битов данных и 1 стоп-бит) по UART представлен на рисунке 6.

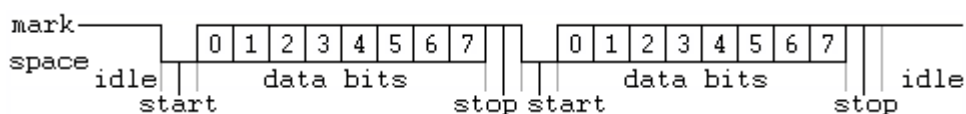


Рисунок 6 – Формат сообщения UART

### 1.4.2. Технология Wi-Fi

Wi-Fi – протокол и стандарт на оборудование для широкополосной радиосвязи, предназначенной для организации локальных беспроводных сетей.

Wi-Fi - это бренд, объединяющий несколько различных стандартов беспроводной связи. IEEE 802.11a, к примеру, обеспечивает скорость до 54

Мбит/с, используя рабочие частоты около 5 ГГц. IEEE 802.11b -11 Мбит/с на частоте 2,4 ГГц, IEEE 802.11g -54 Мбит/с на частоте 2,4 ГГц [18].

#### Преимущества Wi-Fi:

- За счет отсутствия проводов экономит время и средства на их прокладку и разводку. Сеть можно расширять практически бесконечно, увеличивая количество потребителей и геометрию сети установкой дополнительных точек доступа. Иногда проводную сеть нельзя построить физически.
- Глобальная совместимость. Wi-Fi - это семейство глобальных стандартов (несмотря на некоторые ограничения, существующие в разных странах), поэтому по идее устройство, произведенное в США, должно работать в России. И наоборот.
- В пределах Wi-Fi зоны в сеть Интернет могут выходить несколько пользователей с компьютеров, ноутбуков, телефонов и т. д.;

#### Недостатки Wi-Fi:

- Правовой аспект. В различных странах по-разному подходят к использованию частотного диапазона и параметрам передатчиков/приемников беспроводного сигнала стандартов IEEE 802.11. В одних странах, к примеру, требуется регистрация всех Wi-Fi сетей, работающих вне помещений. В других налагается ограничение на используемые частоты или мощность передатчика. В России использование Wi-Fi без разрешения на использование частот от Государственной комиссии по радиочастотам (ГКРЧ) возможно для организации сети внутри зданий, закрытых складских помещений и производственных территорий. Если вы хотите связать радиоканалом два соседних дома, рекомендуется обратиться в вышеупомянутый надзорный орган.
- Стабильность связи. Стандартные домашние Wi-Fi маршрутизаторы распространенных стандартов 802.11b или 802.11g имеют радиус действия порядка 40-50 метров в помещении и до 90 метров снаружи. Некоторые электронные устройства (микроволновая печь), погодные явления (дождь)

ослабляют уровень сигнала. Также расстояние зависит от рабочей частоты и других факторов.

- **Перекрестные помехи.** При большой плотности точек доступа могут возникнуть проблемы доступа к открытой точке доступа при наличии рядом точки доступа, работающего на том же или соседнем канале и использующем шифрование.

- **Факторы производства.** К сожалению, производители не всегда четко придерживаются стандартов, поэтому некоторые устройства могут работать нестабильно или на меньших скоростях.

- **Энергопотребление.** Достаточно высокое потребление энергии, что уменьшает время жизни батарей и повышает температуру устройства.

- **Безопасность.** Стандарт шифрования WEP, по-прежнему остающийся самым популярным, относительно легко взламывается. Более совершенный протокол WPA, к сожалению, не поддерживают многие старые точки доступа.

- **Ограниченная функциональность.** При передаче небольших пакетов данных к ним присоединяется большое количество служебной информации, что ухудшает качество связи. Поэтому Wi-Fi не рекомендуется использовать для работы в IP-телефонии, использующей протокол RTP (передача данных в реальном времени, например, голосовая и видеосвязь): качество связи не гарантировано.

### **1.5. Заключение по разделу 1**

В ходе работы была изучена техническая документация по имеющемуся оборудованию. Для работы с оборудованием было подобрано программное обеспечение.

## 2. Техническая реализация системы дистанционного управления

### 2.1. Подключение оборудования

#### 2.1.1. Запуск модуля ESP12-E

Для подключения оборудования на плате модуля были распаяны разъемы типа «плата-плата». Согласно инструкции по эксплуатации модуля [19], данный модуль имеет следующий порядок выводов (рисунок 7).

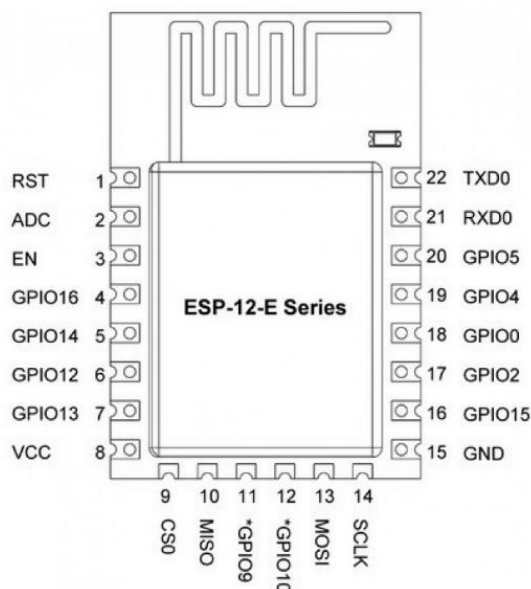


Рисунок 7 – Расположение выводов ESP12-E

Запуск модуля в режиме загрузки прошивки из внутренней памяти осуществлялся подачей высокого активного уровня (+3.3В) на выводы VCC и EN, и низкого (0В) на выводы GND и GPIO15.

Запуск модуля в режиме загрузки прошивки по UART каналу необходимо аналогично подключить выводы VCC, EN, GND и GPIO15, а на GPIO0 необходимо получить низкий логический уровень (использовался провод с разъемами типа «мама-мама» для получения 0В на этом выводе от вывода GND).

Выводы TxD и RxD использовались для приема и передачи данных по протоколу UART.

Описание всех выводов приведено в приложении А.

## 2.1.2. Запуск микроконтроллера STM32F103

Для первичного включения отладочной платы был подключен JTAG программатор к соответствующим выводам на плате в порядке, приведенном ниже:

Вывод STM32	Вывод JTAG программатора
DIO	SWDIO (2)
GND	GND (4)
DCLK	SWCLK (6)
3.3 V	3.3V (8)

Общий вид отладочной платы STM32 представлен на рисунке 8



Рисунок 8 – Общий вид отладочной платы

Программатор имеет 2 стороны, на которой одной из которых располагается USB разъем подключаемый, а с обратной имеет выводы типа «папа» подключённые к STM32. Общий вид представлен на рисунке 9.



Рисунок 9 – Внешний вид ST link v2 программатора JTAG для STM32

### 2.1.3. Подключение TTL-USB преобразователя.

Имеющийся преобразователь имеет 6 выводов: (5V, VCC, 3V3, TXD, RXD, GND). Так как преобразователь используется в работе и для программирования модуля, и для отладки STM32 то в ходе работы он был использован следующим образом:

Для Wi-Fi модуля – были замкнуты выводы VCC и 3V3 – установка уровня логики на 3В. К этому разъему подключался модуль на разъемы питания. GND подключался к GND модуля. TXD к RxD модуля, RXD к TxD модуля.

Для STM32: Так как STM32 не требователен к уровню напряжения логики (3–5В), то перемычка установки уровня логической единицы не была установлена на VCC и +5В. Питание для STM32 было подано с JTAG программатора, поэтому были подсоединены только контакты от STM32 RXD, TXD, GND к соответственно TRX, RXD и GND преобразователя.

Общий вид преобразователя показан на рисунке 10.

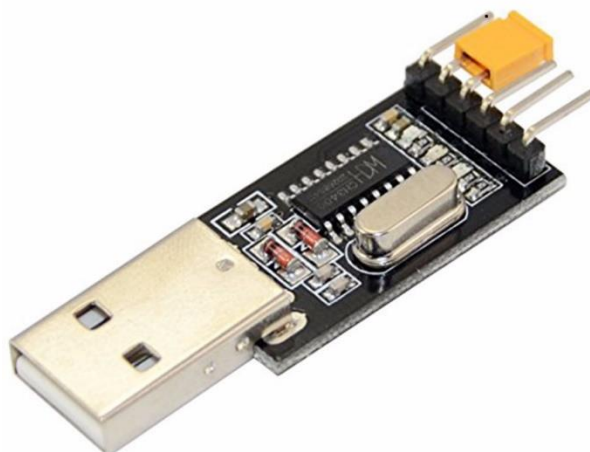


Рисунок 10 – Общий вид USB-to-TTL преобразователя HW-597

## 2.2. Проверка оборудования

После подключения оборудования было решено проверить его на работоспособность. Первым проверяемым звеном выступил TTL-USB преобра-



зователь HW-597. Для того, чтобы его увидела операционная система необходимо было установить драйвер. После этого в системе появился виртуальный порт. Далее замкнув линии TXD и RXD и запустив терминал «Terminal v1.9b», отправив информацию, (к примеру строка «Hello World!»), мы должны получить тоже самое в ответ. В строке, содержащей кнопку «Send» написали сообщение «Hello World» и получили его в строке «Receiver» (рисунок 11). Из этого можно сделать вывод о том, что TTL-USB преобразователь исправен.

