

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

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Система оперативной индикации параметров технологического процесса

УДК 004.514:004.382.7

Студент

Группа	ФИО	Подпись	Дата
8Т5Б	Журман Дмитрий Александрович		

Руководитель ВКР

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

Нормоконтроль

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Суханов Алексей Викторович	К.Х.Н.		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Меньшикова Екатерина Валентиновна	к.ф.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД ШБИП	Винокурова Галина Федоровна	К.Т.Н.		

ДОПУСТИТЬ К ЗАЩИТЕ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ООП	Громаков Евгений Иванович	К.Т.Н., доцент		
Руководитель ОАР ИШИТР	Леонов Сергей Владимирович	К.Т.Н.		

Томск – 2019 г.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Код результата	Результат обучения (Выпускник должен быть готов)
Профессиональные компетенции	
P1	Обладать естественнонаучными и математическими знаниями для решения инженерных задач в области разработки, производства и эксплуатации систем управления техническими объектами и средств автоматизации.
P2	Обладать знаниями о передовом отечественном и зарубежном опыте в области управления техническими объектами с использованием вычислительной техники
P3	Применять полученные знания (P1 и P2) для формулирования и решения инженерных задач при проектировании, производстве и эксплуатации современных систем управления техническими объектами и их составляющих с использованием передовых научно-технических знаний, достижений мирового уровня, современных инструментальных и программных средств.
P4	Уметь выбирать и применять соответствующие методы анализа и синтеза систем управления, методы расчета средств автоматизации, уметь выбирать и использовать подходящее программное обеспечение, техническое оборудование, приборы и оснащение для автоматизации и управления техническими объектами.
P5	Уметь находить электронные и литературные источники информации для решения задач по управлению техническими объектами.
P6	Уметь планировать и проводить эксперименты, обрабатывать данные и проводить моделирование с использованием вычислительной техники, использовать их результаты для ведения инновационной инженерной деятельности в области управления техническими объектами.
P7	Демонстрировать компетенции, связанные с инженерной деятельностью в области научно-исследовательских работ, проектирования и эксплуатации систем управления и средств автоматизации на предприятиях и организациях – потенциальных работодателях, а также готовность следовать их корпоративной культуре
Универсальные компетенции	
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 работу коллектива при решении инновационных инженерных задач в области автоматизации и управления техническими объектами, 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 науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа – Инженерная школа информационных технологий и робототехники
 Направление подготовки – 15.03.04 «Автоматизация технологических процессов и производств»

Уровень образования – Бакалавриат

Отделение (НОЦ) – Отделение автоматизации и робототехники

Период выполнения – осенний / весенний семестр 2018/2019 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)
	Основная часть	75
	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
	Социальная ответственность	10

СОСТАВИЛ:

Руководитель ВКР

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

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Громаков Евгений Иванович	К.Т.Н., ДОЦЕНТ		

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа – Инженерная школа информационных технологий и робототехники
 Направление подготовки – 15.03.04 «Автоматизация технологических процессов и производств»
 Отделение (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
8Т5Б	Журман Дмитрию Александровичу

Тема работы:

Система управления положением объекта в трехмерном пространстве	
Утверждена приказом директора (дата, номер)	№1095/с от 12.02.2019

Срок сдачи студентом выполненной работы:	
--	--

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	<p>Цель работы: создание системы индикации оперативных значений параметров технологического процесса на мобильных устройствах при помощи микрокомпьютера Raspberry Pi.</p> <p>– Объектом исследования является система для индикации параметров технологического процесса для выявления отклонений процесса от заданных пределов и отказов технологического оборудования. В состав системы входят: микрокомпьютер, адаптер USB-RS485, дисплей, контроллер, соединительные провода.</p>
---------------------------------	--

Перечень подлежащих исследованию, проектированию и разработке вопросов	<ol style="list-style-type: none"> 1. Аналитический обзор уже существующих систем оперативной индикации параметров. 2. Выбор протокола для подключения к ПЛК. 3. Выбор библиотек для подключения к ПЛК. 4. Выбор протокола для передачи технологических параметров. 5. Разработка сервера сбора и хранения технологических параметров. 6. Разработка клиента для отображения параметров
Перечень графического материала	<ol style="list-style-type: none"> 1. Блок-схема работы клиентского приложения. 2. Скриншоты клиентского приложения. 3. Схема работы системы. 4. Схема подключения к ПЛК.
Консультанты по разделам выпускной квалификационной работы	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Меньшикова Екатерина Валентиновна
Социальная ответственность	Винокурова Галина Федоровна
Названия разделов, которые должны быть написаны на русском и иностранном языках:	
—	

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
---	--

Задание выдал руководитель:

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

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Т5Б	Журман Дмитрий Александрович		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8Т5Б	Журман Дмитрию Александровичу

Школа	ИШИТР	Отделение (НОЦ)	ОАР
Уровень образования	Бакалавриат	Направление/специальность	15.03.04 «Автоматизация технологических процессов и производств»

Тема ВКР:

Система оперативной индикации параметров технологического процесса	
Исходные данные к разделу «Социальная ответственность»:	
<p>1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения</p>	<p>Объектом исследования и разработки является система индикации параметров технологического процесса для выявления отказов технологического оборудования. В состав системы входят: микрокомпьютер, адаптер RS485-USB, контроллер. Областью применения данной системы являются предприятия, на которых установлена АСУ ТП и необходим постоянный контроль за производством.</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности:	Трудовой кодекс РФ от 30.12.2001 N 197-ФЗ (ред. от 01.04.2019). СанПиН 2.2.2/2.4.1340-03. ГОСТ 12.2.032-78 ССБТ.
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	При разработке и эксплуатации проектируемого решения выделены вредные и опасные факторы: повышенное значение напряжения в электрической цепи, отсутствие или недостаток естественного света, недостаточная освещенность рабочей зоны, физические статические перегрузки, перенапряжение анализаторов.
3. Экологическая безопасность:	Воздействие объекта на литосферу: утилизация электротехнических компонентов и отходов при изготовлении системы.
4. Безопасность в чрезвычайных ситуациях:	Среди возможных ЧС при разработке и эксплуатации проектируемого решения является возникновение пожара.

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

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД ШБИП	Винокурова Галина Федоровна	к.т.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Т5Б	Журман Дмитрий Александрович		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСООБЪЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8Т5Б	Журман Дмитрию Александровичу

Школа	Отделение школы (НОЦ)	15.03.04
Уровень образования	Направление/специальность	«Автоматизация технологических процессов и производств»
	Бакалавриат	

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

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Бюджет – 229749 Затраты на заработную плату – 89712,81 на спец. оборудование – 309 Материальные затраты НТИ – 100483,71 Прочие – 807,36
2. Нормы и нормативы расходования ресурсов	Тариф на электроэнергию - 5,8 кВт/ч Норма амортизации – 10%
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Налог во внебюджетные фонды 27,1 Районный коэффициент – 1,3 Накладные расходы – 16%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Оценка коммерческого потенциала инженерных решений (ИР)	Оценка потенциальных потребителей исследования, SWOT – анализ,
2. Формирование плана и бюджета инженерного проекта (ИП)	Планирование этапов работ, определение трудоемкости и построение календарного графика, формирование бюджета.
3. Оценка ресурсной, финансовой, социальной, бюджетной эффективности ИР и потенциальных рисков	Оценка сравнительной эффективности исследования

Перечень графического материала (с точным указанием обязательных чертежей)

1. Оценка конкурентоспособности НТИ
2. Матрица SWOT
3. График разработки
4. Бюджет НТИ
5. Оценка ресурсной, финансовой и экономической эффективности НТИ

Дата выдачи задания для раздела по линейному графику

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Меньшикова Екатерина Валентиновна	к.ф.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Т5Б	Журман Дмитрий Александрович		

Реферат

Пояснительная записка содержит 97 страницы машинописного текста, 18 рисунков, 11 таблиц, 1 список использованных источников из 23 наименований и 3 приложения.

Цель работы: создание системы индикации оперативных значений параметров технологического процесса на мобильных устройствах при помощи микрокомпьютера Raspberry Pi.

Объектом исследования является система для индикации параметров технологического процесса для выявления отклонений процесса от заданных пределов и отказов технологического оборудования. В состав системы входят: микрокомпьютер, мобильный клиент, сервер для передачи параметров технологического процесса, адаптер USB-RS485, дисплей, контроллер, соединительные провода.

В процессе работы были разработаны: сервер для передачи параметров технологического процесса, мобильный клиент, функция кэширования параметров для сервера и структура запроса.

Также был создан следующий графический материал: архитектура разрабатываемой системы, блок-схема схема кэширования параметров, интерфейс мобильного приложения.

Разработанная система может быть применена на различных предприятиях, где используется АСУ ТП, для того, чтобы оперативно оповещать рабочий персонал об отклонении процесса от заданных пределов и аварийной ситуации.

Ключевые слова: Raspberry Pi, Siemens S7-200, Modbus, TCP, Kivy, ПЛК, сервер, клиент, запрос, параметр.

Содержание

Обозначения и сокращения	12
Введение	13
1. Системы индикации технологических параметров	15
1.1 MasterSCADA 4D	15
1.2 SimpLight	16
1.3 Мобильное приложение мониторинга и управления системами АСУ ТП	18
1.4 Выводы по обзору	19
2. Проектирование системы оперативной индикации параметров технологического процесса	21
2.1 Архитектура разрабатываемой системы	21
2.2 ПЛК Siemens Simatic S7-200	22
2.3 Микрокомпьютер Raspberry Pi	23
2.4 Операционная система Raspbian	25
2.5 Протокол DHCP	25
2.6 Протокол TCP	26
2.7 Протокол Modbus	28
2.8 Python	29
2.9 Kivy	30
2.10 Buildozer	31
2.11 Вывод к главе 2	31
3. Разработка системы оперативной индикации параметров технологического процесса	33
3.1 Настройка точки доступа Wi-Fi	33
3.2 Подключение к промышленному логическому контроллеру	34
3.3 Создание сервера передачи параметров технологического процесса	39
3.4 Создание клиентского приложения для операционной системы Android	42

3.5	Вывод к главе 3	47
4.	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	48
4.1	Потенциальные потребители результатов исследования	48
4.2	Анализ конкурентных технических решений	49
4.3	SWOT-анализ	51
4.4	Планирование научно-исследовательских работ	52
4.4.1	Структура работы в рамках научного исследования	52
4.4.2	Определение трудоемкости выполнения работ и разработка графика проведения	53
4.4.3	Разработка графика проведения научного исследования.....	56
4.5	Бюджет научно-технического исследования (НТИ).....	56
4.5.1	Расчет материальных затрат НТИ.....	56
4.5.2	Расчет затрат на специальное оборудование для научных (экспериментальных) работ	57
4.5.3	Основная заработная плата исполнителей темы	57
4.5.4	Дополнительная заработная плата исполнителей темы	59
4.5.5	Отчисление во внебюджетные фонды.....	59
4.6	Накладные расходы	60
4.7	Прямые затраты	60
4.8	Формирование бюджета затрат научно-исследовательского проекта1	
4.9	Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	61
	Вывод по разделу финансовый менеджмент	64
5.	Социальная ответственность.....	65
5.1	Правовые и организационные вопросы обеспечения безопасности ..	66
5.1.1	Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства	66
5.1.2	Организационные мероприятия при компоновке рабочей зоны.....	67
5.2	Производственная безопасность	68

5.2.1	Анализ вредных и опасных факторов, которые может создать объект исследования	69
5.2.2	Анализ вредных и опасных факторов, которые могут возникнуть в лаборатории при проведении исследований	71
5.2.3	Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов	72
5.3	Экологическая безопасность	74
5.3.1	Анализ влияния объекта исследования на окружающую среду	74
5.3.2	Анализ влияния процесса исследования на окружающую среду	74
5.3.3	Обоснование мероприятий по защите окружающей среды	75
5.4	Безопасность в чрезвычайных ситуациях.....	75
5.4.1	Анализ вероятных ЧС, которые может инициировать объект исследований	75
5.4.2	Анализ вероятных ЧС, которые могут возникнуть в лаборатории при проведении исследований	76
5.4.3	Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС	76
	Заключение к разделу социальная ответственность	77
	Заключение	78
	Conclusion	80
	Список используемых источников.....	82
	ПРИЛОЖЕНИЕ А Листинг программы сервера	85
	ПРИЛОЖЕНИЕ Б Листинг программы клиента	88
	ПРИЛОЖЕНИЕ В Календарный план-график выполнения проекта.....	97

Обозначения и сокращения

В данной ВКР используются следующие определения и обозначения:

АСУ ТП – автоматизированная система управления технологическим процессом;

ПЛК – программируемый логический контроллер;

SCADA – Supervisory Control And Data Acquisition;

TCP – Transmission Control Protocol;

DHCP – Dynamic Host Configuration Protocol;

ПО – программное обеспечение.

Введение

Система оперативной индикации параметров технологического процесса является неотъемлемой составляющей любого производства, так как благодаря данной системе можно выявить отклонение процесса от заданных пределов, отказы технологического оборудования, и получить информацию о технологическом процессе, также произвести сигнализацию об аварийной ситуации.

На многих производствах даже незначительная задержка в выявлении отказа технологического оборудования может привести к значительным экономическим потерям, либо к чрезвычайной ситуации. Для того, чтобы оперативно предупредить рабочий персонал об отклонении процесса от заданных пределов и аварийной ситуации, индикацию параметров можно осуществлять на смартфоне или смарт часах.

К сожалению, готовые решения до сих пор стоят достаточно дорого. Также существующие системы привязаны к конкретному оборудованию, а иногда входят в состав SCADA, поэтому внедрение их на предприятие требует полного переоборудования производства или переустановки SCADA.

Целью данной выпускной квалификационной работы является создание системы индикации оперативных значений параметров технологического процесса на мобильных устройствах при помощи микрокомпьютера Raspberry Pi. Индикация данной системы должна осуществляться на дисплее, подключенном к микрокомпьютеру, и на смартфоне или смарт часах под управлением операционной системы Android. Подключение к промышленному контроллеру может производиться с помощью основных промышленных протоколов таких как Modbus или Ethernet. Данная система не должна быть привязана к конкретному оборудованию, должна иметь возможность работать параллельно с уже установленной SCADA на предприятии, а также обладать низкой стоимостью. Предприятие, которое планирует использовать данную систему, не должно проводить никакого переоборудования для ее установки.

Для реализации цели данной выпускной квалификационной работы были поставлены следующие задачи:

1. Настройка точки доступа Wi-Fi на микрокомпьютере Raspberry Pi.
2. Подключение к промышленному логическому контроллеру.
3. Создание сервера, который обеспечивает передачу параметров технологического процесса.
4. Создание клиентского приложения для операционной системы Android.

Эта работа включает в себя: изучение теоретического материала о различных промышленных протоколах обмена данными, освоение технических и программных средств для работы с ними, получение навыков настройки серверов на операционных системах семейства Linux, создания приложений для операционной системы Android и разработку программ и мобильных приложений на языке Python.

1 Системы индикации технологических параметров

На данный момент уже существует несколько готовых решений по созданию системы оперативной индикации параметров технологического процесса на мобильные устройства. Однако каждое решение обладает своими преимуществами и недостатками. В данной главе приводится обзор трех решений поставленной цели и задач.

1.1 MasterSCADA 4D

MasterSCADA 4D – это российская вертикально-интегрированная SCADA-система с многоуровневой клиент-серверной архитектурой. MasterSCADA 4D является разработкой компании «ИнСАТ» [1].