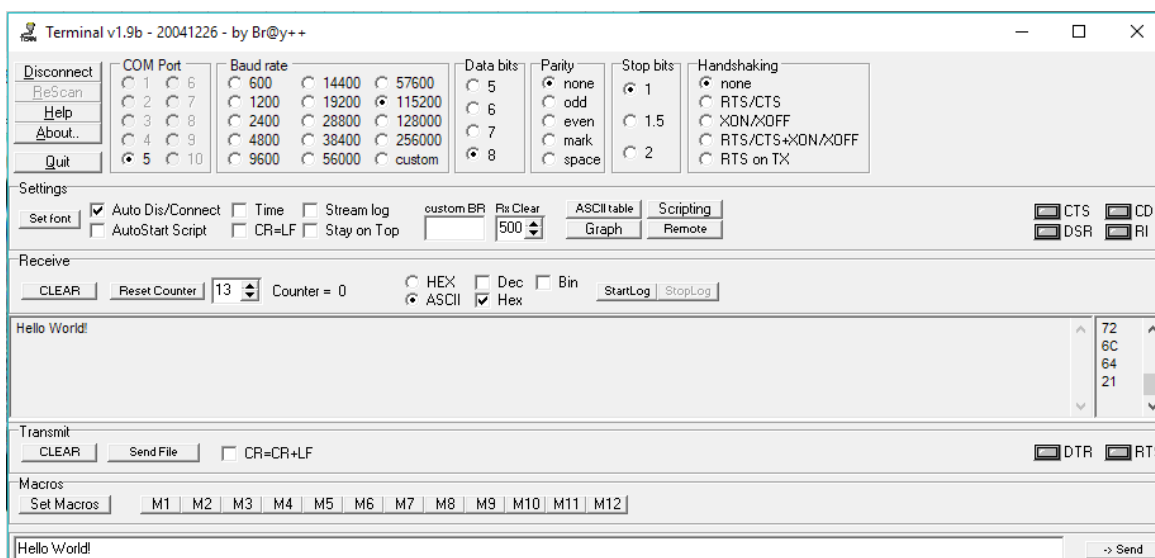


Рисунок 11 – Результат проверки преобразователя HW597

Следующим проверяемым элементом был модуль ESP12-E. Согласно инструкции по модулю в нем предустановлена прошивка, которая работает с помощью AT команд. Список AT-команд для версии прошивки 0.22, используемых для проверки приведен в таблице 1

Таблица 1 – Используемые команды

Команда	Описание	Параметры	Ответ
AT	Проверка модуля	—	«ОК»
AT+RST	Перезапуск модуля	—	После удачного перезапуска «ОК»

Команда	Описание	Параметры	Ответ
AT+GMR	Отобразить версию прошивки	—	Версия отображается в виде 8 цифр. Первая группа 4 цифры - версия AT (например, 0022), вторая группа 4 - версия SDK (например, 0100)
AT+UART_CUR= baudrate, data- bits, stopbits, parity, flow control	Настройка последовательного интерфейса	<p>скорость 4400-4608000;  биты данных:  5: 5 бит; 6: 6 бит; 7: 7 бит;  8: 8 бит</p> <p>стоп биты:  1: 1 стоп бит; 2: 1.5 стоп бит;  3: 2 стоп бит</p> <p>контроль четности:  0: нет; 1: Odd; 2: EVEN</p> <p>flow control  0 : flow control отключен  1 : включен RTS  2 : включен CTS  3 : включены оба RTS и CTS  (MTCK - UART0 CTS ,  MTDO - UART0 RTS)</p> <p>Пример:  AT+UART=115200,8,1,0,0</p>	—
AT+CWSAP_DEF=	Переключение режима wifi	<p>1=Station (WiFi клиент),  2=SoftAP (точка доступа),  3=Оба режима  (Station+SoftAP)</p> <p>Например,  AT+CWMODE_CUR=1</p>	—

При отправке данных команды проверки модуля, указания его версии или перезапуска ничего не происходило. Проблема была в том, что надо было дописывать символы возврата каретки и переноса строки (#010 и #013 код в терминале).

На рисунке 12 показан удачный ответ, после введения правильной команды.

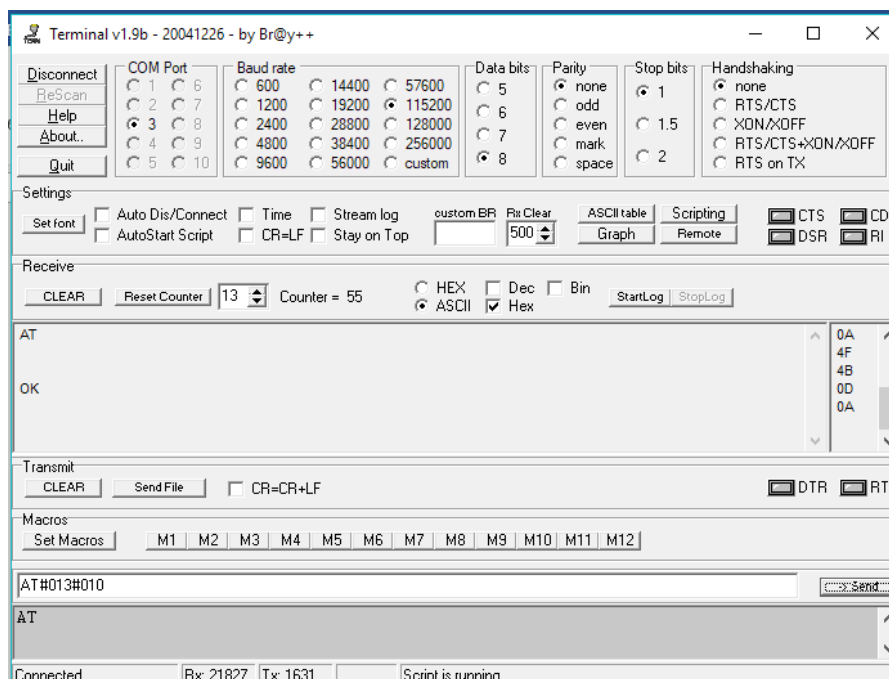


Рисунок 12 – Результат проверки модуля ESP12-E

В итоге, можно утверждать, что модуль ESP12-E исправен.

Проверка STM32 и JTAG программатора отдельно не выполнялась.

### 2.3. Реализация обмена данными между ESP и STM32

Как было описано выше, модуль ESP обладая родной прошивкой работает только AT-командами. Была сделана попытка реализовать связь с модулем через STM32. Для этого был написан код (рисунок 13).

```
while (b==0)
{
    HAL_UART_Receive_DMA(&huart3, (uint8_t*) str2, sizeof(str2)-1;
    HAL_UART_Transmit_DMA(&huart1, (uint8_t*) str2, sizeof(str2)-1;
    HAL_UART_Receive_DMA(&huart3, (uint8_t*) str, sizeof(str)-1;
    HAL_UART_Transmit_DMA(&huart1, (uint8_t*) str, sizeof(str)-1;
    b=1;
}
```

Рисунок 13 – Листинг кода

Программа по следующему принципу: для однократного выполнения программы на этапе отладки было поставлено условие (выполняться пока переменная b равна нулю), и далее вызывается функция приема данных из

UART3 (в данном случае из терминала) из библиотеки HAL. Принятые данные сохраняются в массив str2. `Sizeof(str2)-1` — устанавливает размер буфера приема.

Далее данные из сохранённого массива передаются в функцию `HAL_UART_Transmit_DMA`. Эта функция пересылает переданные в нее данные в порт UART1 (в данном случае в модуль ESP).

Следующая функция `HAL_UART_Receive_DMA` ждет данных из порта UART1 (из модуля ESP), в случае, когда сообщение полностью получено оно сохраняется в массив str.

Последняя функция аналогично пересылает массив данных str в порт UART3 (в консоль).

На рисунке 14 показан результат работы в терминале, а на рисунке 15 показано содержимое массивов в STM32 после выполнения программы.

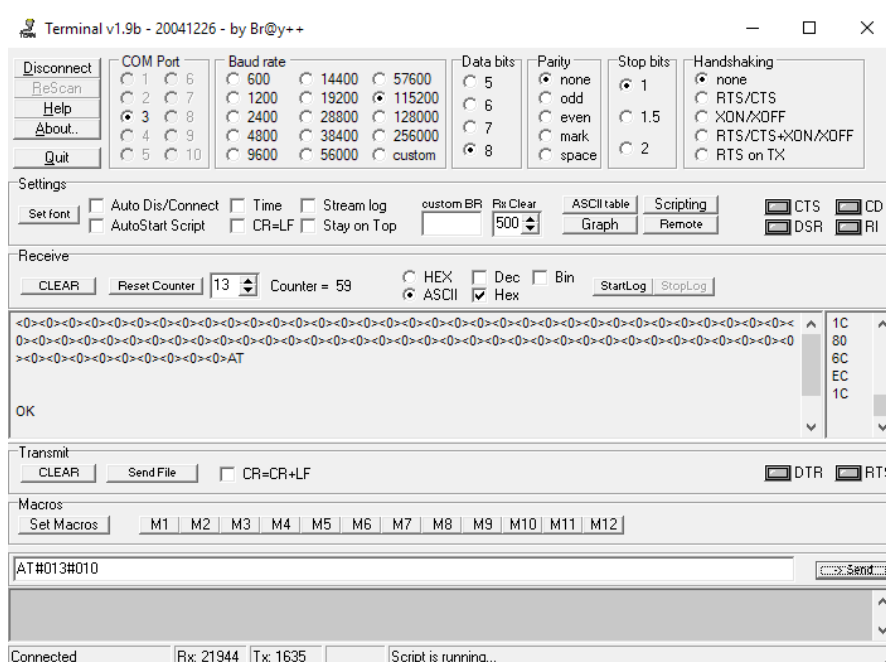


Рисунок 14 – Результат отправки команды AT и приема ответа

Name	Value	Type
str	0x20000018 str[] "AT\r\r\r\r\nOK\r\n"	char[40]
str2	0x20000040 str2[] "AT\r\n"	char[20]
Flag_rx_USART3	0x00000001	int
a	0x00000007	int
b	0x00000007	int
string	0x00000000	char[5]
<Enter expression>		

Рисунок 15 – Содержимое массивов после отработки кода

На рисунке показано, что модуль помимо ответа «ОК» выдал «эхо» введенной команды.

Данная реализация (при помощи AT-команд) связи STM32 и модуля ESP12-E показало низкую эффективность. Невозможность реализации пересылки данных с веб сервера и его реализация на стандартной прошивке. Поэтому на модуле ESP была установлена прошивка NodeMCU.

#### **2.4. Установка прошивки NodeMCU**

NodeMCU - прошивка содержащая в себе интерпретатор языка Lua и MicroPython. Позволяет размещать в памяти модуля файлы, то есть обладает файловой системой, может исполнять скрипты, написанные на Lua.

Для того чтобы установить прошивку была установлена и запущена программа ESP8266flasher. Подключив TTL-USB преобразователь к модулю и подав низкий логический уровень на GPIO0 включили модуль. После этого в программе выбрали файл с прошивкой и нажали кнопку «Flash(F)». В результате программа считывает AP MAC и STA MAC адреса модуля. После этого автоматически начался процесс загрузки и по завершении строка состояния изменила цвет (на зеленый).

После прошивки запустили ESPloer и запустили модуль в режиме загрузки прошивки из внутренней памяти. В результате увидели в консоли, какие модули (modules) включены в прошивку, дату сборки прошивки (build built on), а также ее версию (рисунок 16).

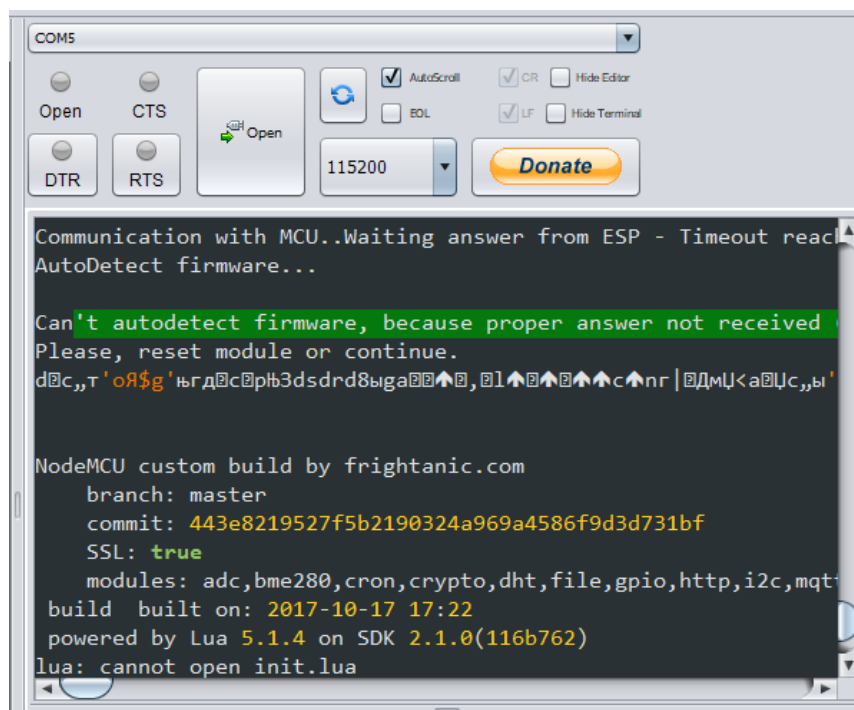


Рисунок 16 – Снимок экрана окна терминала

## 2.5. Реализация веб сервера на модуле ESP12-E

Для реализации веб сервера была создана HTML страница, листинг кода представлен на рисунке 17 Код страницы был написан в терминале ES-Plorer и сохранен как index.html в памяти модуля.

```

<html>
<title>Speed Regulator</title>
<body>
<h1>Управление скоростью вращения двигателя</h1>
<hr>
<form>
<p>Введите скорость:</p>
<p><input type="submit" value="Отправить"></p>
<p><input name="motor_speed" pattern="\d{1,4}"></p>
</form>
Действующая скорость: <div id="realspeed"> </div>
</body>
</html>

```

Рисунок 17 – Листинг кода HTML страницы

Далее был написан код для создания веб сервера. Листинг представлен на рисунке 18.

```

--Создаем сервер
sv=net.createServer(net.TCP)
--Функция получения данных
function receiver(sck, data)
  -- печать полученных данных (возврат данных метода GET)
  print(data)
  -- Send response
  sck:on("sent", function(sck) sck:close() end)
  filecontent = '';
  -- read file:
  if file.open("index.html", "r") then
    filecontent = file.read()
    file.close()
  end
  sck:send(filecontent)
end
--Выполнение подключения к серверу по запросу в 80 порт
if sv then
  sv:listen(80, function(conn)
    conn:on("receive", receiver)
  end)
end

```

Рисунок 18 – Листинг кода создания сервера

Код был сохранен в файле Page2.lua

Данный скрипт создает веб-сервер и ожидает подключения к 80 порту. Когда выполнено успешное подключение к 80 порту, автоматически вызывается функция receiver, в консоль выводится информация GET запроса браузера, браузеру отправляется файл index.html и закрывается соединение.

Далее была выполнена конфигурация сети:

- Сделать точкой доступа (режим StationAP)
- Создать имя точки доступа и пароль
- Установить IP адрес веб сервера.

Для простоты исправления параметров, был написан скрипт, названный NetCfg.lua и загружен в модуль. Листинг кода приведен на рисунке 19

```
wifi.setmode(wifi.STATIONAP)
cfg={}      -- таблица(массив) значений
cfg.ssid="diplom" --Имя точки доступа
cfg.pwd="12345123" --пароль
wifi.ap.config(cfg) --применение введенных данных
cfg=nil    --обнуление таблицы.
cfg={}
cfg.ip="192.168.4.1"--установка IP адреса
cfg.netmask="255.255.255.0" --маски подсети
cfg.gateway="192.168.1.1"
wifi.ap.setip(cfg) --применение значений
cfg=nil
collectgarbage()
print(wifi.ap.getip()) --вывод служебной информации
```

Рисунок 19 – Листинг кода настройки модуля

Теперь подключившись к этой точке доступа и перейдя по адресу 192.168.4.1 в браузере увидели следующее (рисунок 20).

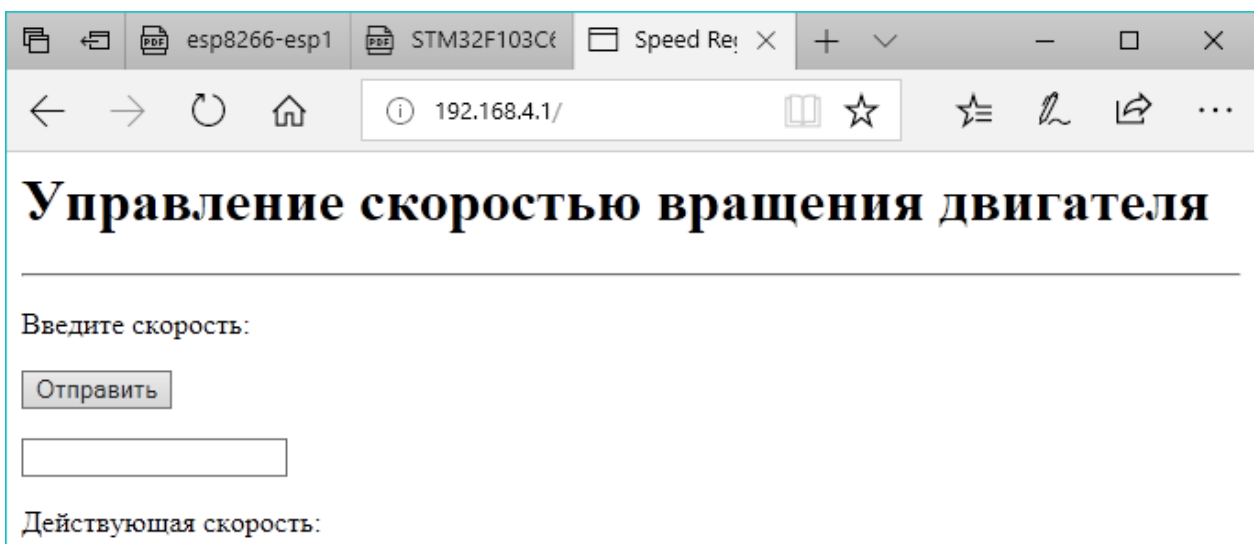


Рисунок 20 – Снимок экрана окна браузера с HTML страницей

Введя число в поле ввода и нажав кнопку «отправить» браузер посылает GET - запрос на сервер в формате, представленном на рисунке 21. Чтобы оперировать с данными формы, т.е. передать скорость двигателя дальше в контроллер обмена данными было необходимо из этого GET запроса извлечь пару ключ-значение, в данном случае motor\_speed=153.



```
GET /?motor_speed=153 HTTP/1.1
Accept: text/html, application/xhtml+xml, image/jxr, */*
Referer: http://192.168.4.1/
Accept-Language: ru-RU
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit
Accept-Encoding: gzip, deflate
Host: 192.168.4.1
Connection: Keep-Alive
```

Рисунок 21 – GET запрос от браузера к серверу, полученный сервером

Для реализации извлечения из GET запроса необходимой информации (значения пары «ключ=значение») был написан код [20], приведенный на рисунке 22.

```
local uri = string.match(datas, "[^?]*\?([^ ]*) [ ]", 1)
-- получение данных из GET запроса
GET={}
if uri then
  for key, value in string.gmatch(uri, "([^=&]*)=([^&]*)") do
    GET[key]=value
    print(key, value)
  end
end
```

Рисунок 22 – Листинг кода извлечения данных из GET запроса

Результат извлечения данных представлен на рисунке 23.

```
motor_speed 153
Accept-Encoding:
```