Данная система универсальна и используется для автоматизации и диспетчеризации объектов во всех отраслях промышленности. В качестве основной технологии для реализации графического интерфейса в MasterSCADA 4D используется стандарт HTML5. Это позволяет использовать в качестве инструмента графического интерфейса не только графический клиент MasterSCADA 4D, а любое устройство, имеющее в составе программного обеспечения Интернет-браузер, в том числе смартфоны и планшеты [1].

Оперативная индикация параметров технологического процесса в MasterSCADA 4D на мобильные устройства может осуществляться двумя способами: через приложение-клиент для операционной система Android или через облачный сервер.

К недостаткам данного решения для индикации параметров технологического процесса можно отнести:

1. Высокая стоимость. В зависимости от используемого сервера MasterSCADA 4D стоимость клиентского приложения может составлять от 11900 рублей до 10752000 рублей [1].

2. В случае использования облачного сервера для индикации параметров необходимо обеспечивать дополнительные средства кибербезопасности для того, чтобы обеспечить сохранность данных.

Использование MasterSCADA 4D для индикации параметров технологического процесса обладает следующими преимуществами:

1. За счет использования облачных технологий и стандарта HTML5 отсутствует необходимость в установке приложения-клиента на мобильные системы.

2. Широкие графические возможности. В MasterSCADA 4D присутствует выше 200 встроенных графических примитив и изображений [1].

3. Наличие исполнительных модулей под большинство контроллеров, представленных на отечественном рынке.

1.2 SimpLight

SCADA SimpLight фирмы ООО «Симп Лайт» представляет собой программное обеспечение для построения систем управления и диспетчеризации различных автоматизируемых объектов [2].

Основной функционал SCADA SimpLight включает в себя следующие функции:

1. Формирование отчетов.

2. Система оповещений.

3. Разграничение прав доступа.

4. Построение трендов.

5. Наличие встроенных средств написания скриптов (поддерживаются Pascal, C, JavaScript, Vbasic).

6. Редактор мнемосхем. Предназначен для создания и редактирования графического представления внешних устройств, подключенных к OPC серверу.

7. Менеджер проектов.

8. Редактор каналов. Используется для настройки связи с

оборудованием через MODBUS или OPC, для создания скриптов, для конфигурирования реальных и виртуальных тегов-каналов, для авторизации [2].

Данная система поддерживает управление технологическими процессами с мобильных устройств через интернет или локальную сеть и включает в себя мобильное приложение SIMP HMI для контроля промышленных и домашних систем автоматизации [2].

SIMP HMI можно установить на любое устройство с поддержкой операционной системы Android будь то телефон или планшет. Данное приложение поддерживает такие функции как управление объектом локально, отладка объекта прямо на месте, управление через интернет, контроль объекта руководителем [2].

Для того, чтобы управлять или контролировать технологический процесс с мобильного устройства с помощью SIMP HMI необходимо:

1. Загрузить и установить SCADA SimpLight на персональный компьютер.
2. Создать проект в SCADA, то есть создать необходимые связи и мнемосхемы.
3. Загрузить данный проект на мобильное устройство, где установлено приложение SIMP HMI.

К недостаткам данного решения для индикации параметров технологического процесса можно отнести:

1. Высокая стоимость. В зависимости от количества тегов, количества мобильных клиентов, драйверов и дополнительных опций стоимость SCADA SimpLight может составлять от 10000 рублей до 600000 рублей [2].
2. Ограниченность поддерживаемого оборудования. Данная SCADA поддерживает оборудование только определенных производителей, указанных в документации.
3. Связь с контроллером осуществляется только через MODBUS или OPC. В данной системе отсутствует поддержка такие распространенных протоколов как Ethernet или Profibus.

Использование SCADA SimpLight для индикации параметров технологического процесса обладает следующими преимуществами:

1. Возможность подключения к системе индикации параметров не только в локальной сети.
2. Широкие графические возможности.
3. Возможность настраивать и редактировать мнемосхему в мобильном приложении.
4. Отсутствие ограничений на количество используемых мнемосхем.

1.3 Мобильное приложение мониторинга и управления системами АСУ ТП

Разработка фирмы ООО "Электротехнические системы Сибирь" представляет собой техническое решение по мониторингу и управлению системами АСУ ТП, выполненными на базе оборудования MITSUBISHI ELECTRIC, использующее мобильные платформы на устройствах с операционной системой Android и IOS [3].

Индикация параметров организована одним из известных способов, применяющих возможности протокола TCP IP. Используемое программное обеспечение для организации контроля технологическим процессом является частично бесплатным (VNC Client) и определяется мобильной платформой. Для панелей MITSUBISHI ELECTRIC применяется встроенный драйвер VNC Server (версия GT WORKS 3 v1.37) с отдельно приобретаемым ключом [3].

Для работы системы контроля на базе персонального компьютера используется свободно распространяемое ПО.

Организация мониторинга и управления данным способом, позволяет увеличить скорость бизнес процессов, принятия управленческих решений руководителями, отвечающими за бесперебойную работу предприятия вне зависимости от того, где они находятся в данный момент. Инженерно-техническому и обслуживающему персоналу предприятия подобная система

позволит держать под контролем важнейшие параметры производства. Пользователю системы для получения количественных и графических данных требуется минимум операций [3].

Основные решения, использованные при реализации данной системы мониторинга, могут быть адаптированы к оборудованию ведущих производителей систем АСУ ТП.

К недостаткам данного решения для индикации параметров технологического процесса можно отнести:

1. Связь с контроллером только через протокол Ethernet. В данной системе отсутствует поддержка таких распространенных протоколов как MODBUS или Profibus.

2. Ограниченность поддерживаемого оборудования. Данная SCADA поддерживает оборудование только определенных производителей, а именно MITSUBISHI ELECTRIC.

Использование решения фирмы ООО "Электротехнические системы Сибирь" для индикации параметров технологического процесса обладает следующими преимуществами:

1. Использование свободно распространяемого программного обеспечения (VNC Client и VNC Server), что позволяет наращивать количество клиентов без увеличения стоимости.

2. Поддержка операционной системы IOS.

3. Возможность подключения к системе индикации параметров не только в локальной сети.

4. Отсутствие ограничений на количество используемых мнемосхем, тегов и сигналов.

1.4 Выводы к главе 1

Из вышеприведенного обзора существующих технологий и решений следует, что основными недостатками являются:

1. Высокая стоимость.
2. Ограниченность поддерживаемого оборудования.
3. В случае, если на предприятии уже установлены контроллеры других производителей то, для использования данного решения необходимо переоборудование предприятия, что влечет за собой, что влечет за собой остановку процесса и финансовые и материальные потери, связанные с этим.

Также для всех приведенных системах требуется использовать локальный сервер или облачный. В данных системах в качестве локального сервера применяется либо обыкновенная станция оператора, либо специальный серверный компьютер. Такое решение влечет за собой дополнительные траты на покупку оборудования и расходы на его постоянную работу.

2 Проектирование системы оперативной индикации параметров технологического процесса

В данной главе приводится описание основных компонентов и принципов проектируемой системы оперативной индикации параметров технологического процесса, которые позволят устранить все недостатки существующих систем.

2.1 Архитектура разрабатываемой системы

Архитектура разрабатываемой системы представлена на рисунке 1. Данная система оперативной индикации параметров технологического процесса состоит из промышленного контроллера, с которого осуществляется чтение параметров технологического процесса по промышленному протоколу, и сервера, который считывает параметры и отправляет по WiFi на смарт часы или смартфон.



Рисунок 1 – Архитектура разрабатываемой системы

2.2 ПЛК Siemens Simatic S7-200

В данной работе для проверки работоспособности системы оперативной индикации параметров технологических процессов используется ПЛК Siemens Simatic S7-200.

Simatic S7-200 – семейство программируемых логических контроллеров фирмы Siemens AG из семейства устройств автоматизации Simatic S7. Микроконтроллеры Simatic S7-200 предназначены для решения задач регулирования и управления в небольших системах автоматизации. При этом Simatic S7-200 позволяют создавать и автономные системы управления, и системы управления, работающие в общей информационной сети. Контроллеры Simatic S7-200 позволяют решать как простейшие задачи автоматизации, для решения которых в прошлом использовались контакторы и простые реле, так и задачи комплексной автоматизации. Simatic S7-200 также используется при создании таких систем управления, для которых в прошлом из соображений экономии необходимо было разрабатывать специальные электронные модули [4].

Программируемые контроллеры S7-200 отвечают требованиям следующих международных и национальных стандартов:

1. Сертификат соответствия требованиям ГОСТ.
2. Метрологический сертификат.
3. Сертификат ISO 9001.

Взаимодействие с устройствами системы управления контроллеры семейства S7-200 осуществляют с помощью коммуникационных модулей и встроенных интерфейсов RS-485.

Контроллеры этого семейства используют для организации связи:

1. MPI интерфейс, в котором S7-200 способен выполнять только функции ведомого устройства.

2. сеть Profibus-DP, в которой S7-200 выполняет только функции ведомого устройства.
3. ASI-интерфейс, в котором контроллеры S7-200 выполняют функции ведущего устройства сети.
4. PPI-интерфейс, работающий в сканирующем или свободно программируемом режимах.
5. Протокол Modbus, в котором S7-200 способен выполнять функции ведомого и ведущего устройства.
6. Протокол Ethernet [4].

2.3 Микрокомпьютер Raspberry Pi

Для того, чтобы передавать параметры технологического процесса необходимо разработать сервер, который будет собирать параметры с контроллера и передавать на мобильные устройства. В качестве устройства, на котором будет работать сервер, был выбран микрокомпьютер Raspberry Pi.

Raspberry Pi – одноплатный компьютер компактного размера. Имеет интерфейс HDMI для подключения монитора, GPIO разъём для подключения низкоуровневой периферии, USB-порты для подключения USB устройств, и Ethernet-порт для подключения к сети. Самой мощной моделью на момент написания работы является Raspberry Pi 3B. Она имеет встроенный Bluetooth 4 и Wi-Fi, 1GB оперативной памяти и ядерный 64-битный процессор ARM 1.2Ghz [5].

К альтернативам Raspberry Pi можно отнести одноплатные компьютеры Orange Pi и Banana Pi.

Основными недостатками Orange Pi являются необходимость в оснащении устройства дополнительным охлаждением и отсутствие WiFi. В свою очередь Banana Pi в 2 раза превосходит по стоимости Raspberry Pi. Также для Orange Pi и Banana Pi нет такого перечня адаптированных программ и пакетов, как для Raspberry Pi [6].

Чтобы держать сервер для передачи технологических параметров, от компьютера не требуется высокая производительность, которая влечет за собой немалые размеры и цену, поэтому, для этой задачи был выбран одноплатный компьютер Raspberry Pi. В сравнении с конкурентами компьютерами Raspberry Pi имеет неплохую производительность и низкую стоимость, а также очень примечательна его стабильная работа и надежность. Кроме того, он имеет очень низкую энергопотребляемость, поэтому данный микрокомпьютер отлично подходит для передачи технологических параметров [6].

Благодаря тому, что Raspberry Pi оснащен портами GPIO существует возможность подключения различных адаптеров и конвертеров, которые увеличат количество интерфейсов и возможностей для подключения к ПЛК.

Также данный микрокомпьютер оснащен встроенным модулем Wi-Fi благодаря этому имеется возможность использования его в качестве точки доступа. Если расстояния, на которое передается Wi-Fi сигнал будет недостаточно, то имеется возможность подключения различных усилителей сигнала.

Так как Raspberry Pi 3B обладает портом HDMI, существует возможность подключения монитора, чтобы отображать параметры технологического процесса и использовать его как станцию оператора.

В данной работе используется модель Raspberry Pi 3B (рисунок 2), однако в дальнейшем она может быть заменена на более дешёвую и энергоэффективную модель Raspberry Pi Zero.



Рисунок 2 – Raspberry Pi

2.4 Операционная система Raspbian

Микрокомпьютер Raspberry Pi работает под управлением операционной системы. Так же, как и в обычных компьютерах, операционная система управляет всеми ресурсами компьютера и предоставляет пользователю определенный интерфейс для работы с программами. От того, какая выбрана система, зависит удобство использования устройства.

В данной работе для Raspberry Pi используется операционная система Raspbian.

Raspbian – основанная на Debian операционная система для Raspberry Pi. Существует несколько версий Raspbian, в том числе Raspbian Stretch и Raspbian Jessie. В 2015 году Raspbian официально была представлена Raspberry Pi Foundation как основная операционная система для одноплатных компьютеров Raspberry Pi. Данная операционная система находится в стадии активной разработки. Так как в линейке компьютеров Raspberry Pi используются низкопроизводительные процессоры ARM, Raspbian оптимизирована специально для них [7].

Оболочка PIXEL (Pi Improved Xwindows Environment, Lightweight) используется в качестве основной среды рабочего стола в Raspbian. Она состоит из менеджера окон Openbox с новой темой и несколькими другими изменениями и модифицированной среды рабочего стола LXDE [7].

Также Raspbian содержит несколько предустановленных программ для обучения и программирования такие, как Python, Scratch, Sonic Pi, Java Mathematica и другие [7].

2.5 Протокол DHCP

Для того, чтобы создать точку доступа Wi-Fi необходимо создать сервер, работающий по протоколу DHCP.

DHCP (Dynamic Host Configuration Protocol) – протокол динамической настройки узла [7].

DHCP предназначен для настройки хоста через удаленный сервер и позволяет сетевым устройствам автоматически получать IP-адрес и другие параметры. DHCP поддерживается по умолчанию большинством современных операционных систем, и этот протокол выступает в качестве альтернативы ручной настройки параметров сети для сервера или подключаемого устройства. Сетевой администратор может задать диапазон адресов, распределяемых сервером среди компьютеров. Это позволяет избежать ручной настройки компьютеров сети и уменьшает количество ошибок [8].

Также использование данного протокола позволит увеличить контроль за подключаемыми и отключаемыми устройствами.

2.6 Протокол TCP

В данной работе передача данных технологических параметров мобильным устройствам осуществляется по протоколу TCP [9].

TCP, сокращенно от Transmission Control Protocol – протокол, определяющий как устанавливать и поддерживать связь, через которую две программы могут обмениваться данными. Данный протокол обеспечивает сквозную доставку данных между прикладными процессами, запущенными на узлах, взаимодействующих по сети. Выполняет функции транспортного уровня модели OSI [8]. У протокола две основные задачи:

1. Разбивать сообщения или файлы на более мелкие части (называемые пакетами данных), которые затем отправляются через Интернет. После, эти пакеты обрабатываются другим уровнем TCP, который собирает данные в полный файл.

2. Проверка потока данных на наличие ошибок для обеспечения доставки данных. Если ошибка будет обнаружена, TCP повторно передаст пакеты [7].

Помимо этого, TCP обеспечивает:

1. Надёжную доставку сегментов. Надёжная доставка подразумевает автоматическую повторную пересылку недошедших сегментов. Каждый сегмент маркируется при помощи специального поля – порядкового номера (sequence number). После отправки некоторого количества сегментов TCP на отправляющем узле ожидает подтверждения от получающего, в котором указывается порядковый номер следующего сегмента, который адресат желает получить. В случае, если такое подтверждение не получено, отправка автоматически повторяется. После некоторого количества неудачных попыток, TCP считает, что адресат не доступен, и сессия разрывается [9].

2. Упорядочивание сегментов при получении. Часто пакеты с целью распределения нагрузки могут идти по сети разными путями, с разной скоростью, через разные промежуточные устройства. Таким образом получатель, декапсулировав их, может получить сегменты не в том порядке, в котором они отправлялись. TCP автоматически собирает их в нужной последовательности, используя поле порядковых номеров, и передает после соединения на уровень приложений [9].