Рисунок 23 – Результат получения данных из GET запроса

Далее было реализовано получение данных по UART и передача их в модуль ESP12-E. Для этого был реализован код (рисунок 24). В нем обрабатывается событие получения данных по присутствию специального символа «\r» символа окончания строки.

```

uart.on("data", "\r",
function(data)
print("Received from uart:", data)
GET1={}
if data then
  for key, value in string.gmatch(data, "([^\&]*)=([^\&]*)") do
    GET[key]=value
    realspeed=value
    print("проверка принятых данных", value)
    -- print(key, value)
  end
end
uart.on("data") -- unregister callback function
end, 0)

```

Рисунок 24 – Листинг кода получения данных из UART

виде «ключ=значение»

Результат работы приведен на рисунке 25. Первая строчка – отправленная в терминал (UART порт) строка данных. Вторая строка сообщение-ответ, какие данные получены(эхо), третья строка показывает правильность расшифровки «ключ=значение», и выводит значение.

```

real_speed=50
Received from uart: real_speed=50
проверка принятых данных 50

```

Рисунок 25 – Принятые данные из UART «real\_speed=50»

## 2.6.Обмен данными с STM32 по UART

Для реализации обмена данными между ESP12-E и STM32 был написан код (рисунок 26).

```

HAL_UART_Receive_DMA(&huart3, (uint8_t*) str, sizeof(str)-1;
HAL_UART_Transmit_DMA(&huart1, (uint8_t*) str, sizeof(str)-1;

```

Рисунок 26 – Листинг кода

Данный код пересылает принятые из модуля ESP данные в систему управления двигателем. В качестве системы управления на момент отладки используем терминал компьютера.

В ходе работы возникла проблема, WiFi модуль хоть и создает точку доступа, но не выдает ответа браузеру на запрос подключения. Было выяснено, что для работы lua скриптов в модуле необходим их запуск, так как до

этого скрипты запускались автоматически при загрузке из терминала в модуль. Для этого был создан файл `init.lua`, листинг которого показан на рисунке 27.

```
dofile ("page2.lua")
```

Рисунок 27 – Листинг файла `init.lua`

Данный файл автоматически исполняется прошивкой модуля после его запуска. Данный файл запускает необходимый для работы файл `page2.lua`, который реализует веб сервер и отправляет пользователю html страницу.

Теперь снова было выполнено подключение к веб серверу, в форму ввода данных ввели число 153. Это значение вместе с названием формы ввода (`motor_speed`) и знаком «=>» были отправлены в контроллер обмена данными (STM32) по UART.

После нажатия кнопки «отправить» получили следующее: на рисунке 28 показаны значения регистров UART1 микроконтроллера STM32, где SR – флаг состояния на прерывание, DR – регистр данных. (снимок экрана получен в момент получения, пока обработчик не начал работу).

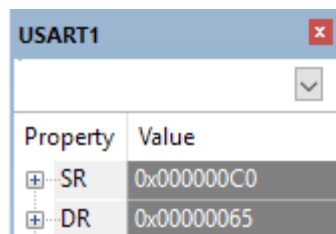


Рисунок 28 – Снимок экрана значений регистров

Принятые от модуля данные от модуля ESP12-E в STM32 в программе Keil режиме отладки показаны на рисунке 29.

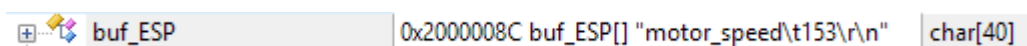


Рисунок 29 – Снимок окна с принятыми данными

## 2.7. Заключение по разделу 2

В данном разделе была достигнута основная задача работы, а именно реализация веб-интерфейса для управления параметрами электродвигателя. Было изучено и протестировано оборудование, а именно Wi-Fi модуль ESP12-E и микроконтроллер STM32, а также необходимые для их отладки и

программирования JTAG-программатор и TTL-USB преобразователь. Создан прототип системы. В ходе работы были частично изучены язык программирования Си и скриптовый язык Lua. Был переписан Wi-Fi модуль и написан код для STM32. Была разработана система с возможностью отображения HTML страниц, изменением технологического параметра и дальнейшей пересылкой данных из контроллера обмена данными в контроллер управления двигателя.

### **3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение**

Данный раздел определяет оценки коммерческого потенциала, перспективности и альтернатив проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения, а также планирование и формирование бюджета научных исследований, определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.

Научно-исследовательская работа направлена на разработку системы для дистанционного изменения параметров электрического двигателя посредством веб – интерфейса.

#### **3.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения**

##### **3.1.1. Потенциальные потребители продукта**

Целевой рынок обусловлен используемым оборудованием.

Объектом разработки является система дистанционного изменения параметров. Группу потребителей продукции могут составлять физические лица, а также малые компании.

Сегментирование рынка произведено по следующим критериям: размер компании заказчика и применяемое оборудование (контроллеры).

Таблица 2 – Карта сегментирования рынка

		Применяемое оборудование			
		Arduino	STM	Schneider Electric	Siemens
Размер компании	Крупные				
	Средние				
	Мелкие				
	Физические лица				

Из приведенной карты сегментирования можно сделать следующие выводы:

1. Крупные предприятия используют оборудование Schneider Electric или Siemens в виду их высокой надежности в качестве промышленного оборудования, а также в виду присутствия сертификатов соответствия.
2. Средние предприятия чаще используют продукцию компании Siemens в виду меньшей стоимости по отношению к Schneider Electric.
3. Мелкие предприятия реже используют Siemens, хотя у них и есть модели низкого ценового сегмента. Часто используются не промышленные контроллеры Arduino или STM.
4. Для физических лиц покупка контроллеров марки Siemens нецелесообразно, в виду их высокой стоимости и широкого функционала, редко необходимого вне предприятий.

### 3.1.2. Анализ конкурентных технических решений

В качестве отличительного критерия для выбора конкурентных технических решений используется используемое оборудование. Рассмотрим следующие варианты:

1. Микроконтроллер STM32
2. Микроконтроллер ARDUINO
3. Промышленный микроконтроллер Siemens.

Оценка технических характеристик представлена в таблице 3

Таблица 3 - Оценочная карта сравнения конкурентных технических решений

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б <sub>ф</sub>	Б <sub>к1</sub>	Б <sub>к2</sub>	К <sub>ф</sub>	К <sub>к1</sub>	К <sub>к2</sub>
1	2	3	4	5	6	7	8
Технические критерии оценки ресурсоэффективности							
1. Функциональная мощность (предоставляемые возможности)	0,1	4	3	5	0,4	0,3	0,5
2. Время создания проекта	0,05	4	4	3	0,2	0,2	0,15
3. Удобство в эксплуатации	0,1	4	4	5	0,4	0,4	0,5
4. Скорость работы	0,05	5	3	5	0,25	0,15	0,25
5. Безопасность	0,1	5	4	5	0,5	0,4	0,5
6. Простота разработки	0,1	4	5	2	0,4	0,5	0,2
7. Потребность в дополнительных ресурсах	0,2	4	4	4	0,8	0,8	0,8
Экономические критерии оценки эффективности							
1. Конкурентоспособность	0,05	5	3	4	0,25	0,15	0,2
2. Уровень проникновения на рынок	0,05	5	4	3	0,25	0,2	0,15
3. Цена продукта	0,1	5	5	2	0,5	0,5	0,2
4. Доступность	0,1	3	3	2	0,3	0,3	0,2
ИТОГО	1	48	42	40	4,25	3,9	3,65

- Под критерием «скорость работы» подразумевается частота процессора микроконтроллера;
- Под критерием «потребность в дополнительных ресурсах» понимается необходимость использования модулей расширения функционала;
- Под критерием «безопасность» подразумевается защита оборудования при сбоях;
- Под критерием «доступность» подразумевается фактическое наличие оборудования, например, для замены.

### 3.1.3. Технология QuaD

QuaD (QUality ADvisor) представляет собой гибкий инструмент измерения характеристик, описывающих качество новой разработки и ее перспективность на рынке и позволяющие принимать решение целесообразности вложения денежных средств в научно-исследовательский проект.

Таблица 4 - Оценочная карта сравнения конкурентных технических решений по технологии QuaD

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение(3/4)	Средневзвешенное значение (5*2)
1	2	3	4	5	6
<b>Показатели оценки качества разработки</b>					
1. Функциональная мощность (предоставляемые возможности)	0,1	90	100	0,9	0,09
2. Время создания проекта	0,05	50	100	0,5	0,025
3. Удобство в эксплуатации	0,1	70	100	0,7	0,07
4. Скорость работы	0,05	80	100	0,8	0,04
5. Безопасность	0,1	80	100	0,8	0,08
6. Простота разработки	0,1	80	100	0,8	0,08
7. Потребность в доп. Ресурсах	0,2	75	100	0,75	0,15
<b>Показатели оценки коммерческого потенциала разработки</b>					
1. Конкурентоспособность	0,05	100	100	1	0,05
2. Уровень проникновения на рынок	0,05	100	100	1	0,05
3. Цена продукта	0,1	90	100	0,9	0,09
4. Доступность	0,1	80	100	0,8	0,08
<b>ИТОГО</b>	<b>1</b>				<b>0,81</b>

Оценка качества и перспективности по технологии QuaD определяется по формуле:

$$P_{cp} = \sum B_i \times B_i$$



- где  $\Pi_{cp}$  – средневзвешенное значение показателя качества и перспективности научной разработки;  $B_i$  – вес показателя (в долях единицы);  $B_i$  – средневзвешенное значение  $i$ -го показателя. Значение  $\Pi_{cp}$  позволяет говорить о перспективах разработки и качестве проведенного исследования. Если значение показателя  $\Pi_{cp}$  получилось от 100 до 80, то такая разработка считается перспективной. Если от 79 до 60 – то перспективность выше среднего. Если от 69 до 40 – то перспективность средняя. Если от 39 до 20 – то перспективность ниже среднего. Если 19 и ниже – то перспективность крайне низкая.

$$\Pi_{cp} = 81$$

Данное значение лежит в интервале от 80 до 100, следовательно, такая разработка является перспективной.