3. Работу с сессиями. Перед началом передачи данных, TCP должен убедиться в том, что получатель существует, слушает нужный отправителю порт и готов принимать. Далее, в рамках сессии передаются полезные пользовательские данные. После завершения передачи сессия закрывается, тем самым получатель извещается о том, что данных больше не будет, а отправитель извещается о том, что получатель извещён [9].

4. Контроль за скоростью передачи. Контроль за скоростью передачи позволяет изменить скорость отправки данных в зависимости от возможностей получателя. Например, если сервер с быстрой скоростью передачи отправляет данные клиенту с низкой скоростью, то сервер будет передавать данные с приемлемой для клиента скоростью [9].

Работа TCP состоит из трех этапов:

1. Установка соединения.

2. Передача данных.
3. Завершение соединения.

2.7 Протокол Modbus

Raspberry Pi имеет возможность подключения к контроллерам по протоколам Modbus, благодаря открытым библиотекам, таким как PyModbus, MinimalModbus, Modbus-ТК, обеспечивающим обмен данными по данным протоколам и различным переходникам.

Modbus – открытый коммуникационный протокол, основанный на архитектуре ведущий-ведомый (master-slave). Широко применяется в промышленности для организации связи между электронными устройствами [9].

К альтернативам Modbus можно отнести протокол Profibus, однако на данный момент для работы с ним написано мало библиотек и все они находятся на ранних стадиях разработки. Также данный протокол не так распространен, так как является разработкой фирмы Siemens и применяется только в контроллерах данной фирмы [10].

Подключение к ПЛК по протоколу Modbus может осуществляться через один из четырех входов USB или интерфейсы GPIO посредством специальных адаптеров и переходников (например, рисунок 3).



Рисунок 3 – Адаптер USB-RS485

Протокол Modbus разработан для использования в программируемых логических контроллерах. В настоящее время является одним из самых распространенных протоколов, применяемых в различных промышленных системах. Для передачи данных используются последовательные линии связи, основанные на интерфейсах RS-232, RS-422, RS-485. В начале развития применялся интерфейс RS-232, как один из наиболее простых промышленных интерфейсов для последовательной передачи данных. В настоящее время протокол часто использует интерфейс RS-485, так как он обладает высокой скоростью передачи, позволяет передавать данные на большие расстояния и объединять несколько устройств в единую сеть, благодаря тому, что протокол Modbus поддерживает адресацию. Из-за простоты, надежности и возможности легко интегрировать устройства, поддерживающие Modbus, в единую сеть данный протокол получил широкое распространение [11].

В сети не может находиться более одного ведущего устройства (master), что является главной особенностью протокола. Только ведущее устройство может опрашивать остальные устройства сети, которые являются ведомыми (slave). Так как работа сети строится только по принципу "запрос-ответ", подчиненное устройство не может самостоятельно инициировать передачу данных или запрашивать какие-либо данные у других устройств. Ведущее устройство также может подать широковещательный запрос, адресованный всем устройствам в сети, в таком случае ответное сообщение не посылается [11].

2.8 Python

Python – один из самых известных и используемых языков программирования.

Python выступает в роли основного языка программирования для Raspberry Pi. Пакет Python уже установлен в операционной системе Raspbian в двух версиях – 2 и 3.

Несмотря на то, что Raspberry Pi поддерживает и другие языки программирования, управление портами GPIO, через которые может осуществляться подключение к ПЛК, осуществляется только с помощью языка Python.

Сам язык программирования Python характеризуется простым синтаксисом и высокой функциональностью, поддерживает автоматическое управление памятью, многопоточность вычислений, обработку исключений. Для этого языка существует большое количество библиотек для самых разнообразных сфер применения [13]. Для работы с протоколом Modbus – это библиотеки PyModbus, Modbus-TK, MinimalModbus. Для работы с протоколом TCP – библиотеки socket, socketserver и twisted.

2.9 Kivy

Так как язык Python не содержит встроенных средств для создания приложений для операционной системы Android, поэтому для создания интерфейса программы отображения параметров технологического процесса и создания приложения для операционной системы Android используется Kivy.

Kivy – библиотека с открытым исходным кодом Python для быстрой разработки приложений, которые используют пользовательские интерфейсы. Также данная библиотека поддерживает кросс-платформенность, то есть один и тот же программный код можно запустить на операционных системах Linux, Windows, OS X, Android и iOS [14].

Единственным аналогом Kivy на python является библиотека BeeWare Project. Однако на данный момент в связи с тем, что BeeWare Project – это относительно недавняя разработка, для нее отсутствует большое количество документации и поддержка сторонних разработчиков как у Kivy. Также BeeWare Project уступает Kivy в стабильности [14].

2.10 Buildozer

Несмотря на все свои достоинства у Python отсутствует возможность создания исполняемых файлов и пакетов установки написанных программ. Для решения данной проблемы в работе используется специальная утилита – Buildozer.

Инструмент Buildozer используется для упаковки проекта в качестве приложения для Android. После установки Buildozer автоматизирует процесс создания приложения. Buildozer готовит среду в соответствии со всеми требованиями Android SDK и NDK. В настоящее время Buildozer поддерживает создание пакетов для операционных систем Android и IOS [15].

2.11 Выводы к главе 2

Благодаря тому, что в разрабатываемой системе в качестве сервера передачи данных используется одноплатный микрокомпьютер Raspberry Pi с использованием энергоэффективных ядер семейства A55, обеспечивается низкая энергопотребляемость. Так как операционная система микрокомпьютера основана на Debian, она является более безопасной по отношению к Windows из-за меньшего количества вредоносных программ ввиду меньшей популярности. Также данный микрокомпьютер обеспечивает низкую стоимость относительно специальных серверных компьютеров.

Использование промышленных протоколов Modbus и Ethernet обеспечивает совместимость с различными типами промышленного оборудования.

Так как для разработки приложения для оперативной индикации параметров технологического процесса используется библиотека Kivy, обеспечивается совместимость с основными операционными системами, что позволяет запускать приложение как на мобильных устройствах, так и на микрокомпьютере.

3 Разработка системы оперативной индикации параметров технологического процесса

В данной главе приводятся методы и этапы разработки системы и описание проблем, возникнувших в результате работы.

3.1 Настройка точки доступа Wi-Fi

Raspberry Pi можно использовать в качестве беспроводной точки доступа с автономной сетью. Raspberry Pi 3 имеет встроенный модуль Wi-Fi, поэтому отсутствует необходимость в подключении дополнительных устройств. В случае, если дальности действия встроенного модуля недостаточно, имеется возможность подключения USB-адаптеров для ее увеличения. Точка доступа требовалась для создания локальной сети. Данная локальная сеть использовалась для передачи параметров технологического процесса мобильному клиенту с сервера.

На микрокомпьютере была установлена операционная система Raspbian без пользовательского интерфейса. Данное решение позволило повысить производительность Raspberry Pi и уменьшить количество памяти, используемой операционной системой. Однако в данном дистрибутиве Raspbian отсутствовали средства автоматической настройки Wi-Fi сети, поэтому чтобы создать точку доступа на Raspberry Pi, было установлено программное обеспечение точки доступа. Так как автономная сеть настраивается в качестве сервера, Raspberry Pi был назначен статический IP-адрес и интерфейс беспроводного порта. Чтобы настроить статический IP-адрес, был отредактирован файл конфигурации `dhcpcd.conf` и установлены следующие параметры:

1. `interface = wlan0;`
2. `static ip_address=192.168.4.1.`

Затем было установлено программное обеспечение DHCP-сервера для назначения подключаемым устройствам сетевого адреса [16]. После этого был

отредактирован файл конфигурации `dnsmasq.conf` и установлены следующие параметры:

1. `interface = wlan0;`
2. `dhcp-range=192.168.4.2,192.168.4.50;`

Затем был отредактирован файл `hostapd.conf` и установлены следующие параметры:

1. `interface=wlan0;`
2. `driver=nl80211;`
3. `ssid=NameOfNetwork;`
4. `hw_mode=g;`
5. `channel=7;`
6. `wmm_enabled=0;`
7. `macaddr_acl=0;`
8. `auth_algs=1;`
9. `ignore_broadcast_ssid=0;`
10. `wpa=2;`
11. `wpa_passphrase=AardvarkBadgerHedgehog;`
12. `wpa_key_mgmt=WPA-PSK;`
13. `wpa_pairwise=TKIP;`
14. `rsn_pairwise=CCMP.`

Для применения данной конфигурации DHCP-сервер был перезапущен.

После запуска утилит `hostapd` и `dnsmasq` настроенная Wi-Fi сеть стала доступна для подключения при введении указанного пароля.

3.2 Подключение к промышленному логическому контроллеру

Используемый в данной работе контроллере Siemens S7-200 оснащен двумя портами RS485, однако только порт 0 может осуществлять передачу данных по протоколу Modbus.

В ходе выполнения работы обнаружилось, что назначения контактов портов контроллера отличаются от стандартного интерфейса RS485. Более того в официальной документации для Siemens S7-200 было обнаружено, что контакты, отвечающие за прием и передачу информации, приведены неправильно. Для правильного подключения их требовалось поменять местами. Поэтому для подключения к ПЛК был сделан специальный кабель, который учитывал назначения контактов порта контроллера.

Также было обнаружено, что большинство адаптеров для подключения к интерфейсу RS485 требуют внешнего питания в 12 или 15 В, но ни микрокомпьютер, ни контроллер не способны были давать такое напряжение. Поэтому для подключения был использован адаптер USB-RS485 на основе микросхемы MAX485, которому для работы достаточно 5 В. Данный адаптер имеет 2 контакта для передачи данных (А и В), которые необходимо было подключить к контактам порта 0 в соответствии с рисунком 4. Питание адаптера осуществляется от USB порта микрокомпьютера Raspberry Pi [17].

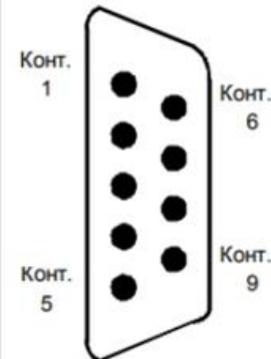
Разъем	Номер контакта	Порт 0/Порт 1
	1	Заземляющий контакт
	2	Общий провод логики
	3	RS-485, сигнал А
	4	RTS (TTL)
	5	Общий провод логики
	6	+5 В, последовательно включенный резистор 100 Ом
	7	+24 В
	8	RS-485, сигнал В
	9	10-битовый протокол (вход)
	Корпус разъема	Заземляющий контакт

Рисунок 4 – Назначение контактов коммуникационного порта S7-200

Для работы с контроллерами Siemens S7-200 используется программное обеспечение STEP 7-Micro/WIN. Для применения протокола Modbus необходима специальная библиотека Modbus Slave или Modbus Master. Данные библиотеки в зависимости от комплектации контроллера уже входят в состав STEP 7-Micro/WIN или приобретаются отдельно. Библиотека Modbus Slave позволяет

работать контроллеру в качестве ведомого устройства, а Modbus Master – в качестве ведущего [17].

В данной работе Siemens S7-200 выступает в качестве ведомого устройства (Slave), а микрокомпьютер Raspberry Pi в качестве ведущего (Master). Такой выбор режимов работы устройств обусловлен тем, что Modbus Master способен получать данные одновременно с нескольких контроллеров, работающих в режиме Modbus Slave. Это позволяет подключать одновременно несколько ПЛК к микрокомпьютеру и считывать с них параметры [17].

Для активации протокола Modbus и разрешения считывания параметров в контроллер была загружена программа, реализованная на языке LD, изображенная на рисунке 5.

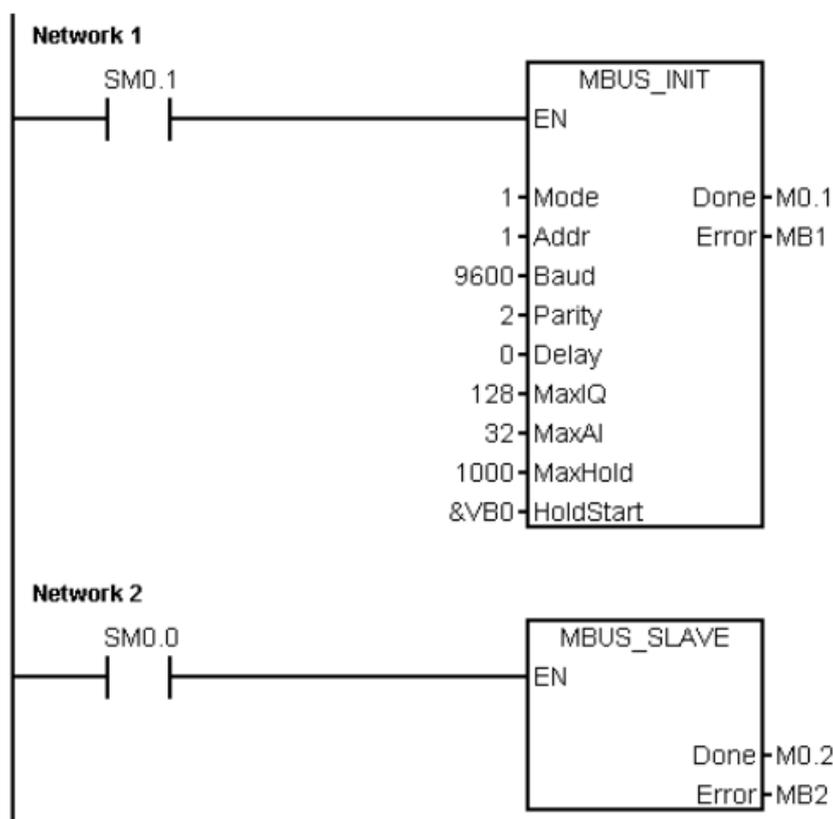


Рисунок 5 – Программа, активирующая протокол Modbus для S7-200

Команда MBUS_INIT используется для активизации и инициализации или деактивизации обмена данных через протокол Modbus. Прежде чем сможет быть применена команда MBUS_SLAVE, должна быть без ошибок выполнена команда MBUS_INIT. При завершении этой команды немедленно устанавливается бит Done (Готово), Команда выполняется в каждом цикле, когда

включен вход EN. Команда MBUS_INIT должна выполняться ровно один раз для каждого изменения состояния обмена данными. Поэтому вход EN должен включаться импульсно с помощью элемента обнаружения фронта или исполняться только в первом цикле [17].

Для подключения к микрокомпьютеру были установлены следующие параметры соединения:

1. Скорость передачи 9600 Бод.
2. Установлена проверка на четность.
3. Задержка отсутствует.
4. Разрешено обращение ко всем цифровым входам и выходам S7-200.
5. Разрешено обращение ко всем аналоговым входам S7-200.
6. Разрешен доступ к 1000 регистрам временного хранения информации, начиная с регистра 0.

Ключевым параметром является проверка на четность, поскольку при ее отключении или включении проверки на нечетность, прочитать параметры с ПЛК не удалось. Остальные параметры могут принимать другие значения в зависимости от поставленной задачи.

Команда MBUS_SLAVE используется для обслуживания запроса от master-устройства Modbus и должна выполняться в каждом цикле, чтобы дать ей возможность контролировать и реагировать на запросы Modbus. Команда выполняется в каждом цикле, когда включен вход EN [17].

Считывать значения параметров с контроллера Siemens S7-200 можно только через специальное программное обеспечение STEP 7-Micro/WIN, однако данное ПО разработано только для операционных систем Windows. Поэтому после того, как программа была загружена в ПЛК, было необходимо разработать собственную программу для работы по протоколу Modbus для Raspberry Pi.

Основными библиотеками Python для использования протокола Modbus являются PyModbus, Modbus-TK, MinimalModbus. Данные библиотеки были установлены на Raspberry Pi с помощью специальных программ pip и pip3.

Для подключения и считывания параметров с Siemens S7-200 была написана программа с использованием библиотеки PyModbus, приведенная в таблице 1, на основе официальной документации к данной библиотеке. Для подключения в данной программе указаны такие параметры соединения, что и в программе, загруженной в контроллер. Также указан USB порт, к которому подключен адаптер USB-RS485. Затем выполняется запрос на чтение 10 регистров временного хранения информации, начиная с регистра 0.

Таблица 1 – Листинг тестовой программы

```
from pymodbus.client.sync import ModbusSerialClient as ModbusClient
from pymodbus.register_read_message import ReadInputRegistersResponse

client = ModbusClient(method='rtu', port='/dev/ttyUSB0', stopbits=1,
                      bysize=8, parity = 'E', baudrate = 9600, timeout=0)
connection=client.connect()
print(connection)

value = client.read_holding_registers(0, 10 ,unit=0x01)
print(value.registers)
```

При запуске данной программы были успешно получены на микрокомпьютере значения 10 регистров временного хранения информации с контроллера Siemens S7-200, начиная с регистра 0.

После успешного подключения к контроллеру было произведено сравнение библиотек PyModbus, Modbus-TK, MinimalModbus.

Библиотека PyModbus не обеспечивала стабильного подключения к контроллеру, так как некоторые запросы, посылаемые данной библиотекой, не возвращали никакого результата.

Библиотека MinimalModbus смогла обеспечить стабильное подключение к контроллеру, однако один запрос на чтение не мог содержать больше 10 параметров. С учетом того, что скорость, с которой контроллер Siemens S7-200

отправляет ответы на запрос, не превышает 4 запросов в секунду, ограничение в 10 параметров на 1 запрос позволит считывать не более 40 параметров в секунду.

Библиотека Modbus-ТК смогла обеспечить стабильное подключение к контроллеру, при этом один запрос на чтение не мог содержать больше 120 параметров. С учетом того, что скорость, с которой контроллер Siemens S7-200 отправляет ответы на запрос, не превышает 4 запросов в секунду, ограничение в 120 параметров на 1 запрос позволит считывать не более 480 параметров в секунду. Однако контроллер Siemens S7-200 способен обработать не более 75 параметров в 1 запросе на чтение.

Таким образом, Modbus-ТК способна считывать наибольшее количество параметров за 1 запрос и обеспечивает стабильное подключение к ПЛК, поэтому в дальнейшей работе при создании сервера и повторном подключении к контроллеру Siemens S7-200 будет использоваться именно она.

3.3 Создание сервера передачи параметров технологического процесса

Одной из главных частей системы оперативной индикации параметров технологического процесса является сервер, основной функцией которого является передача параметров технологического процесса, полученных с ПЛК, мобильному устройству. Данный сервер был создан на языке программирования Python с использованием открытой библиотеки socketserver, которая позволяет осуществлять передачу данных по протоколу TCP и входит в перечень стандартных библиотек Python.

Полный листинг программы сервера передачи параметров технологического процесса приведен в Приложении А.

При одновременном подключении нескольких клиентов, формировалось несколько одновременных запросов на чтение параметров с контроллера, что приводило к ошибке, так как контроллер не может обрабатывать несколько

запросов одновременно. Для решения данной проблемы были использованы потоки, работающие в асинхронном режиме.

Поток – это виртуальный процессор, имеющий свой собственный набор регистров и стек, аналогичных регистрам настоящего центрального процессора, при этом другие потоки также могут их использовать [18].

При подключении нового клиента создается новый поток, который обрабатывает запрос на чтение параметров только этого клиента независимо от остальных потоков (рисунок 6). При отключении клиента поток, отвечающий за обработку его запросов, уничтожается. Благодаря тому, что потоки работают в асинхронном режиме, запросы на чтение формируются не одновременно, и благодаря этому никаких ошибок не возникает.

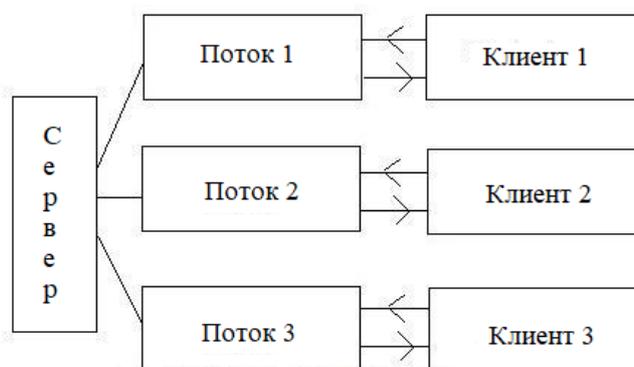


Рисунок 6 – Принцип работы потоков

Согласно протоколу Modbus существует 4 типа параметров: цифровые выходы, цифровые входы, регистры аналоговых входов, регистры временного хранения информации [17].

Для того, чтобы получить значения с сервера параметров технологического процесса, клиент должен отправить запрос, который состоит из 3 значений:

1. Тип параметра. Представляет собой целое число от 0 до 4 (0 - цифровые выходы, 1 - цифровые входы, 3 - регистры аналоговых входов, 4 - регистры временного хранения информации).
2. Начальный адрес параметра.
3. Количество параметров.

В случае, если количество параметров больше 75, то для того, чтобы получить их, сервер формирует несколько запросов к контроллеру. Количество параметров в каждом запросе не превышает 75. Затем полученные с контроллера параметры, принадлежащие одному запросу клиента, объединяются и отправляются.

В ходе выполнения работы было обнаружено, что, если нескольким клиентам требуется получить одни и те же параметры технологического процесса, сервер и контроллер формировали и обрабатывали несколько одинаковых запросов. С учетом того, что контроллер не может обрабатывать более 4 запрос в секунду, это значительно тормозило работу всей системы. Для решения данной проблемы в сервере реализована возможность кэширования параметров технологического процесса.

Принцип работы кэширования приведен на рисунке 7.

При получении сервером запроса от клиента, сервер проверяет наличие данного запроса в списке всех запросов, полученных сервером за последнюю секунду. Если данный запрос не обнаружен, то сервер формирует новый запрос на чтение параметров с контроллера и сохраняет запрос клиента и параметры, удовлетворяющие данному запросу, в списки. В случае если запрос найден в списке всех запросов, полученных сервером за последнюю секунду, то клиенту отправляются параметры, сохраненные при обработке этого запроса. Сохраненные запросы и параметры удаляются по истечении 1 секунды.

Применение возможности кэширования позволило увеличить скорость работы системы в 4 раза.

Также было обнаружено, что библиотека `socketserver` передает и принимает данные только в виде байтовых строк, поэтому реализована функция автоматического перевода в строковый тип, а затем в последовательность байтов перед отправкой данных. Принятые запросы от клиента, наоборот, переводятся в целочисленные значения.

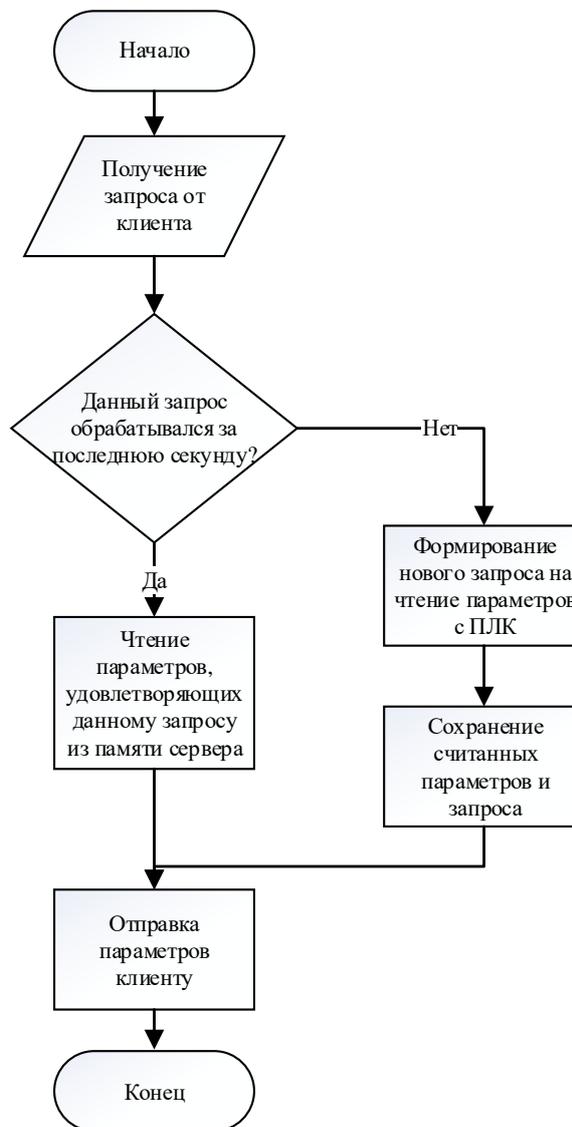


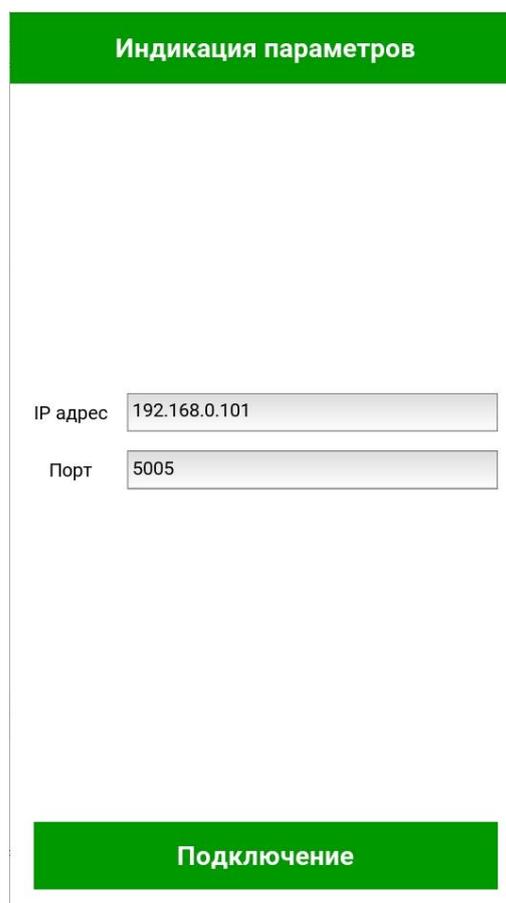
Рисунок 7 – Блок-схема кэширования параметров

3.4 Создание клиентского приложения для операционной системы Android

После того, как был написан программный код сервера, необходимо было написать программу, которая бы получала и считывала полученные значения, отправленные с сервера. Для создания интерфейса была использована библиотека Kivy, а для установления обмена данными использовалась стандартная библиотека socket.

Полный листинг программы сервера передачи параметров технологического процесса приведен в Приложении Б.

При запуске приложения появляется окно подключения, изображенное на рисунке 8, где пользователю необходимо указать IP адрес и порт сервера.



The image shows a window titled "Индикация параметров" (Parameter Indication) with a green header. Inside the window, there are two input fields. The first is labeled "IP адрес" (IP address) and contains the text "192.168.0.101". The second is labeled "Порт" (Port) and contains the text "5005". At the bottom of the window, there is a green button with the text "Подключение" (Connection).

Рисунок 8 – Окно подключения к серверу

После успешного подключения к серверу, открывается окно индикации параметров технологического процесса, изображенное на рисунке 9. Данное окно отображает параметры выбранной группы. Выбор группы осуществляется по четырем вкладкам снизу. Значения параметров обновляются автоматически каждую секунду. Так как значения, получаемые библиотекой socket, являются одной байтовой строкой, перед отображением происходит ее декодирование в обычную строку, а затем разбиение на отдельные параметры.

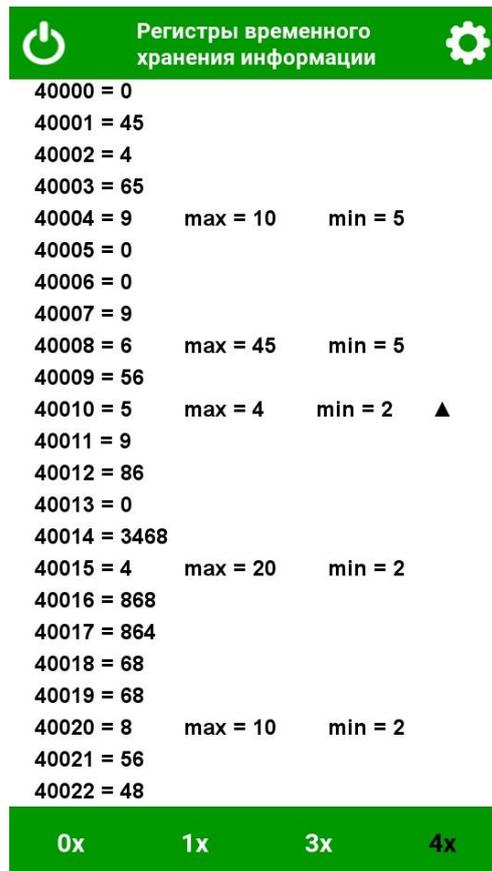


Рисунок 9 – Окно индикации параметров технологического процесса
 При нажатии на значение параметра возникает окно настройки сигнализации, изображенное на рисунке 10.

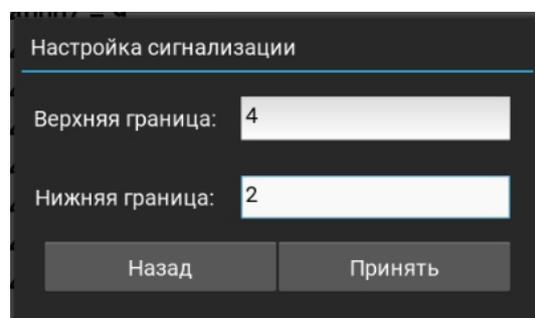


Рисунок 10 – Окно настройки сигнализации

В данном окне пользователь может задать верхнюю и нижнюю границу параметра. Если значение параметра выйдет за указанные пределы, то возникнет окно сигнализации, изображенное на рисунке 11, а после закрытия окна, напротив значения будет отображаться стрелка, указывающая на превышение или понижения значения.

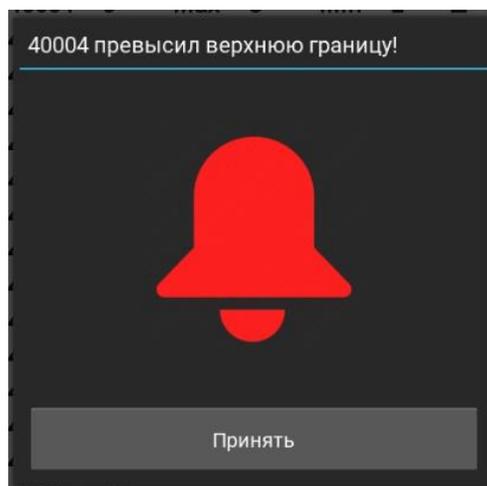


Рисунок 11 – Окно сигнализации

Также в данном приложении реализовано окно настройки, в которое можно перейти нажатием по кнопке в правом верхнем углу окна отображения параметров. Данное окно изображено на рисунке 12. Здесь пользователь может задать количество параметров каждого типа и начальный адрес.