### 3.1.4. SWOT-анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – представляет собой комплексный анализ научно-исследовательского проекта. SWOT анализ применяют для исследования внешней и внутренней среды проекта.

Матрица SWOT представлена в таблице 5

Таблица 5 – Матрица SWOT

	Сильные стороны научно-исследовательского проекта: С1: Безопасность С2: Стоимость С3: Скорость работы	Слабые стороны научно-исследовательского проекта: Сл1: Невысокое качество оборудования Сл2: Сложность разработки Сл3: Время разработки
Возможности: В1: Появление на рынке нового товара	В1В3С1С2С3: Система с высокой безопасностью, невысокой стоимостью и хорошей скоростью работы В2С2С1:	В2Сл1: Отладка системы на предприятиях на безвозмездной основе В1В3Сл2Сл3:

В2: Прирост предприятий потребителей В3: Выделение на фоне конкурентов	Снижение стоимости системы с той же безопасностью	Уменьшение времени разработки за счет найма опытных сотрудников
Угрозы: У1: Конкуренция У2: Незаинтересованность рынка в данном продукте	У1С2: Продажа продукции за счет регулирования стоимости У2С1С3: Создание спроса на рынке за счет достоинств	У1Сл2Сл3: Постоянный контроль за рынком и наём опытных кадров. У2Сл1: Рекламные акции продукции

Таблица 6 – Интерактивная матрица проекта (сильные стороны)

	С1	С2	С3
В1	+	+	+
В2	+	+	-
В3	+	+	+
У1	-	+	-
У2	+	-	+

Таблица 7 - Интерактивная матрица проекта (слабые стороны)

	Сл1	Сл2	Сл3
В1	-	+	+
В2	+	-	-
В3	-	+	+
У1	-	+	+
У2	+	-	

### 3.2. Определение возможных альтернатив проведения исследований

Для определения альтернативных путей проведения научных исследований и вариантов реализации технической задачи используется морфологический подход. Морфологическая матрица для составляющих реализации рассматриваемого проекта представлена в таблице 8.

Таблица 8 – Морфологическая матрица

		1	2	3	4
А	Контроллер	Siemens	STM32	ARDUINO	

		1	2	3	4
Б	Среда разработки	TIA Portal	Kail uVision	CoIDE	Arduino IDE
В	веб сервер	На модуле	На контроллере		
Г	Выход в глобальную сеть	Да	Нет		

- Выбор контроллера основывался на его стоимости и возможностях;
- Среда разработки выбирается согласно выбранному контроллеру;

В результате анализа была выбрана серия микроконтроллеров STM32.

Из полученной морфологической матрицы (таблица 8) можно получить следующие реализации и направления исследования:

- Исполнение 1: А1Б1В1Г2
- Исполнение 2: А2Б2В2Г1
- Исполнение 3: А2Б2В1Г2
- Исполнение 4: А2Б2В1Г1
- Исполнение 5: А2Б3В1Г2
- Исполнение 6: А3Б4В1Г2

Из них более предпочтительны варианты 2-4. Эти варианты будут использованы в дальнейших расчетах.

### **3.3. Планирование научно-исследовательских работ**

#### **3.3.1. Структура работ в рамках научного исследования**

Перечень работ и этапов научного исследования представлен в таблице 9.

Таблица 9 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№	Содержание работы	Исполнитель
Разработка технического задания	1	Составление и утверждение технического задания	Руководитель, Исполнитель
	2	Календарное планирование	Руководитель, Исполнитель
Аналитический обзор	3	Выбор материалов по теме и их изучение	Руководитель, Исполнитель
	4	Поиск похожих решений	Исполнитель
Проектирование системы	5	Разработка структуры системы	Исполнитель
	6	Подбор оборудования	Исполнитель
Монтаж оборудования	7	Проверка оборудования на неисправность	Исполнитель
	8	Сборка модулей в единую систему	Исполнитель
Реализация функциональной части	9	Изучение программной среды Kai uVision	Исполнитель
	10	Изучение протокола передачи данных	Исполнитель
	11	Изучение веб серверов.	Исполнитель
	12	Создание веб сервера	Исполнитель
	13	Обмен данными между модулями системы	Исполнитель
	14	Отладка	Исполнитель
Обобщение и оценка результатов	15	Оценка эффективности полученных результатов	Руководитель, Исполнитель
	16	Оценка целесообразности проведения дальнейший исследований по теме	Руководитель, Исполнитель

### 3.3.2. Определение трудоемкости выполнения работ

Трудовые затраты в большинстве случаев образуют основную часть стоимости разработки, поэтому важным моментом является определение трудоемкости работ каждого из участников научного исследования.

Трудоемкость выполнения научного исследования оценивается экспертным путем в человеко-днях и носит вероятностный характер, т.к. зависит от множества трудно учитываемых факторов. Для определения ожидаемого (среднего) значения трудоемкости  $t_{ож i}$  используется следующая формула:

$$t_{ож i} = \frac{3t_{min i} + 2t_{max i}}{5},$$

где  $t_{ож\ i}$  – ожидаемая трудоемкость выполнения  $i$ -ой работы чел.-дн.;  
 $t_{min\ i}$  – минимально возможная трудоемкость выполнения заданной  $i$ -ой работы (оптимистическая оценка: в предположении наиболее благоприятного стечения обстоятельств), чел.-дн.;

$t_{max\ i}$  – максимально возможная трудоемкость выполнения заданной  $i$ -ой работы (пессимистическая оценка: в предположении наиболее неблагоприятного стечения обстоятельств), чел.-дн.

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях  $T_p$ , учитывающая параллельность выполнения работ несколькими исполнителями. Такое вычисление необходимо для обоснованного расчета заработной платы, так как удельный вес заработной платы в общей сметной стоимости научных исследований составляет около 65 %.

$$T_{p\ i} = \frac{t_{ож\ i}}{Ч_i},$$

где  $T_{p\ i}$  – продолжительность работы, раб. дн.;

$t_{ож\ i}$  – ожидаемая трудоемкость выполнения одной работы, чел/дн.;

$Ч_i$  – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

### 3.3.3. Разработка графика проведения научного исследования

Для удобства построения графика, длительность каждого из этапов работ из рабочих дней следует перевести в календарные дни. Для этого необходимо воспользоваться следующей формулой:

$$T_{k\ i} = T_{p\ i} * k_{кал},$$

где  $T_{k\ i}$  – продолжительность выполнения  $i$ -й работы в календарных днях;

$T_{p\ i}$  – продолжительность выполнения  $i$ -й работы в рабочих днях;

$k_{кал}$  – коэффициент календарности.

Коэффициент календарности определяется по следующей формуле:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}},$$

где  $T_{\text{кал}}$  – количество календарных дней в году (365 дней);

$T_{\text{вых}}$  – количество выходных дней в году;

$T_{\text{пр}}$  – количество праздничных дней в году. ( $T_{\text{вых}} + T_{\text{пр}} = 66$ , для шестидневной рабочей недели)

$$k_{\text{кал}} = \frac{365}{365 - 66} \approx 1,22$$

Таблица 10 – Расчет трудозатрат

Название работы	Исполнители	Продолжительность работ, дни			Длительность работ, дн			
		t <sub>min</sub>	t <sub>max</sub>	t <sub>ож</sub>	Т <sub>рд</sub>		Т <sub>кд</sub>	
					НР	И	НР	И
Составление и утверждение технического задания	Р,И	4	4	4	4	4	4	4
Календарное планирование	Р,И	1	1	2	1	1	1	1
Выбор материалов по теме и их изучение	Р,И	10	15	12	5	10	5	12
Поиск похожий решений	И	3	5	4	-	4	-	5
Разработка структуры системы	И	3	8	4	-	4	-	5
Подбор оборудования	И	1	2	1	-	1	-	1
Проверка оборудования на неисправность	И	7	7	7	-	7	-	7
Сборка модулей в единую систему	И	1	2	1	-	1	-	1
Изучение программной среды Kail uVision	И	2	7	3	-	4	-	6
Изучение протокола передачи данных	И	5	7	6	-	8	-	12
Изучение веб серверов.	И	20	25	20	-	30	-	38
Создание веб сервера	И	10	15	12	-	15	-	15
Обмен данными между модулями системы	И	15	20	15	-	15	-	20
Отладка	И	4	8	5	-	9	-	14
Оценка эффективности полученных результатов	Р,И	14	20	14	3	6	3	7
Оценка целесообразности проведения дальнейший исследований по теме	Р,И	5	5	5	1	5	1	6
Итого по дням:							12	128

График проведения научного исследования представлен на рисунках 30 — 31.