Настройки

Цифровые выходы:

Количество

Адрес

Цифровые входы:

Количество

Адрес

Аналоговые входы:

Количество

Адрес

Регистры временного хранения информации:

Количество

Адрес

Рисунок 12 – Окно настройки

После написания программного кода необходимо было создать пакет приложения для операционной системы Android. Для этого была использована специальная утилита Buildozer. После запуска программы был сформирован

файл конфигурации, где необходимо было указать название программы, режим работы, режим отображения (портретный) и требования к устройству. В требованиях к устройству были указаны все используемые библиотеки, а именно: kivy, socket, time, os. Также в файле конфигурации необходимо было указать разрешение на доступ в интернет.

После данной конфигурации была произведена компиляция пакета программы. Утилита Buildozer автоматически скачивает набор средств разработки android SDK при компиляции приложения.

Однако была обнаружена проблема, что в утилите Buildozer отсутствуют инструкции, как компилировать приложения, использующие библиотеку socket, поэтому при запуске возникали ошибки. Для решения данной проблемы была найдена библиотека Twisted, которая позволяет принимать и отправлять данные по протоколу TCP. Для данной библиотеки утилита Buildozer содержала инструкции по компиляции, поэтому разработанное приложение было переделано уже с использованием найденной библиотеки и заново скомпилировано.

После компиляции приложения был получен установочный файл apk, который был загружен и установлен на смартфон. После установки программы на устройство под управлением системы Android и ее запуска было осуществлено подключение к серверу и успешно получены параметры технологического процесса.

Таким образом, разработанное приложение было успешно протестировано и в его работе не наблюдалось никаких сбоев и ошибок.

3.5 Вывод к главе 3

В ходе разработки системы была создана точка доступа на микрокомпьютере Raspberry Pi для создания локальной сети. Данная локальная сеть использовалась для передачи параметров технологического процесса мобильному клиенту с сервера.

Также было осуществлено подключение к контроллеру Siemens S7-200 через адаптер USB-RS485 и специальные библиотеки для работы с протоколом Modbus. Было произведено исследование основных библиотек Python, с помощью которых осуществлялось чтение параметров. Для дальнейшей работы была выбрана Modbus-TK, так как она обеспечивает стабильное подключение к ПЛК и способна считывать до 120 параметров за 1 запрос.

Помимо этого, был разработан сервер передачи параметров технологического процесса, который способен создавать отдельные потоки для каждого клиента и поддерживает кэширование считанных параметров.

В завершении было разработано мобильное приложение для операционной системы Android. В данном приложении реализованы возможности настройки сигнализации и выбора параметров, которые необходимо получить.

Разработанная система была успешно протестирована на контроллере Siemens S7-200 с передачей параметров на смартфон, в ее работе никаких сбоев и ошибок не наблюдалось.

4 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является проектирование и создание конкурентоспособных разработок, технологий, отвечающих современным требованиям в области ресурсоэффективности и ресурсосбережения.

Достижение цели обеспечивается решением задач:

1. оценка коммерческого потенциала и перспективности проведения научных исследований;
2. определение возможных альтернатив проведения научных исследований, отвечающих современным требованиям в области ресурсоэффективности и ресурсосбережения;
3. планирование научно-исследовательских работ;
4. определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.

4.1 Потенциальные потребители результатов исследования

Потенциальными потребителями результатов исследований являются предприятия, на которых установлена или внедряется автоматизированная система управления технологическим процессом и необходим постоянный контроль за производством.

В таблице 2 приведена сегментация рынка по следующим критериям: размер компании-заказчика и направление деятельности. Анализ рынка был проведен на основе фирм-производителей ООО "Электротехнические системы Сибирь" (фирма «А»), ООО «Симп Лайт» (фирма «Б»), АО «ЭлеСи» (фирма «В»).

Таблица 2 – Карта сегментирования

Критерий		Направление деятельности		
		Мониторинг АСУ ТП на мобильных платформах	Разработка SCADA систем	Внедрение SCADA систем
Размер	Крупные	А	В	В
	Средние	Б	А,В	А
	Мелкие		Б	Б

Таким образом, из карты сегментирования видно, что сегмент рынка мониторинга АСУ ТП на мобильных платформах для мелких предприятий остается свободным.

4.2 Анализ конкурентных технических решений

Анализ приведен в таблице 3. Конкурентами для разрабатываемой системы являются SCADA SimpLight (Конкурент 1) и разработка системы Мониторинг и управление системами АСУ ТП на мобильных платформах с ОС Android фирмы ООО "Электротехнические системы Сибирь" (Конкурент 2).

SCADA SimpLight фирмы ООО «Симп Лайт» представляет собой программное обеспечение для построения систем управления и диспетчеризации различных автоматизируемых объектов. Данная система поддерживает управление технологическими процессами с мобильных устройств через интернет или локальную сеть и включает в себя мобильные приложения для контроля промышленных и домашних систем автоматизации.

Разработка фирмы ООО "Электротехнические системы Сибирь" представляет собой техническое решение по мониторингу и управлению системами АСУ ТП, выполненными на базе оборудования MITSUBISHI ELECTRIC, использующее мобильные платформы на устройствах Android, iPhone, iPad.

Таблица 3 – Оценочная карта для сравнения конкурентных технических решений

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Разработка	Конкурент 1	Конкурент 2	Разработка	Конкурент 1	Конкурент 2
Технические критерии оценки ресурсоэффективности							
1.Поддерживаемое оборудование	0,15	5	3	3	0,77	0,46	0,46
2. Удобство в эксплуатации	0,01	4	4	4	0,05	0,05	0,05
3.Помехоустойчивость	0,05	4	5	4	0,21	0,26	0,21
4.Энергоэкономичность	0,12	5	4	4	0,58	0,46	0,46
5.Надежность	0,10	3	4	4	0,31	0,41	0,41
6.Безопасность	0,06	4	5	5	0,26	0,32	0,32
7.Потребность в ресурсах памяти	0,04	5	4	5	0,19	0,15	0,19
8.Предоставляемые возможности	3	4	5	0,38	0,51	0,64	3
Экономические критерии оценки эффективности							
1.Конкурентоспособность	0,08	3	5	5	0,23	0,38	0,38
2.Цена	0,14	5	4	3	0,71	0,56	0,42
3.Предполагаемый срок эксплуатации	0,09	4	4	4	0,36	0,36	0,36
4.Послепродажное обслуживание	0,03	3	5	5	0,08	0,13	0,13
Итого	1	48	51	51	4,12	4,06	4,04

Из анализа конкурентов было выявлено, что коэффициент конкурентоспособности проекта составляет 4,12, что немного выше, чем у других разработок. Исходя из этого следует, что разрабатываемая система

является конкурентоспособной в условиях существующего рынка.

Также из оценочной карты можно сделать вывод о том, что разрабатываемая система выигрывает в цене и поддерживаемом оборудовании, но проигрывает в надежности и послепродажном обслуживании.

Таким образом, данная разработка благодаря своей низкой стоимости и широкому диапазону поддерживаемого оборудования может занять свою нишу.

4.3 SWOT-анализ

Для исследования внутренних и внешних факторов, оценке рисков и конкурентоспособности был проведен комплексный анализ проекта – SWOT анализ, результаты которого приведены в таблице 4.

Таблица 4 – Итоговая матрица SWOT-анализа

	Сильные стороны: С1. Низкая стоимость. С2. Энергоэффективность технологии. С3. Минимальные затраты ресурсов на обслуживание. С4. Отсутствие привязки к конкретному оборудованию	Слабые стороны: Сл1. Ограниченные функциональные возможности. Сл2. Отсутствие опыта внедрения на предприятия. Сл3. Отсутствие производственных испытаний
Возможности: В1. Рост количества предприятий, где необходима оперативная индикация параметров технологического процесса. В2. Выход на иностранный рынок. В3. Повышение стоимости разработок конкурентов. В4. Увеличение требований безопасности производств	Низкая стоимость и минимальные затраты на обслуживание в случае увеличения требований безопасности производств позволит быстро увеличить объем прибыли, расширить границы сбыта и клиентской базы, так как предприятиям будет необходимо быстро и с минимальными затратами соответствовать новым требованиям. Так как система не привязана к конкретному оборудованию, это повышает шансы выхода на иностранный рынок.	Рост количества предприятий, где необходима оперативная индикация параметров процесса и повышение стоимости разработок конкурентов позволят получить опыт внедрения и произвести достаточное количество испытаний. Для того, чтобы соответствовать новым требованиям безопасности, функциональные возможности данной системы могут дополнять установленную SCADA систему или оборудование, вместо полной замены АСУ ТП.

Продолжение таблицы 4 – Итоговая матрица SWOT-анализа

Угрозы: У1. Снижение стоимости разработок конкурентов У2. Нестабильность экономической ситуации в стране. У3. Появление новых технологий оперативной индикации параметров технологического процесса. У4. Введение дополнительных государственных требований к сертификации продукции	Благодаря энергоэффективности технологии и минимальным затратам на обслуживание, несмотря на снижение стоимости разработок конкурентов, разрабатываемая система может оказаться для предприятия более выгодной. Из-за того, что у системы отсутствует привязка к конкретному оборудованию или SCADA, снижение стоимости разработок конкурентов может не повлечь за собой снижение спроса на данную систему.	Так как разрабатываемая система не перегружена функциональными возможностями, существует вероятность быстрого внедрения в нее новых технологий оперативной индикации параметров технологического процесса. Создание сильной рекламной кампании может позволить наработать клиентскую базу и получить опыт внедрения на предприятия и производственные испытания.
---	--	--

Таким образом, благодаря своим сильным сторонам проект может реализовать все свои возможности. Низкая стоимость и энергоэффективность технологии позволяют использовать практически все возможности для развития разработки. С появлением первых клиентов такие слабые стороны, как отсутствие опыта внедрения на предприятия и отсутствие производственных испытаний будут устранены.

4.4 Планирование научно-исследовательских работ

4.4.1 Структура работы в рамках научного исследования

Планирование комплекса предполагаемых работ осуществляется в следующем порядке:

- определение структуры работ в рамках научного исследования;
- определение участников каждой работы;
- установление продолжительности работ;
- построение графика проведения научных исследований.

Для реализации проекта необходимы исполнители в лице

руководителя и инженера. Перечень этапов, работ и исполнителей приведен в таблице 5.

Таблица 5 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ работы	Содержание работы	Должность исполнителя
Разработка технического задания - ТЗ	1	Составление и утверждение технического задания	Руководитель, инженер
Теоретические исследования	2	Подбор и изучение материалов по теме	Инженер
	3	Изучение существующих систем	Инженер
	4	Календарное планирование работ по теме	Руководитель, инженер
Проведение ОКР			
Разработка и проектирование системы	5	Настройка и установка сервера	Инженер
	6	Создание Android приложения	Инженер
	7	Подключение к ПЛК	Инженер
	8	Тестирование и отладка	Руководитель, Инженер
	9	Оценка полученных результатов	Руководитель
Оформление отчета по НИР (комплекта документации по ОКР)	10	Проведение технико-экономических расчетов и оценка безопасности и экологичности проекта	Инженер
	11	Составление пояснительной записки	Инженер

4.4.2 Определение трудоемкости выполнения работ и разработка графика проведения

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 $t_{ожи}$ используется следующая формула:

$$t_{\text{ож}i} = \frac{3t_{\text{min}i} + 2t_{\text{max}i}}{5}, \quad (1)$$

где $t_{\text{ож}i}$ – ожидаемая трудоемкость выполнения i -ой работы чел.-дн.;

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

$t_{\text{max}i}$ – максимально возможная трудоемкость выполнения заданной i -ой работы, чел.-дн.

Была найдена продолжительность каждой работы в рабочих днях T_p , учитывающая параллельность выполнения работ несколькими исполнителями.

$$T_{pi} = \frac{t_{\text{ож}i}}{Ч_i}, \quad (2)$$

где T_{pi} – продолжительность одной работы, раб. дн.;

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

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

Наиболее удобной и наглядной формой представления графика проведения работ является ленточный график в форме диаграммы Ганта. Для удобства построения графика, длительность каждого из этапов работ из рабочих дней следует перевести в календарные дни. Для этого необходимо воспользоваться формулой:

$$T_{ki} = T_{pi} \cdot k_{\text{кал}}, \quad (3)$$

где T_{ki} – продолжительность выполнения i -ой работы в календарных днях;

T_{pi} – продолжительность выполнения i -ой работы в рабочих днях;

$k_{\text{кал}}$ – коэффициент календарности.

Коэффициент календарности определяется по формуле:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}} = \frac{365}{365 - 52 - 14} = 1,22, \quad (4)$$

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

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

$T_{\text{пр}}$ – количество праздничных дней в году.

Полученные данные сведены в таблицу 6.

Таблица 6 – Временные показатели проведения научного исследования

Название работы	Трудоемкость работы						Длительность работы в рабочих днях, T_{pi}		Длительность работы в календарных днях, T_{ki}	
	t_{min} , чел-дни		t_{max} , чел-дни		$t_{\text{ож}}$, чел-дни		Руководитель	Инженер	Руководитель	Инженер
	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер				
Составление и утверждение технического задания	3		7		4,6		4,6		6	
Подбор и изучение материалов по теме		14		28		9,8		19,6		24
Изучение существующих систем		3		7		4,6		4,6		6
Календарное планирование работ по теме	1	1	4	4	2,2	2,2	1,1	1,1	1	1
Разработка сервера		7		14		9,8		9,8		12
Создание Android приложения		10		20		9,8		9,8		12
Подключение к ПЛК		7		14		14		14		17
Тестирование и отладка	1	1	4	4	14	9,8	4,9	4,9	6	6
Оценка полученных результатов	1		4		2,2		2,2		1	
Проведение технико-экономических расчетов и оценка безопасности проекта		7		14		9,8		9,8		12
Составление пояснительной записки		7		14		9,8		9,8		12
Итого							12,8	83,4	14	102

4.4.3 Разработка графика проведения научного исследования

На основе полученной таблицы строится календарный план-график. График строится для максимального по длительности исполнения работ с разбивкой по месяцам и декадам. График работ приведен в Приложении В.

4.5 Бюджет научно-технического исследования (НТИ)

4.5.1 Расчет материальных затрат НТИ

Расчет материальных затрат осуществляется по следующей формуле:

$$Z_m = (1 + k_T) \cdot \sum_{i=1}^m C_i \cdot N_{расхi}, \quad (5)$$

где m – количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{расхi}$ – количество материальных ресурсов i -го вида;

C_i – цена приобретения единицы i -го вида материальных ресурсов;

k_T – коэффициент, учитывающий транспортно-заготовительные расходы.

Таблица 7 – Материальные затраты

Наименование	Единица измерения	Количество		Цена за ед.,руб.	Сумма, руб.	
		Исп.1	Исп.2		Исп.1	Исп.2
Ноутбук Xiaomi Mi Notebook Air 13.3	Шт.	1	1	56700	56700	56700
Микрокомпьютер Raspberry Pi 3 Model B+	Шт.	1	1	3950	3950	3950
Точка доступа TP-LINK TL-WA801ND	Шт.	1	1	1450	1450	1450
Клавиатура DEXP K-501BU	Шт.	1	1	399	399	399
Сетевое зарядное устройство	Шт.	2	2	180	360	360
Кабель FinePower HDMI - HDMI, 1 м	Шт.	1	1	350	350	350
Кабель управления RS-232	Шт.	1	1	1026	1026	1026
Адаптер USB-RS485	Шт.	1	1	132	132	132
7" дисплей для Raspberry Pi	Шт.	1	1	3690	3690	3690
NAIKU WristWatch A1	Шт.	0	10	2950	0	29500
Всего за материалы					68057	97557
Транспортно-заготовительные расходы (3%)					2041,71	2926,71
Итого					70098,71	100483

4.5.2 Расчет затрат на специальное оборудование для научных (экспериментальных) работ

В данную статью включают все затраты, связанные с приобретением специального оборудования (приборов, контрольно-измерительной техники, стендов, устройств и механизмов), однако оборудование специально для проекта не приобреталось, поэтому была рассчитана амортизация оборудования на время проекта.

Таблица 8 – Расчет амортизации оборудования на время проекта

Наименование	Цена за ед.,руб.	Количество	Срок службы	Время проекта	Отчисления в месяц
ПЛК Siemens Simatic S7-200	18500	1	10 лет	1 месяц	309
Итого					309

4.5.3 Основная заработная плата исполнителей темы

В настоящую статью включается основная заработная плата научных и инженерно-технических работников, рабочих макетных мастерских и опытных производств, непосредственно участвующих в выполнении работ по данной теме. Величина расходов по заработной плате определяется исходя из трудоемкости выполняемых работ и действующей системы окладов и тарифных ставок. В состав основной заработной платы включается премия, выплачиваемая ежемесячно из фонда заработной платы в размере 20 –30 % от тарифа или оклада.

Статья включает основную заработную плату работников, непосредственно занятых выполнением НТИ, и дополнительную заработную плату:

$$Z_{ЗП} = Z_{осн} + Z_{доп}, \quad (6)$$

где $Z_{осн}$ – основная заработная плата;

$Z_{доп}$ – дополнительная заработная плата (12-20% от основной).

Основная заработная плата руководителя (лаборанта, инженера) от предприятия рассчитывается по следующей формуле:

$$Z_{осн} = Z_{дн} \cdot T_p, \quad (7)$$

где $Z_{осн}$ – основная заработная плата одного работника;

$Z_{дн}$ – среднедневная заработная плата работника, руб.;

T_p – продолжительность работ, выполняемых работником, раб.дн.

Среднедневная зарплата рассчитывается по формуле:

$$Z_{дн} = \frac{Z_m \cdot M}{F_d}, \quad (8)$$

где Z_m – месячный должностной оклад работника, руб.;

M – количество месяцев работы без отпуска в течение года:

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.

Баланс рабочего времени приведен в таблице 9.

Таблица 9 – Баланс рабочего времени

Показатели рабочего времени	Руководитель	Студент
Календарное число дней	365	365
Количество нерабочих дней		
-выходные дни	52	52
-праздничные дни	14	14
Потери рабочего времени (отпуск/невыходы по болезни)	48	48
Действительный годовой фонд рабочего времени	251	251

Месячный должностной оклад работника:

$$Z_m = Z_{ТС} \cdot (1 + k_{np} + k_d) \cdot k_p, \quad (9)$$

где $Z_{ТС}$ – заработная плата по тарифной ставке, руб.;

k_p – районный коэффициент, равный 1,3 для Томска.

Так как работа над системой не связана с угрозой для жизни и не является вредным производством, $k_{пр}$ (премиальный коэффициент) и k_d (коэффициент доплат) равны 0.

Расчет основной платы представлен в таблице 10.

Таблица 10 – Расчет основной заработной платы

Исполнители	Оклад	$k_{пр}$	k_d	k_p	Z_m , руб	$Z_{дн}$, руб.	T_p , раб. дн.	$Z_{осн}$, руб.
Руководитель	33664	-	-	1,3	43763,2	1813,3	12,8	23210,24
Инженер	12664	-	-	1,3	16463,2	682,14	83,4	56890,48
Итого:								80100,72

4.5.4 Дополнительная заработная плата исполнителей темы

Расчет дополнительной заработной платы ведется по формуле:

$$Z_{доп} = k_{доп} \cdot Z_{осн}, \quad (10)$$

где $k_{доп}$ – коэффициент дополнительной заработной платы, на стадии проектирования принимается равным 0,12. Расчеты дополнительной заработной платы представлены в таблице 11.

Таблица 11 – Расчет дополнительной заработной платы

Исполнители	$Z_{осн}$, руб.	$k_{доп}$	$Z_{доп}$, руб.
Руководитель	23210,24	0,12	2785,23
Инженер	56890,48	0,12	6826,86
Итого:			9612,09

4.5.5 Отчисление во внебюджетные фонды

В данной статье отражаются обязательные отчисления по установленным законодательствам Российской Федерации нормам органам государственного социального страхования, пенсионного фонда и медицинского страхования.

Величина отчислений во внебюджетные фонды определяется исходя из следующей формулы:

$$Z_{внеб} = k_{внеб} \cdot (Z_{осн} + Z_{доп}), \quad (11)$$

где $k_{внеб}$ - коэффициент отчислений на уплату во внебюджетные фонды.

Расчет отчислений приведен в таблице 12.

Таблица 12 – Расчет отчисления во внебюджетные фонды

Исполнитель	Основная з/п, руб.	Дополнительная з/п, руб.	Коэффициент отчислений	Отчисления
Руководитель	23210,24	2785,23	0,271	7044,77
Инженер	56890,48	6826,86		17267,4
Итого				24312,17

4.6 Накладные расходы

В накладные расходы должны быть включены те затраты организации, которые не попали в предыдущие статьи расходов: оплата электроэнергии, услуг связи, размножение материалов, печать и ксерокопирование материалов и т.д.

Накладные расходы определяются по следующей формуле:

$$Z_{\text{накл}} = k_{\text{нр}} \cdot (Z_{\text{осн}} + Z_{\text{доп}}), \quad (12)$$

где $k_{\text{нр}}$ – коэффициент, учитывающий накладные расходы, 16 %.

Получим:

$$Z_{\text{накл}} = 0,16 \cdot (80100,72 + 9612,09) = 14354,05 \text{ руб}$$

Таким образом, накладные расходы составили 14354,05 рублей.

4.7 Прямые затраты

В этих расходах нужно посчитать затраты на электроэнергию, потребляемую оборудованием. Для этого нужно узнать мощность, время использования оборудования и рассчитать затраты.

Стоимость 1 кВт – составляет 5,8 руб.

Средняя мощность, потребляемая электрооборудованием во время работы, – 200 Вт.

Время использования электрооборудования составляет 498 часов.

Тогда затраты на электроэнергию составят:

$$Z_{\text{эн}} = 0.2 \cdot 498 \cdot 5,8 = 577,7 \text{ руб.}$$

Таким образом, прямые расходы составили 577,7 рублей.

4.8 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в качестве нижнего предела затрат на разработку научно-технической продукции

Определение бюджета затрат приведено в таблице 13.

Таблица 13 – Расчет бюджета НТИ

Наименование статьи	Сумма, руб	
	Исп.1	Исп.2
1. Материальные затраты НТИ	70098,71	100483,71
2. Затраты на специальное оборудование для научных работ	309	309
3. Затраты по основной заработной плате исполнителей темы	80100,72	80100,72
4. Затраты по дополнительной заработной плате исполнителей темы	9612,09	9612,09
5. Отчисление во внебюджетные фонды	24312,17	24312,17
6. Накладные расходы	14354,05	14354,05
7. Прямые расходы	577,7	577,7
8. Бюджет затрат НТИ	199365	229749

4.9 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Определение эффективности происходит на основе расчета интегрального финансового показателя, определяемого по следующей формуле:

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{\text{pi}}}{\Phi_{\text{max}}}, \quad (13)$$

где $I_{\text{финр}}^{\text{исп.}i}$ – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта.

$$I_{\text{исп1}} = \frac{199365}{229749} = 0,89 \quad (14)$$

$$I_{\text{исп2}} = \frac{229779}{229779} = 1 \quad (15)$$

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{\text{pi}} = \sum a_i \cdot b_i \quad (16)$$

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

a_i – весовой коэффициент i -го варианта исполнения разработки;

b_i^a, b_i^p – бальная оценка i -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Сравнительный анализ приведен в таблице 14.

Таблица 14 – Сравнительная оценка вариантов исполнения

Критерии	Весовой коэффициент	Исп. 1	Исп. 2
Поддерживаемое оборудование	0,25	5	5
Удобство в эксплуатации	0,04	4	5
Помехоустойчивость	0,11	4	4
Энергоэкономичность	0,21	4	5
Надежность	0,18	3	3
Безопасность	0,14	4	4
Потребность в ресурсах памяти	0,07	5	4
ИТОГ	1	29	30

$$I_{p-исн1} = 0,25 \cdot 5 + 0,04 \cdot 4 + 0,11 \cdot 4 + 0,21 \cdot 4 + 0,18 \cdot 3 + 0,14 \cdot 4 + 0,07 \cdot 5 = 4,14$$

$$I_{p-исн2} = 0,25 \cdot 5 + 0,04 \cdot 5 + 0,11 \cdot 4 + 0,21 \cdot 5 + 0,18 \cdot 3 + 0,14 \cdot 4 + 0,07 \cdot 4 = 4,32$$

Интегральный показатель эффективности вариантов исполнения разработки вычисляется на основании показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{исн.1} = \frac{I_{p-исн1}}{I_{финр}} = \frac{4,14}{0,89} = 4,65 \quad (17)$$

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Сравнительная эффективность проекта (\mathcal{E}_{cp}):

$$\mathcal{E}_{cp} = \frac{I_{исн.1}}{I_{исн.2}} = \frac{4,65}{4,32} = 1,08 \quad (18)$$

Результаты расчета показателей сведены в таблицу 15.

Таблица 15 – Сравнительная эффективность разработок

Показатель	Исп. 1	Исп. 2
Интегральный финансовый показатель	0,87	1
Интегральный показатель ресурсоэффективности	4,14	4,32
Интегральный показатель эффективности	4,65	4,32
Сравнительная эффективность вариантов исполнения	$\mathcal{E}_{12}=1,08$	

Исходя из полученных данных сравнения финансовой и ресурсной эффективности различных вариантов исполнения, несколько более эффективным является первый вариант исполнения.

Вывод к разделу финансовый менеджмент

В результате проведения исследования по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» были определены показатели затрат научно-исследовательской работы.

Также была создана карта сегментирования, из анализа которой было выявлено, что сегмент рынка мониторинга АСУ ТП на мобильных платформах для мелких предприятий остается свободным.

Помимо этого, был произведен анализ конкурентных решений и составлена оценочная карта. Из оценочной карты был сделан вывод о том, что разрабатываемая система выигрывает в цене и поддерживаемом оборудовании, но проигрывает в надежности и послепродажном обслуживании.

Были определены сильные и слабые стороны системы, ее угрозы и возможности, и исходя из этих характеристик были разработаны 4 стратегии.

Было произведено планирование научно-исследовательских работ.

В конце была проведена сравнительная оценка различных вариантов исполнения системы, и было выяснено, что, исходя из полученных данных сравнения финансовой и ресурсной эффективности различных вариантов исполнения, несколько более эффективным является первый вариант исполнения.

5 Социальная ответственность

В разделе рассматриваются вопросы соблюдения прав персонала на труд, выполнения требований к безопасности и гигиене труда, к промышленной безопасности, охране окружающей среды и ресурсосбережению. В соответствии со стандартом целями составления настоящего раздела является принятие проектных решений, исключающих несчастные случаи в производстве, и снижение вредных воздействий на окружающую среду.

Объектом исследования и разработки является система для индикации параметров технологического процесса. В состав системы входят: микрокомпьютер, адаптер USB-RS485, дисплей, контроллер, соединительные провода. Разрабатываемая система оперативной индикации параметров технологического процесса является неотъемлемой составляющей любого производства, так как благодаря данной системе можно выявить отклонение процесса от заданных пределов, отказы технологического оборудования, и получить информацию о технологическом процессе. К сожалению, готовые решения до сих пор стоят достаточно дорого, поэтому целью данной работы является создание системы оперативной индикации параметров технологического процесса на микрокомпьютере Raspberry Pi. Индикация данной системы должна осуществляться на дисплее, подключенном к микрокомпьютеру, и на смартфоне или смарт часах под управлением операционной системы Android.

Потенциальными потребителями разрабатываемого решения являются предприятия, на которых установлена или внедряется автоматизированная система управления технологическим процессом и необходим постоянный контроль за производством.

Разработка данной системы осуществлялась в лабораторных условиях, так как доступ к промышленным контроллерам для подключения системы на производстве ограничен в виду их высокой стоимости и возможности возникновения аварий.

5.1 Правовые и организационные вопросы обеспечения безопасности

В подразделе рассмотрены специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства.

Также тезисно приводятся основные эргономические требования к правильному расположению и компоновке рабочей зоны исследователя, проектируемой рабочей зоны в производственных условиях для создания комфортной рабочей среды.

5.1.1 Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства

На предприятии, где планируется применение разрабатываемой системы нормальная продолжительность рабочего времени не может превышать 40 часов в неделю. Порядок исчисления нормы рабочего времени на определенные календарные периоды (месяц, квартал, год) в зависимости от установленной продолжительности рабочего времени в неделю определяется федеральным органом исполнительной власти, осуществляющим функции по выработке государственной политики и нормативно-правовому регулированию в сфере труда. Работодатель обязан вести учет времени, фактически отработанного каждым работником [19].

Режим рабочего времени с разрабатываемой системой должен предусматривать продолжительность рабочей недели (пятидневная с двумя выходными днями, шестидневная с одним выходным днем, рабочая неделя с предоставлением выходных дней по скользящему графику, неполная рабочая неделя), работу с ненормированным рабочим днем для отдельных категорий работников, продолжительность ежедневной работы (смены), в том числе неполного рабочего дня (смены), время начала и окончания работы, время перерывов в работе, число смен в сутки, чередование рабочих и нерабочих дней,

которые устанавливаются правилами внутреннего трудового распорядка в соответствии с трудовым законодательством и иными нормативными правовыми актами, содержащими нормы трудового права, коллективным договором, соглашениями, а для работников, режим рабочего времени которых отличается от общих правил, установленных у данного работодателя, - трудовым договором.

Так как работа с данной системой на предприятии подразумевает сбор и анализ персональных данных для того, чтобы ограничить доступ к параметрам технологического процесса, в целях обеспечения прав и свобод человека и гражданина работодатель и его представители при обработке персональных данных работника обязаны осуществлять защиту персональных данных работника от неправомерного их использования или утраты, которая должна быть обеспечена работодателем за счет его средств в порядке, установленном настоящим Кодексом и иными федеральными законами.

5.1.2 Организационные мероприятия при компоновке рабочей зоны

Работа с разрабатываемой системой подразумевает, что оператор осуществляет взаимодействие с ней сидя.

Общие эргономические требования к рабочему месту при выполнении работ сидя:

1. Конструкция рабочего места и взаимное расположение всех его элементов (сиденье, органы управления, средства отображения информации и т.д.) должны соответствовать антропометрическим, физиологическим и психологическим требованиям, а также характеру работы.

2. Рабочее место должно быть организовано в соответствии с требованиями стандартов, технических условий и (или) методических указаний по безопасности труда.

3. Конструкцией рабочего места должно быть обеспечено выполнение трудовых операций в пределах зоны досягаемости моторного поля.

4. Выполнение трудовых операций "часто" и "очень часто" должно быть обеспечено в пределах зоны легкой досягаемости и оптимальной зоны моторного поля.

5. При проектировании оборудования и организации рабочего места следует учитывать антропометрические показатели женщин (если работают только женщины) и мужчин (если работают только мужчины); если оборудование обслуживают женщины и мужчины - общие средние показатели женщин и мужчин [20].

5.2 Производственная безопасность

В подразделе проанализированы вредные и опасные факторы, которые могут возникать при проведении исследований в лаборатории, при разработке или эксплуатации проектируемого решения.

Перечень опасных и вредных факторов, характерных для объекта исследования представлен в виде таблицы 16 [21].