Рисунок 30 – График проведения научного исследования для исполнителя

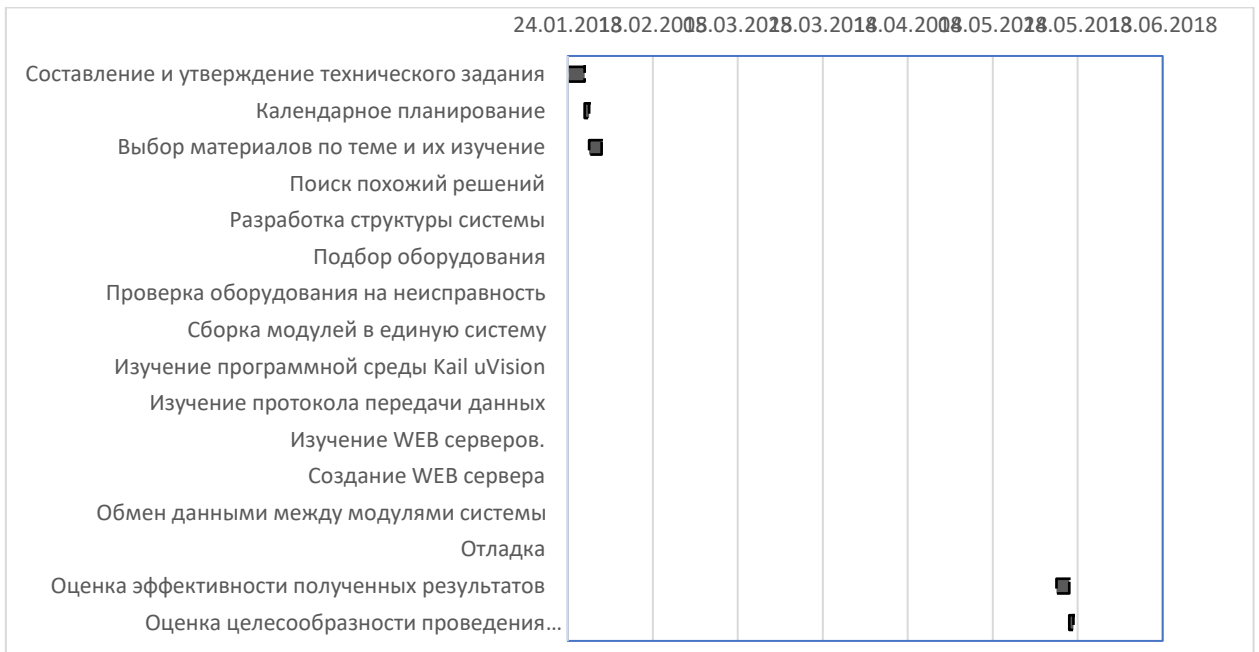


Рисунок 31 – График проведения научного исследования для руководителя

### 3.4. Бюджет научно-технического исследования

#### 3.4.1. Основная заработная плата исполнителей темы

Расчеты затрат на основную заработную плату приведены в таблице 11. При расчете учитывалось, что в 2018 году 299 рабочих дней. Основная заработная плата (ЗП) сотрудника от предприятия рассчитывается по следующей формуле:

$$Z_{\text{осн}} = Z_{\text{дн}} * T_p,$$



где  $Z_{\text{осн}}$  – основная ЗП одного работника;

$T_p$  – продолжительность работ, выполняемых научно-техническим работником (раб. дн.);

$Z_{\text{дн}}$  – среднедневная ЗП (руб.)

Среднедневная ЗП рассчитывается по формуле:

$$Z_{\text{дн}} = \frac{Z_M * M}{F_D},$$

где  $Z_M$  – месячный должностной оклад работника, руб.;

$M$  – количество месяцев работы без отпуска в течение года:

1. при отпуске в 24 раб. дня  $M = 11,2$  месяца, 5-дневная неделя;
2. при отпуске в 48 раб. дней  $M = 10,4$  месяца, 6-дневная неделя;

$F_D$  – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн. (299 рабочих дней в 2018 году).

Месячный должностной оклад работника:

$$Z_M = Z_{\text{ок}} * k_p,$$

где  $Z_{\text{ок}}$  – оклад (руб.);  $k_p$  – районный коэффициент, равный 1,3 (для Томска).

Расчет основной ЗП приведен в таблице 11

Таблица 11 – расчёт основной ЗП

Исполнитель	$Z_{\text{ок}}$ , руб.	$k_p$	$Z_{\text{дн}}$ , руб.	$T_p$ , раб. дн.	$Z_{\text{осн}}$ , руб.
НР	26300	1,3	1189	12	14268
И	17000	1,3	769	128	98432
Итого:					112700

Таким образом, затраты на основную заработную плату составили 81940 рублей.

#### 3.4.2. Дополнительная заработная плата исполнительской

Затраты по дополнительной заработной плате исполнителей темы учитывают величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда, а также выплат, связанных с обеспе-

чением гарантий и компенсаций (при исполнении государственных и общественных обязанностей, при совмещении работы с обучением, при предоставлении ежегодного оплачиваемого отпуска и т.д.).

Расчет дополнительной заработной платы ведется по следующей формуле:

$$З_{\text{доп}} = k_{\text{доп}} * З_{\text{осн}},$$

где  $k_{\text{доп}}$  – коэффициент дополнительной заработной платы (0,12-0,15);

$З_{\text{осн}}$  – основная заработная плата.

Получим:

$$З_{\text{доп НР}} = 0,12 * 14268 = 1712 \text{ руб.};$$

$$З_{\text{доп И}} = 0,12 * 98432 = 11812 \text{ руб.}$$

Заработная плата сотрудника без вычетов рассчитывается по формуле:

$$З_{\text{зп}} = З_{\text{осн}} + З_{\text{доп}}.$$

Тогда:

$$З_{\text{зп}} = 1712 + 14268 = 15981 \text{ для НР,}$$

$$З_{\text{зп}} = 98432 + 11812 = 110244 \text{ для И.}$$

### **3.4.3. Отчисление во внебюджетные фонды (страховые отчисления)**

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников.

Величина отчислений во внебюджетные фонды определяется исходя из следующей формулы:

$$З_{\text{внеб}} = k_{\text{внеб}} * (З_{\text{осн}} + З_{\text{доп}}),$$

где  $k_{\text{внеб}}$  – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.)

На 2014 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30%. На основании пункта 1 ст.58 закона №212-ФЗ для учреждений, осуществляющих образовательную и научную деятельность в 2014 году водится пониженная ставка – 27,1%. Отчисления во внебюджетные фонды рекомендуется представлять в табличной форме (таблица 12).

Таблица 12 – расчет отчислений во внебюджетного фонды

Исполнитель	Основная ЗП, руб.	Дополнительная ЗП, руб.	Коэффициент отчислений во внебюджетные фонды	$Z_{\text{внеб}}$ , руб.
НР	14268	1712	0,271	4331
И	98432	11812	0,271	29876
Итого:				34207

Получили, что всего будет перечислено 34207 руб. во внебюджетные фонды.

#### 3.4.4. Прочие расходы

Данная статья включает стоимость всех материалов, покупных изделий, полуфабрикатов и других материальных ценностей, расходуемых непосредственно в процессе выполнения работ. Цена материальных ресурсов определяется по соответствующим ценникам и приведена в таблице 13.

Таблица 13 – Затраты на оборудование и материалы

Наименование материалов	Цена за ед., руб.	Количество	Сумма, руб.
Бумага для принтера А4	170	1 шт.	170
Плата отладочная на базе микроконтроллера STM32	210	1 уп.	210
Внутрисхемный программатор/отладчик JTAG для МК STM8 и STM32	620	1 шт.	620
USB – TTL преобразователь на базе чипа sp2102	160	1 шт.	160
WI-FI модуль ESP12-E	480	1 шт.	480
Итого:			1640

Таким образом, расходы на материалы составляют 1640 руб.

Материальные затраты учитываются с учетом количества использованной электроэнергии. Для юридических лиц стоимость 1 кВт\*ч составляет 5,8 рублей. При умеренном пользовании компьютер средней мощности затрачивает 1,176 кВт в день в среднем.

$$\begin{aligned}
 Z_{\text{мат}} &= 1,176 \text{ кВт} * 128 \text{ дн.} * 7 \text{ ч} * 5,8 \frac{\text{руб.}}{\text{кВт} * \text{ч}} + Z_{\text{мат}} \\
 &= 6111 \text{ руб.} + 1640 \text{ руб.} = 7751 \text{ руб.}
 \end{aligned}$$

#### 3.4.5. Формирование бюджета затрат научно-исследовательского проекта

Определение бюджета затрат на научно-исследовательский проект по каждому варианту исполнения приведен в таблице 14. Так как все виды исполнений включают одинаковое оборудование, отличаясь только программной реализацией, то данные в таблице будут одинаковыми.

Таблица 14 – Бюджет затрат по каждому исполнению НТИ

Наименование статьи	Сумма, руб. (Исп2)	Сумма, руб. (Исп3)	Сумма, руб. (Исп4)
1. Материальные затраты НТИ и прочие расходы	7751	7751	7751
2. Затраты по основной заработной плате исполнителей	98432	98432	98432
3. Затраты по дополнительной заработной плате	11812	11812	11812
4. Отчисления во внебюджетные фонды	29876	29876	29876
5. Бюджет затрат НТИ	147871	147871	147871

### 3.4.6. Определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования

Научно-технический уровень характеризует, в какой мере выполнены работы и обеспечивается научно-технический прогресс в данной области. Для оценки научной ценности, технической значимости и эффективности, планируемых и выполняемых НИР, используется метод бальных оценок.

Таблица 15 – Сравнительная оценка характеристик вариантов исполнения объекта

Объект исследования Критерии	Весовой коэффициент параметра	Исп2	Исп3	Исп4
Способствует росту производительности труда	0,3	5	5	5
Удобство в эксплуатации (соответствует требованиям потребителей)	0,3	5	5	5
Помехоустойчивость	0,05	5	5	4
Энергосбережение	0,05	4	4	4

Надежность	0,15	5	5	4
Материалоемкость	0,15	4	5	4
Итого	1			

Рассчитаем интегральный показатель ресурсоэффективности по формуле:

$$I_{p i} = \sum a_i b_i,$$

где,  $I_{p i}$  – интегральный показатель ресурсоэффективности для  $i$ -го варианта исполнения разработки;

$a_i$  – весовой коэффициент  $i$ -го варианта исполнения разработки;

$b_i$  – балльная оценка  $i$ -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

В итоге, исходя из данных таблицы 15, получим:

$$I_{p \text{ исп}2} = 4,8;$$

$$I_{p \text{ исп}3} = 4,95;$$

$$I_{p \text{ исп}4} = 4,6.$$

Интегральный показатель эффективности вариантов исполнения разработки ( $I_{p i}$ ) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{\text{исп } i} = \frac{I_{p \text{ исп } i}}{I_{\text{финр}}^{\text{исп } i}},$$

Так как все исполнения имеют одинаковую стоимость, то  $I_{\text{финр}}^{\text{исп } i} = 1$  для каждого исполнения. Тогда интегральный показатель для исполнения 3 имеет наилучшую эффективность проекта и является наиболее целесообразным вариант из предложенных.