Таблица 16 – Возможные опасные и вредные факторы

Факторы (ГОСТ 12.0.003-2015)	Этапы работ			Нормативные документы
	Разраб отка	Изгото вление	Эксплу атация	
1.Перенапряжение анализаторов, в том числе вызванное информационной нагрузкой	+	+	+	ТОИ Р-45-084-01 СП 52.13330.2016 Естественное и искусственное освещение. СанПиН 2.2.1/2.1.1.1278-03 ГОСТ 12.4.011-89
2.Отсутствие или недостаток естественного света	+	+	+	
3.Недостаточная освещенность рабочей зоны	+	+	+	
4.Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека		+	+	

5.2.1 Анализ вредных и опасных факторов, которые может создать объект исследования

Основная характеристика анализаторов – высокая чувствительность, хотя не всякий раздражитель, действующий на анализатор, вызывает ощущение. Чтобы ощущение проявилось, необходима определенная интенсивность раздражителя. Всякое воздействие, превышающее предел интенсивности, вызывает боль и нарушение деятельности анализаторов. Задача правильного проектирования технологического процесса – не допустить перенапряжения анализаторов, которое может привести к стрессам.

1. источник возникновения фактора – поступающая информация с монитора микрокомпьютера или смартфона о ходе технологического процесса;
2. воздействие фактора на организм человека - интенсивное или длительное воздействие перенапряжение анализаторов может привести к функциональному чрезмерному напряжению, стать причиной профессиональных заболеваний;
3. допустимые нормы указаны в таблице 17 [22];

Таблица 17 – Категории работы по тяжести и напряженности

Категория работы	Уровень нагрузки за рабочую смену при видах работы с компьютером			Суммарное время регламентированных перерывов, мин.	
	Группа А Количество знаков	Группа Б Количество знаков	Группа В Время работы, ч	При 8-ми часовой рабочей смене	При 12-ти часовой рабочей смене
I	До 20000	До 15000	До 2,0	30	70
II	До 40000	До 30000	До 4,0	50	90
III	До 60000	До 40000	До 6,0	70	120

4. предлагаемыми средствами защиты (коллективные и индивидуальные) для минимизации воздействия фактора являются регулярные

перерывы для сотрудников, работающих с данной системой, снижение количество сигналов, передаваемых разрабатываемой системе.

Помещения должны иметь как естественное, так и искусственное освещение. Естественное освещение осуществляется через светопроемы, обеспечивающие необходимый коэффициент естественной освещенности (КЕО) не ниже 1,2 %.

Искусственное освещение в помещениях пульта управления должно осуществляться системой равномерного освещения.

1. источник возникновения фактора - вредное воздействие параметров освещения проявляется в отсутствии или недостатке естественного света, а также недостаточной освещенности рабочей зоны;

2. воздействие фактора на организм человека - недостаточное освещение влияет на функционирование зрительного аппарата, то есть определяет зрительную работоспособность, на психику человека, его эмоциональное состояние, вызывает усталость центральной нервной системы, возникающей в результате прилагаемых усилий для опознания четких или сомнительных сигналов;

3. допустимые нормы: Применение светильников без рассеивателей и экранирующих решеток не допускается. Яркость светильников общего освещения в зоне углов излучения от 50 до 90 градусов с вертикалью в продольной и поперечной плоскостях должна составлять не более 200 кд/м², а защитный угол светильников должен быть не менее 40 градусов. Коэффициент запаса (Кз) для осветительных установок общего освещения должен приниматься равным 1,5. Коэффициент пульсации не должен превышать 15%, что должно обеспечиваться применением газоразрядных ламп в светильниках общего освещения с высокочастотными пускорегулирующими аппаратами (ВЧ ПРА) для любых типов светильников;

4. предлагаемые средства защиты - согласно ГОСТ 12.4.011-89 к средствам нормализации освещенности производственных помещений рабочих

мест относятся: источники света, осветительные приборы, световые проемы, светозащитные устройства, светофильтры, защитные очки.

5.2.2 Анализ вредных и опасных факторов, которые могут возникнуть в лаборатории при проведении исследований

Помимо факторов, описанных в пункте 5.2.1 в лаборатории при проведении исследований может возникнуть повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека

При эксплуатации электрического оборудования необходимо помнить о возможности случайного прикосновения или приближения к токоведущим частям и прикосновения к металлическим нетоковедущим частям, случайно оказавшимся под напряжением.

1. источник возникновения фактора – электрическое оборудование (микрокомпьютер, промышленный контроллер);

2. воздействие фактора на организм человека - электрический ток, проходя через организм, оказывает термическое, электролитическое и биологическое действие. Термическое действие выражается в ожогах отдельных участков тела, нагреве кровеносных сосудов, нервов и других тканей. Электролитическое действие выражается в разложении крови и других органических жидкостей, что вызывает значительные нарушения их физико-химических составов. Биологическое действие является особым специфическим процессом, свойственным лишь живой материи. Оно выражается в раздражении и возбуждении живых тканей организма, а также в нарушении внутренних биоэлектрических процессов, протекающих в нормально действующем организме и теснейшим образом связанных с его жизненными функциями;

3. допустимые нормы: номинальное напряжение не превышает 50 В переменного тока (действующее значение) или 120 В постоянного (выпрямленного) тока. Напряжения прикосновения и токи, протекающие через

тело человека при нормальном (неаварийном) режиме электроустановки, не должны превышать значений, указанных в таблице 18 [23];

Таблица 18 – Допустимые нормы

Род тока	U, В	I, мА
	Не более	
Переменный, 50 Гц	2,0	0,3
Переменный, 400 Гц	3,0	0,4
Постоянный	8,0	1,0

4. предлагаемые средства - для защиты от поражения электрическим током все токоведущие части должны быть защищены от случайных прикосновений кожухами, корпус устройства должен быть заземлен. Заземление выполняется изолированным медным проводом сечением 1.5 мм, который присоединяется к общей шине заземления с общим сечением 48 м при помощи сварки. Общая шина присоединяется к заземлению, сопротивление которого не должно превышать 4 Ом.

5.2.3 Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов

Для защиты исследователя от поражения электрическим током, от действия электрической дуги и т.п. все электроустановки должны быть снабжены средствами защиты, а также средствами оказания первой помощи в соответствии с действующими правилами применения и испытания средств защиты, используемых в электроустановках.

Применяемое в исследованиях электрооборудование, электротехнические изделия и материалы должны соответствовать требованиям государственных стандартов или технических условий, утвержденных в установленном порядке.

В электропомещениях с установками напряжением до 1 кВ допускается применение неизолированных и изолированных токоведущих частей без защиты от прикосновения, если по местным условиям такая защита не является необходимой для каких-либо иных целей (например, для защиты от механических воздействий). При этом доступные прикосновению части должны располагаться так, чтобы нормальное обслуживание не было сопряжено с опасностью прикосновения к ним.

В отношении опасности поражения людей электрическим током различаются следующие типы помещений:

1. Помещения без повышенной опасности, в которых отсутствуют условия, создающие повышенную или особую опасность (см. пп. 2 и 3).

2. Помещения с повышенной опасностью, характеризующиеся наличием одного из следующих условий, создающих повышенную опасность:

- сырость или токопроводящая пыль;
- токопроводящие полы (металлические, земляные, железобетонные, кирпичные);
- высокая температура;
- возможность одновременного прикосновения человека к металлоконструкциям зданий, имеющим соединение с землей, технологическим аппаратам, механизмам и т.п., с одной стороны, и к металлическим корпусам электрооборудования (открытым проводящим частям), с другой.

3. Особо опасные помещения, характеризующиеся наличием одного из следующих условий, создающих особую опасность:

- особая сырость;
- химически активная или органическая среда;
- одновременно два или более условий повышенной опасности.

4. Территория открытых электроустановок в отношении опасности поражения людей электрическим током приравнивается к особо опасным помещениям.

Таким образом, лаборатория, в которой проводятся исследования, соответствует «Правилам устройства электроустановок» и другим нормативам и не требует мероприятий по защите исследователя от действия опасных и вредных факторов

5.3 Экологическая безопасность

В подразделе рассмотрен характер воздействия проектируемого решения на окружающую среду. Выявляются предполагаемые источники загрязнения окружающей среды, возникающие в результате разработки и реализации, предлагаемых в ВКР решений.

5.3.1 Анализ влияния объекта исследования на окружающую среду

Объект исследования не оказывает влияния на окружающую среду, так как микрокомпьютер не осуществляет выбросов вредных веществ в атмосферу и гидросферу. В случае неисправности объекта исследования или демонтажа, чтобы избежать загрязнения литосферы, необходимо произвести его утилизацию.

5.3.2 Анализ влияния процесса исследования на окружающую среду

В ходе выполнения исследования не осуществляет выбросов вредных веществ в атмосферу и гидросферу. Загрязнение атмосферного воздуха может возникнуть в случае возникновения пожара в помещении, в этом случае дым и газы от пожара будут являться антропогенным загрязнением атмосферного воздуха.

Основные виды загрязнения литосферы – твердые бытовые и промышленные отходы, а также отходы возникающие в случае поломки ноутбука. В ходе выполнения исследования, образовывались различные твердые

отходы. К ним можно отнести: бумагу, лампочки, отходы личной гигиены, отходы от канцелярских принадлежностей и т.д.

5.3.3 Обоснование мероприятий по защите окружающей среды

Защита почвенного покрова и недр от твердых отходов реализуется за счет сбора, сортирования и утилизации отходов и их организованного захоронения. Главными нормативными актами, регулирующими вопрос утилизации микрокомпьютеров, являются федеральные законы РФ «Об охране окружающей среды» и «Об отходах производства и потребления». Согласно этим законам вся оргтехника подлежит утилизации с соблюдением определенных правил: демонтаж запчастей, сортировка отходов и утилизация.

5.4 Безопасность в чрезвычайных ситуациях

В подразделе проведен краткий анализ возможных чрезвычайных ситуаций (ЧС), которые могут возникнуть при разработке, производстве или эксплуатации проектируемого решения. Чрезвычайные ситуации могут быть техногенного, природного, биологического, социального или экологического характера.

5.4.1 Анализ вероятных ЧС, которые может инициировать объект исследований

Объект исследований может инициировать возникновение такой чрезвычайной ситуации как пожар. Причинами пожара могут быть неисправность источника питания или микрокомпьютера.

5.4.2 Анализ вероятных ЧС, которые могут возникнуть в лаборатории при проведении исследований

При проведении исследований в лаборатории также может возникнуть пожар. Причинами пожара могут быть: игнорирование основных правил пожарной безопасности, неисправность электрической проводки, возгорание устройств искусственного освещения, возгорание устройств вычислительной аппаратуры вследствие нарушения изоляции или неисправности самой аппаратуры.

5.4.3 Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС

Для защиты от пожаров необходимо иметь в наличии такое пожарное оборудование как пожарные шкафы, пожарные щиты и огнетушители. Сотрудники должны уметь пользоваться таким оборудованием. Для ликвидации пожаров, вызванных возгоранием электрооборудования, применяются углекислотные огнетушители.

Сотрудники должны знать план эвакуации из помещения, расположение выходов из здания. Также необходимо проводить плановые эвакуации из здания, для того чтобы подготовить сотрудников к действиям в чрезвычайной ситуации.

Заключение к разделу социальная ответственность

Таким образом, в данном разделе были рассмотрены вопросы соблюдения прав персонала на труд, выполнения требований к безопасности и гигиене труда, к промышленной безопасности, охране окружающей среды и ресурсосбережению.

Было выявлено, что разрабатываемая система не оказывает никакого влияния на атмосферу и гидросферу, но для того, чтобы предотвратить загрязнение литосферы необходимо производить утилизацию отходов в ходе исследования.

Также в ходе работ было выяснено, что объект исследования или выполнение исследований в лаборатории могут инициировать возникновение такой чрезвычайной ситуации как пожар.

Помимо этого, были определены опасные и вредные факторы, которые могут возникнуть на разных этапах работ, а именно: перенапряжение анализаторов, в том числе вызванное информационной нагрузкой, отсутствие или недостаток естественного света, недостаточная освещенность рабочей зоны, повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека.

Полученными результатами также являются рекомендации по устранению и предотвращению пожаров, проведению утилизации отходов и устранения опасных и вредных факторов, рассматриваемых в данной работе. Также были сформулированы требования к организации рабочего места и организации труда. Данные рекомендации и требования могут быть внедрены на производствах, где планируется использование рассматриваемой системы.

Заключение

В ходе выполнения дипломной работы были выполнены следующие задачи:

1. Была осуществлена настройка точки доступа Wi-Fi на микрокомпьютере Raspberry Pi. Данная точка доступа использует встроенный модуль Wi-Fi, но в случае, если дальности действия встроенного модуля недостаточно, имеется возможность подключения USB-адаптеров для ее увеличения. Также для данной точки доступа назначен статический IP-адрес, имя сети, пароль и диапазон IP-адресов, который будет предоставляться подключаемым устройствам. Данные параметры настройки доступа при необходимости могут быть изменены путем редактирования файлов конфигурации.

2. Было произведено подключение к промышленному логическому контроллеру. Подключение осуществлялось с использованием промышленного протокола Modbus и адаптера USB-RS485. Для того чтобы активировать протокол Modbus и разрешить доступ к параметрам для контроллера Siemens S7-200 была написана программа на языке LD, в которой указаны параметры подключения. Эти же параметры подключения использовались в программе для Raspberry Pi. Помимо этого, было произведено исследование основных библиотек Python, с помощью которых осуществлялось чтение параметров. Для дальнейшей работы была выбрана Modbus-TK, так как она обеспечивает стабильное подключение к ПЛК и способна считывать до 120 параметров за 1 запрос.

3. Был создан сервер, который обеспечивает передачу параметров технологического процесса. Данный сервер реализован с использованием протокола TCP и библиотеки socketserver. Для того, чтобы обеспечить одновременное подключение нескольких клиентов, были использованы потоки, которые создаются для каждого клиента и обрабатывают запросы на чтение параметров только этого клиента независимо от остальных потоков. Также реализована возможность кэширования считанных параметров. Благодаря

кэшированию, скорость работы сервера возрастает почти в 3 раза, если к нему подключено несколько клиентов, которым требуются одни и те же параметры.

4. Было разработано клиентское приложение для операционной системы Android. Данное приложение реализовано с использованием библиотек Kivy и Socket. В данном приложении реализованы возможности настройки сигнализации и выбора параметров, которые необходимо получить. Также данное приложение, благодаря кросс-платформенности библиотеки Kivy способно работать на операционных системах Window и Linux.

Таким образом, разработанная система была успешно протестирована на контроллере Siemens S7-200 с передачей параметров на смартфон, в ее работе никаких сбоев и ошибок не наблюдалось.

Благодаря тому, что в данной системе используется протокол Modbus, ее применение не ограничивается контроллером Siemens S7-200. Она может быть применена для чтения параметров с любых контроллеров, которые поддерживают протокол Modbus.

Данная система может быть применена на различных предприятиях, где используется АСУ ТП, для того, чтобы оперативно оповещать рабочий персонал об отклонении процесса от заданных пределов и аварийной ситуации.

В дальнейшем могут быть произведены следующие улучшения системы:

1. Использование более компактного, дешевого и энергоэффективного микрокомпьютера Raspberry Pi Zero взамен Raspberry Pi 3B.
2. Компиляция приложения для операционной системы IOS для поддержки устройств фирмы Apple.
3. Реализация чтения параметров технологического процесса по протоколу Ethernet для увеличения количества поддерживаемых контроллеров.