Сравнительная эффективность проекта определяется по формуле:

$$\mathcal{E}_{\text{ср}} = \frac{I_{\text{исп } 1}}{I_{\text{исп } 2}},$$

Тогда получим:

$$\mathcal{E}_{\text{ср } 1} = \frac{4,8}{4,95} = 0,97$$

$$\mathcal{E}_{\text{ср } 2} = \frac{4,95}{4,95} = 1;$$

$$\mathcal{E}_{\text{ср } 3} = \frac{4,6}{4,95} = 0,93$$

Сравнительная я эффективность разработки представлена в таблице 16.

Таблица 16 – Сравнительная эффективность разработки

Показатель	Исп2	Исп3	Исп4
Интегральный финансовый показатель разработки	1	1	1
Интегральный показатель ресурсоэффективности разработки	4,8	4,95	4,6
Интегральный показатель эффективности	4,8	4,95	4,6
Сравнительная эффективность вариантов исполнения	0,97	1	0,93

### 3.5. Заключение по разделу 3

Исходя из полученных данных и проведенного анализа эффективности можно сделать вывод, что вариант исполнения 3 является наиболее эффективным с позиции финансовой и ресурсоэффективности.

## **4. Социальная ответственность**

### **4.1. Аннотация**

Представление о понятии «Социальная ответственность» будущий специалист может получить из международного стандарта ICCSR26000:2011 «Социальная ответственность организации». В настоящем стандарте используются термины и определения, такие как: Социальная ответственность (social responsibility) – ответственность организации за воздействие ее решений и деятельности на общество и окружающую среду через прозрачное и этичное поведение, которое:

- Содействует устойчивому развитию, включая здоровье и благосостояние общества;
- Учитывает ожидания заинтересованных сторон;
- Соответствует применяемому законодательству и согласуется с международными нормами поведения;
- Интегрировано в деятельность всей организации и применяется в ее взаимоотношениях [21].

### **4.2. Введение**

В представленной работе разработана система дистанционного изменения параметров управления электрическим двигателем с применением веб-интерфейса.

Разработанная система позволяет дистанционно изменять параметры электрического двигателя с помощью устройства, подключенного к Wi-Fi сети системы, и изменять эти параметры на веб странице.

Работа выполнялась в 117а аудитории 10 корпуса НИ ТПУ в отделении автоматизации и робототехники.

### **4.3. Состав системы**

Основная часть работы включала использование персональных компьютеров.

В основе системы используется следующее оборудование:



- Wi-Fi 802.11 b/g/n модуль ESP12-E на базе процессора ESP8266 (рисунок 32);
- Микроконтроллер STM32 серии F103 (рисунок 33);
- Внутрисхемный программатор/отладчик JTAG для микроконтроллеров STM8 и STM32 (рисунок 34);
- USB – TTL преобразователь (рисунок 35).

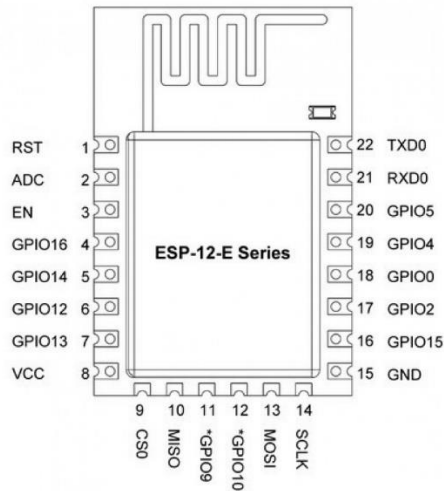


Рисунок 32 – Общий вид модуля ESP12-E



Рисунок 33 – Общий вид микроконтроллера STM32 серии F103



Рисунок 34 – JTAG для STM32



Рисунок 35 – USB – TTL преобразователь

Общая структурная системы представлена на рисунке 36.

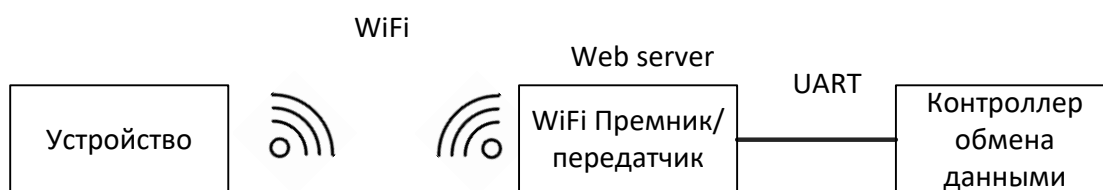


Рисунок 36 – Структурная схема системы управления

- Под устройством понимается любое оборудование, имеющее Wi-Fi приемно-передающую часть, будь то смартфон, ноутбук и т.д.;
- Wi-Fi приемник/ передатчик – в данном случае им выступает модуль ESP12-E. На данном модуле располагается веб -сервер;
- Контроллером обмена данными выступает микроконтроллер STM32 серии F103;

#### 4.4. Принцип работы системы

Оператор с помощью устройства, обладающего Wi-Fi технологией подключается к сети Wi-Fi системы. Далее необходимо зайти на веб -сервер, расположенный на Wi-Fi модуле ESP12-E. Теперь у пользователя в окне браузера отобразится информация о состоянии системы (статус двигателя, скорость его вращения и т.д.). Если необходимо изменить параметры, то пользователь изменяет их, и эти данные пересылаются в микроконтроллер STM32, откуда уходят дальше. Данные состояния присылаются пользователю через определенные интервалы времени.

## 4.5. Методы защиты данных

В данной работе есть 2 потенциальных места, в которых возможно возникновение ошибки в данных:

- При передаче данных с устройства оператора по Wi-Fi каналу связи;
- При обмене данными между Wi-Fi модулем.

### 4.5.1. Wi-Fi

Wi-Fi — это технология беспроводной локальной сети с устройствами на основе стандартов IEEE 802.11.

Для защиты данных от несанкционированного просмотра, изменения и отправки у данной технологии имеется средство защиты. Таким средством защиты являются протоколы шифрования WPA и WPA2. Эти протоколы используют авторизацию пользователя по логину и паролю. Таким образом, система получает защиту от несанкционированного доступа.

Еще одно из средств защиты сети, это запрещение трансляции SSID (Service Set Identification) – идентификатора сети. При запрещении трансляции данного идентификатора сеть будет существовать, однако она будет не видна для устройств. Чтобы к ней подключиться, необходимо вручную ввести этот идентификатор.

Защита данных от ошибок в передаче реализована на уровне стандарта Wi-Fi. Это реализовано аппаратно, в каждом устройстве, которое имеет знак соответствия Wi-Fi.

С другой стороны, технология Wi-Fi использует частоту 2.4 ГГц, на которой работают также микроволновые печи и Bluetooth устройства, что вносит помехи в данные в канале связи. Поэтому возможно использование другого частотного диапазона 5 ГГц. Так же возможно использование наименее загруженного канала.

Использование всех этих возможностей позволяет защитить сеть от несанкционированного доступа, защитить данные от ошибок при передаче.

#### **4.5.2. UART**

Данные между Wi-Fi модулем и микроконтроллером осуществляются посредством протокола UART.

UART – Universal asynchronous receiver/transmitter (универсальный асинхронный приемо-передатчик), старейший и самый распространенный на сегодняшний день физический протокол передачи данных.

Технологии UART минимально необходимо использовать 4 провода, 2 из них питания, остальные 2 – сигнальные (Rx – прием, Tx – передача).

Так как уровень сигнала достаточно невысок, то в линии могут возникать помехи и шумы, которые искажают цифровой код передающегося сигнала. Чтобы данные были правильно разобраны на стороне приемника, необходимо их защитить от помех.

Для защиты данных от ошибок возможно применение нескольких способов. Например, программная проверка данных на достоверность, использование разных скоростей передачи данных (чем ниже скорость передачи данных, тем меньше вероятность получить ошибочный бит данных, но тем меньше общая скорость передачи, т.е. необходимо соблюсти баланс).

Программная проверка данных на ошибки будет реализована в ходе улучшения системы, как только будет полностью налажена работа системы. Как вариант проверки данных можно использовать коды Хэмминга с обнаружением и исправлением ошибки, или использовать контрольную– сумму, и если рассчитанная сумма не соответствует переданной, то данные можно считать ошибочными, тогда следует повторить передачу.

#### **4.6. Факторы, влияющие на человека**

В данной системе электромагнитное излучение возникает только при использовании Wi-Fi передатчика-приемников.

В России точки беспроводного доступа, а также адаптеры Wi-Fi с эквивалентной изотропно-излучаемой мощностью (ЭИИМ), превышающей 100

мВт (20 дБм), подлежат обязательной регистрации [22]. Так как ЭИИМ модуля меньше 100 мВт, то следует, что используемое оборудование не вызывает негативного влияния на организм человека.

#### **4.7.Удобство в использовании**

При использовании разработанной системы у работников снижается физическое напряжение, связанное с нахождением в области работы двигателя, а также сотрудникам придется меньше отвлекаться на объект управления, так как данные представлены на экранах их устройств.

#### **4.8.Надежность системы**

Данная система достаточно надежна, так как все события, связанные с обработкой данных, исполняются не в бесконечном цикле, как например у микроконтроллеров Atmega AVR, а с помощью приоритета прерываний. В таком случае, когда в UART порт микроконтроллера приходят данные, ядро системы моментально производит обработку данных, временно приостанавливая работу основного кода. Это гарантирует, что поток данных будет обработан сразу, и не будет происходить потери информации, связанной с тем, что обработчик будет вызван в бесконечном цикле в какой-то момент времени.

Так же, микроконтроллер STM32 поддерживает аппаратную возможность использования RTC (Real Time Clock) – часов реального времени. Это гарантирует, что основная программа будет выполняться не более заданного времени и не сможет зациклиться.

#### **4.9.Возможности развития системы**

Для разрабатываемой системы планируется внести следующие улучшения:

- Разграничение прав доступа (начальник, сотрудник, программист и т.д.);
- Доступ к системе через сеть Интернет;

- Внесение ограничений изменяемых параметров для предотвращения опасных ситуаций;

- Реализация сигнализаций;

- Хранение данных (История операций).

Разграничением прав доступа к системе решается проблема того, что не всем сотрудникам нужно иметь полный доступ к настраиваемым параметрам.

Доступ к системе из сети Интернет имеет как плюсы, так и недостатки:

- К плюсам можно отнести то, что в любой момент времени сотрудник может следить за состоянием системы и менять параметры, даже не находясь на рабочем месте (например, начальник отдела может точно знать, что происходит).

- К минусам можно отнести большую уязвимость системы к кибератакам.

Ограничение изменения параметров позволит избежать опасных, предаварийных и аварийных ситуаций.

Реализация сигнализаций позволяет операторам визуально и с помощью звуковой сигнализации судить о состоянии системы: аварийный режим, предаварийный или обычный режим работы.

Хранение данных предполагается производить на внешней памяти с помощью карт расширения. На карте можно будет создать разделы с частью, которая хранит код программы, для резервного восстановления, а также другой раздел, в котором будет храниться информация о событиях.

#### **4.10. Заключение по разделу 4**

В ходе работы было дано понятие социальной ответственности, проанализировано влияние разработанной системы на человека. Доказано, что система не несет негативного воздействия на человека, а наоборот упрощает производственную деятельность. Так же в данном разделе была рассмотрена возможность дальнейшего развития системы.

## Заключение

В ходе работы была спроектирована структурная схема системы дистанционного изменения параметров управления скоростью электродвигателя с применением веб интерфейса. Было проверено и смонтировано оборудование. Был перепрошит Wi-Fi модуль. Так же были изучены среды графического редактирования настроек микроконтроллеров STM, среды программной разработки Keil uVision и изучен терминал-редактор кода ESPlorer для Wi-Fi модуля. Были получены начальные навыки в программировании на языках Си и Lua.

В процессе тестирования работоспособности системы были выявлены особенности обработки AT-команд модулем Wi-Fi, связанные с необходимостью добавлять в конце каждого сообщения символ возврата каретки. Также имеющаяся версия программного обеспечения микроконтроллера не позволяла реализовать функции веб-сервера, что вызвало необходимость смены прошивки. Отдельное решение было разработано для запуска веб-сервера. При каждой загрузке микроконтроллера автоматически загружается только системный файл, а запуск пользовательских файлов не происходил. Это потребовало внести изменения в системный файл. В результате доработки система показала стабильную работу на модельных сигналах.

На следующем этапе разработки планируется проведение тестовой эксплуатации на реальных объектах.

## Список литературы

1. Веб-сервер [Электронный ресурс]// Национальная библиотека им. Н.Э. Баумана. URL: <https://ru.bmstu.wiki/Веб-сервер> (дата обращения 16.03.2018).
2. HTTP [Электронный ресурс]// Национальная библиотека им. Н.Э. Баумана. URL: [https://ru.bmstu.wiki/HTTP\\_\(Hypertext\\_Transfer\\_Protocol\)](https://ru.bmstu.wiki/HTTP_(Hypertext_Transfer_Protocol)) (дата обращения 18.03.2018).
3. Lua [Электронный ресурс]// Национальная библиотека им. Н.Э. Баумана. URL: [https://ru.bmstu.wiki/Lua\\_\(язык\\_программирования\)](https://ru.bmstu.wiki/Lua_(язык_программирования)) (дата обращения 11.04.2018).
4. Язык программирования Си [Электронный ресурс]// Википедия. URL: [https://ru.wikipedia.org/wiki/Си\\_\(язык\\_программирования\)](https://ru.wikipedia.org/wiki/Си_(язык_программирования)) (дата обращения 13.04.2018).
5. Микроконтроллеры в современном мире. Лекция 1 [Электронный ресурс]//Студопедия. URL: [https://studopedia.su/13\\_93566\\_lektsiya--mikrokontrolleri-v-sovremennom-mire.html](https://studopedia.su/13_93566_lektsiya--mikrokontrolleri-v-sovremennom-mire.html) (дата обращения 16.04.2018).
6. Cortex M [Электронный ресурс]// ARM. URL: <http://www.arm.com/products/processors/cortex-m/index.php> (дата обращения 18.04.2018).
7. Новости электроники [Электронный ресурс]// Компэл. URL: <https://www.compel.ru/lib/ne/2011/2/3-stm32-epocha-32-bitnyih-mikrokontrollerov-nastupila> (дата обращения 21.04.2018).
8. STM32F103C8T6 [Электронный ресурс]// ЧипДип. URL: <https://www.chipdip.ru/product/stm32f103c8t6> (дата обращения 25.04.2018).
9. Модельно-ориентированное проектирование Cortex M3 с помощью Matlab/Simulink [Электронный ресурс]// Хабр. URL: <https://habr.com/post/129519/> (дата обращения 25.04.2018).



10. Wi-Fi модуль ESP-12E (на базе ESP8266) [Электронный ресурс]// RadioProg. URL: <http://radioprogram.ru/shop/merch/33> (дата обращения 26.04.2018).
11. Программатор-отладчик ST-Link V2 для STM8 STM32, поддержка JTAG [Электронный ресурс]// 5 Вольт. Электроника для творчества. URL: [http://5volt.com.ua/programming/st-link\\_ver2\\_jtag.html](http://5volt.com.ua/programming/st-link_ver2_jtag.html) (дата обращения 30.04.2018).
12. USB-UART-CH340 [Электронный ресурс]// IMRAD Электронные компоненты. URL: [https://imrad.com.ua/userdata/modules/wproducts/wprod\\_products/132485/Arduino%20USB-UART-CH340.pdf](https://imrad.com.ua/userdata/modules/wproducts/wprod_products/132485/Arduino%20USB-UART-CH340.pdf) (дата обращения 02.05.2018).
13. Keil uVision [Электронный ресурс]// Schem.net. Сайт паяльник. URL: <http://schem.net/software/keil.php> (дата обращения 03.05.2018).
14. Визуализация возможностей: графический генератор кода STM32CubeMX [Электронный ресурс]// Компэл. URL: <https://www.compel.ru/lib/ne/2014/11/4-vizualizatsiya-vozmozhnostey-graficheskii-generator-koda-stm32cubemx> (дата обращения 06.05.2018).
15. ESPlorer [Электронный ресурс]// esp8266 - Сообщество разработчиков. URL: <https://esp8266.ru/esplorer/> (дата обращения 09.05.2018).
16. ESPlorer — IDE для ESP8266 [Электронный ресурс]// esp8266 - Сообщество разработчиков. URL: <https://esp8266.ru/category/esplorer-info/> (дата обращения 09.05.2018).
17. Передача данных протокол UART [Электронный ресурс]// livejournal URL: <https://varyag-nord.livejournal.com/64490.html> (дата обращения 05.04.2018).
18. «Вай-Фай»: что это такое и с чем его «едят»? [Электронный ресурс]// ABNET URL: [https://abnet.am/all\\_about\\_wi-fi.html](https://abnet.am/all_about_wi-fi.html) (дата обращения 09.05.2018).

- 19.ESP-12E WiFi Module [Электронный ресурс]// kloppenborg. URL: <https://www.kloppenborg.net/images/blog/esp8266/esp8266-esp12e-specs.pdf> (дата обращения 23.04.2018).
- 20.Как сделать Web-интерфейс для ESP8266 под NodeMCU [Электронный ресурс]// Avislab. URL: [http://www.avislab.com/blog/nodemcu-web\\_ru/](http://www.avislab.com/blog/nodemcu-web_ru/) (дата обращения 26.04.2018).
- 21.Романенко С.В., Анищенко Ю.В. Социальная ответственность: методическое указание / Томский политехнический университет. - Томск: Изд-во Томского политехнического университета, 2016. -21 с.
- 22.Wi-Fi [Электронный ресурс]// Википедия. URL: <https://ru.wikipedia.org/wiki/Wi-Fi> (дата обращения 10.05.2018).

## Приложение А.

(справочное)

### Обозначение выводов модуля ESP12-E

Название вы- вода	Функция
RST	Сброс модуля при высоком уровне
ADC	АЦП 10 бит. Вход 0÷1В, значения 0÷1023
EN	При высоком уровне активирует модуль
GPIO16	GPIO16; Может быть запрограммирован для «пробуждения» модуля из режима «глубокого сна»
GPIO14	GPIO14; Может быть использован для тактирования HSPI
GPIO12	GPIO12; Может быть использован для HSPI_MISO
GPIO13	GPIO13; Может быть использован для HSPI_MOSI; UART0_CT
VCC	Вход питания модуля (3.3 В)
CS0	Chip Selection
MISO	Slave output Main input (передача от «подчиненного» к «мастеру»)
GPIO9	GPIO9
GPIO10	GPIO10
MOSI	Main output Slave input (передача от «мастера» к «подчиненному»)
GND	Общий
GPIO15	GPIO15; Может быть использован для: MTDO; HSPICS; UART0_RTS
GPIO2	GPIO2; Может быть использован для UART1_TXD
GPIO0	GPIO0
GPIO4	GPIO4
GPIO5	GPIO5
RxD	UART0_RxD – Приемник UART0; Может быть использован как GPIO3
TxD	UART0_TxD – Передатчик UART0; Может быть использован как GPIO1