Conclusion

In this paper, the following tasks were performed:

1. The Wi-Fi access point was configured on the Raspberry Pi microcomputer. This access point uses the built-in Wi-Fi module, but if the range of the built-in module is not enough, it is possible to connect USB adapters to increase it. Also, this access point is assigned a static IP address, a network name, a password and a range of IP addresses that will be provided to connected devices. These access settings can be changed if necessary by editing configuration files.

2. The connection was made to the industrial logic controller. The connection was made using the industrial Modbus protocol and USB-RS485 adapter. In order to activate the Modbus protocol and allow access to the parameters for the Siemens S7-200 controller, an LD program was written in which the connection parameters are specified. The same connection parameters were used in the program for the Raspberry Pi. In addition, Modbus-TK was chosen for further work, as it provides a stable connection to the PLC and is capable of reading up to 120 parameters in 1 request.

3. The server was created that provides the transfer of process parameters. This server is implemented using the TCP protocol and the socketserver library. In order to provide simultaneous connection of several clients, threads were created that are created for each client and thread read requests for parameters of this client only, independently of other streams. Also implemented the ability to cache read parameters. Thanks to caching, the speed of the server increases by almost 3 times, if several clients are connected to it, which require the same parameters.

4. The client application for the Android operating system was developed. This application is implemented using the Kivy and Socket libraries. This application provides the ability to customize the alarm and select the parameters that you want to receive. Also, this application, thanks to the cross-platform library Kivy is able to work on operating systems Window and Linux.

Thus, the developed system was successfully tested on the Siemens S7-200 controller with the transfer of parameters to the smartphone. Failures and errors were not observed in its operation.

Due to the fact that this system uses the Modbus protocol, its application is not limited to the Siemens S7-200 controller. It can be used to read parameters from any controllers that support the Modbus protocol.

This system can be applied at various enterprises where the process control system is used, in order to promptly notify the operating personnel of the process deviation from the specified limits and emergency situation.

In the future, the following system improvements can be made:

1. Using a more compact, cheap and energy-efficient microcomputer Raspberry Pi Zero instead of Raspberry Pi 3B.
2. Compile an application for the IOS operating system to support Apple devices.
3. Implementing process readings via Ethernet protocol to increase the number of controllers supported.

Список используемых источников

1. Официальный сайт ИнСАТ [Электронный ресурс] / MasterSCADA 4D // URL: <https://insat.ru/products/?category=1536>. – дата обращения 01.05.2019.
2. Официальный сайт SimpLight [Электронный ресурс] / SIMP HMI // URL: <https://simplight.ru/mobile/#more>. – дата обращения 01.05.2019.
3. Электротехнические системы Сибирь [Электронный ресурс] / Мониторинг и управление системами АСУ ТП на мобильных платформах с ОС Android // URL: http://es-electro.ru/novosti/monitoring_i_upravlenie_sistemami_asu_tp_na_mobilnyh_platfor_mah_s_os_android/. – дата обращения 01.05.2019.
4. Промэнерго Автоматика [Электронный ресурс] / Siemens S7-200 Программируемый контроллер // URL: <https://www.siemens-pro.ru/components/s7-200.htm>. – дата обращения 01.05.2019.
5. Русскоязычное сообщество Raspberry Pi [Электронный ресурс] / Заглавная страница // URL: <https://raspberrypi.ru/>. – дата обращения 01.05.2019.
6. Портал о микроконтроллере Raspberry PI [Электронный ресурс] / Аналоги одноплатного мини-ПК Raspberry Pi // URL: <https://myraspberrypi.ru/analogi-raspberry-pi.html>. – дата обращения 01.05.2019.
7. Официальный сайт Raspberry Pi [Электронный ресурс] / Raspbian // URL: <https://www.raspberrypi.org/downloads/raspbian/>. – дата обращения 01.05.2019.
8. Greendail [Электронный ресурс] / Что такое DHCP и как он работает? // URL: <https://greendail.ru/node/chto-takoe-dhcp-i-kak-rabotaet-obyasnenie-osnovnyh-principov>. – дата обращения 01.05.2019.
9. CiscoTips [Электронный ресурс] / Протокол TCP — назначение и функционал // URL: <http://ciscotips.ru/tcp>. – (дата обращения 01.05.2019).
10. ООО Промэнерго Автоматика [Электронный ресурс] / Сети Profibus // URL: <https://www.siemens->

pro.ru/docs/kip/Flowmeters/Others/Profibus/1_Network_PROFIBUS_r.pdf. – дата обращения 01.05.2019.

11. Onitex Industrial Solutions [Электронный ресурс] / Описание протокола Modbus // URL: <http://onitex.ru/modbus-protokol>. – дата обращения 01.05.2019.

12. Энциклопедия АСУ ТП [Электронный ресурс] / Промышленный Ethernet // URL: https://www.bookasutp.ru/Chapter2_9.aspx. – дата обращения 01.05.2019.

13. Портал о микроконтроллере Raspberry Pi [Электронный ресурс] / Программирование на Python на Raspberry Pi // URL: <https://myraspberrypi.ru/programmirovanie-python-na-raspberry-pi.html>. – дата обращения 01.05.2019.

14. Kivy [Электронный ресурс] / Заглавная страница // URL: <https://kivy.org/#home>. – дата обращения 01.05.2019.

15. GitHub [Электронный ресурс] / Репозиторий buildozer // URL: <https://github.com/kivy/buildozer>. – дата обращения 01.05.2019.

16. Официальная документация к Raspberry Pi [Электронный ресурс] / Access point // URL: <https://www.raspberrypi.org/documentation/configuration/wireless/access-point.md>. – дата обращения 01.05.2019.

17. Официальный сайт Siemens [Электронный ресурс] / документация Modbus // URL: https://www.siemens-ru.com/doc/12_Modbus_r.pdf. – дата обращения 01.05.2019.

18. Сайт Microsoft [Электронный ресурс] / Поток и работа с ними // URL: <https://docs.microsoft.com/ru-ru/dotnet/standard/threading/threads-and-threading>. – дата обращения 01.05.2019.

19. Трудовой кодекс Российской Федерации от 30.12. 2001 г. № 197–ФЗ (ред. от 01.04.2019 г.). – М., 2015. – 123 с.

20. ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя.

Общие эргономические требования. – М: ИПК Издательство стандартов, 2001. – 9 с.

21. ГОСТ 12.0.003-2015 Система стандартов безопасности труда (ССБТ). Опасные и вредные производственные факторы. Классификация. – Взамен ГОСТ 12.0.003 — 74; введ. 2017 – 03 – 01. – Москва Стандартиформ, 2016. – 9 с.

22. ТОИ Р-45-084-01 Типовая инструкция по охране труда при работе на персональном компьютере. – М., Минсвязи РФ, 2001. – 5с.

23. ГОСТ 12.1.038-82 Система стандартов безопасности труда (ССБТ). Электробезопасность. Предельно допустимые значения напряжений прикосновения и токов. – М: ИПК Издательство стандартов, 2001. – 7 с.

Приложение А

(обязательное)

Листинг программы сервера

```
#!/usr/bin/env python

import socket
import serial
import threading
import socketserver
import time

import modbus_tk
import modbus_tk.defines as cst
from modbus_tk import modbus_rtu

PORT = '/dev/ttyUSB0'

master = modbus_rtu.RtuMaster(serial.Serial(
    port=PORT, baudrate=9600, bytesize=8, parity='E', stopbits=1, xonxoff=0))
master.set_timeout(5.0)
master.set_verbose(True)
global request_list, data_list
request_list = [';', ';', ';']
data_list = [';', ';', ';']

class ThreadedTCPRequestHandler(socketserver.BaseRequestHandler):

    def handle(self):
        data = str(self.request.recv(1024), 'utf-8')
        data = data.split(', ')
        global request_list, data_list
        for i in reversed(range(len(request_list))):
            if data == request_list[i]:
                mes = data_list[i]
```

```

        request_list.append(' ')
        data_list.append(' ')
        break
else:
    hr = []
    for i in range(int(data[1])//120,
                   (int(data[1])+int(data[2]))//120 + bool((
                       int(data[0])+int(data[1]))% 120)):
        if int(data[0]) == 4:hr.extend(list(master.execute(
            1, cst.READ_HOLDING_REGISTERS,i*120,120)))
        elif int(data[0]) == 0:hr.extend(list(master.execute(
            1, cst.READ_COILS,i*120,120)))
        elif int(data[0]) == 1:hr.extend(list(master.execute(
            1, cst.READ_DISCRETE_INPUTS,i*120,120)))
        elif int(data[0]) == 3:hr.extend(list(master.execute(
            1, cst.READ_INPUT_REGISTERS,i*120,120)))
    print(hr)
    mes = str(hr[int(data[1])//120:int(data[1])//120+int(data[2])])
    mes = bytes(mes, 'utf-8')
    request_list.append(data)
    data_list.append(mes)

    request_list = request_list[1:]
    data_list = data_list[1:]
    self.request.sendall(mes)

class ThreadedTCPServer(socketserver.ThreadingMixIn, socketserver.TCPServer):
    pass

if __name__ == "__main__":
    HOST, PORT = "192.168.0.103", 5005

    server = ThreadedTCPServer((HOST, PORT), ThreadedTCPRequestHandler)
    ip, port = server.server_address

```

```
server_thread = threading.Thread(target=server.serve_forever)
server_thread.daemon = True
server_thread.start()
while 1:
    time.sleep(1)

server.shutdown()
server.server_close()
```

Приложение Б

(обязательное)

Листинг программы клиента

```
# -*- coding: utf-8 -*-
import sys
import os
import time
import socket
from kivy.app import App
from kivy.uix.button import Button
from kivy.uix.textinput import TextInput
from kivy.uix.label import Label
from kivy.uix.gridlayout import GridLayout
from kivy.uix.boxlayout import BoxLayout
from kivy.config import Config
from kivy.adapters.listadapter import ListAdapter
from kivy.uix.listview import ListView
from kivy.uix.listview import ListItemButton
from kivy.uix.popup import Popup
from kivy.core.window import Window
from kivy.uix.image import Image
from kivy.clock import Clock

Window.clearcolor = (1, 1, 1, 1)
Config.set('graphics', 'resizable', 0)
Window.size = (540, 960)
Address = [] # "192.168.0.103", 5005]
alarm_list = []
global do_par, di_par, ai_par, hr_par, send_info
do_par = '0, 0, 100'
di_par = '1, 0, 100'
ai_par = '3, 0, 100'
hr_par = '4, 0, 100'
send_info = hr_par

class EmailApp(App):
    def clear_all(self):
        child = self.root.children[:]
        for i in child:
            time.sleep(0.01)
            self.root.remove_widget(i)

    def dismiss_popup(self, instance):
        self._popup.dismiss()

    def do_set(self, instance):
        global do_par, di_par, ai_par, hr_par, send_info
        self.titl.text = 'Цифровые выходы'
        self.but_0x.color = [0, 0, 0, 1]
```

```

self.but_1x.color = [1, 1, 1, 1]
self.but_3x.color = [1, 1, 1, 1]
self.but_4x.color = [1, 1, 1, 1]
send_info = do_par

def di_set(self, instance):
    global do_par, di_par, ai_par, hr_par, send_info
    self.titl.text = 'Цифровые входы'
    self.but_0x.color = [1, 1, 1, 1]
    self.but_1x.color = [0, 0, 0, 1]
    self.but_3x.color = [1, 1, 1, 1]
    self.but_4x.color = [1, 1, 1, 1]
    send_info = di_par

def ai_set(self, instance):
    global do_par, di_par, ai_par, hr_par, send_info
    self.titl.text = 'Аналоговые входы'
    self.but_0x.color = [1, 1, 1, 1]
    self.but_1x.color = [1, 1, 1, 1]
    self.but_3x.color = [0, 0, 0, 1]
    self.but_4x.color = [1, 1, 1, 1]
    send_info = ai_par

def hr_set(self, instance=0):
    global do_par, di_par, ai_par, hr_par, send_info
    self.titl.text = 'Регистры временного хранения информации'
    self.but_0x.color = [1, 1, 1, 1]
    self.but_1x.color = [1, 1, 1, 1]
    self.but_3x.color = [1, 1, 1, 1]
    self.but_4x.color = [0, 0, 0, 1]
    send_info = hr_par

def alarm_popup(self, message):
    b2 = BoxLayout(orientation='vertical', padding=10, spacing=10)
    I1 = Image(source='E:\images\Alarm1.png')
    b2.add_widget(I1)
    popup = Popup(title=message, title_size=22, content=b2,
                 auto_dismiss=False, size_hint=(0.9, 0.5))
    b2.add_widget(Button(text='Принять', on_release=popup.dismiss, font_size=20,
size_hint=(1, 0.2)))
    popup.open()

def error_popup(self, message):
    b2 = BoxLayout(orientation='vertical')
    popup = Popup(title='Ошибка', title_size=22, content=b2, size_hint=(0.9, 0.3))
    b2.add_widget(Label(text=message, font_size=20))
    b2.add_widget(Button(text='OK', on_release=popup.dismiss, size_hint=(1, 0.3)))
    popup.open()

def alarm_set(self, instance):
    global reg_num
    try:

```

```

flag = False
if self.down_alarm.text == "":
    flag = True
else:
    int(self.down_alarm.text)
if self.up_alarm.text == "":
    flag = True
else:
    int(self.up_alarm.text)
if flag == False:
    if int(self.down_alarm.text) >= int(self.up_alarm.text): raise ValueError
for i in alarm_list:
    if i[0] == reg_num[4:9]:
        i[1] = self.up_alarm.text
        i[2] = self.down_alarm.text
        i[3] = 0
        break
    else:
        alarm_list.append([reg_num[4:9], self.up_alarm.text, self.down_alarm.text, 0])
        if self.up_alarm.text == " and self.down_alarm.text == "":
            alarm_list.pop()
except ValueError:
    self.error_popup('Неправильно указаны границы')

def show_popup(self):
    g1 = GridLayout(cols=2, spacing=30, padding=10)
    g1.add_widget(Label(text='Верхняя граница:', font_size=20, halign='left', size_hint=(.6,
.1)))
    self.up_alarm = (TextInput(multiline=False, font_size=20))
    g1.add_widget(self.up_alarm)
    g1.add_widget(Label(text='Нижняя граница:', font_size=20, halign='left', size_hint=(.6,
.1)))
    self.down_alarm = (TextInput(multiline=False, font_size=20))
    g1.add_widget(self.down_alarm)

    b2 = BoxLayout(orientation='vertical', padding=10)
    b3 = BoxLayout(orientation='horizontal', padding=10, size_hint=(1, .5))
    b3.add_widget(Button(text='Назад', on_release=self.dismiss_popup, font_size=20))
    choose_but = Button(text='Принять', font_size=20, on_press=self.alarm_set,
on_release=self.dismiss_popup)
    b3.add_widget(choose_but)
    b2.add_widget(g1)
    b2.add_widget(b3)
    self._popup = Popup(title="Настройка сигнализации", title_size=20, content=b2,
size_hint=(0.9, 0.3))
    self._popup.open()

class list_but(ListItemButton):
    def get_num(self):
        global reg_num
        reg_num = (self.text)

```

```

on_press = get_num

def save_set(self, instance):
    try:
        global do_par, di_par, ai_par, hr_par, send_info
        do_par = ', '.join(['0', self.do_adr.text, self.do_num.text])
        di_par = ', '.join(['1', self.di_adr.text, self.di_num.text])
        ai_par = ', '.join(['3', self.ai_adr.text, self.ai_num.text])
        hr_par = ', '.join(['4', self.hr_adr.text, self.hr_num.text])
        int(self.do_adr.text)
        int(self.do_num.text)
        int(self.di_adr.text)
        int(self.di_num.text)
        int(self.ai_adr.text)
        int(self.ai_num.text)
        int(self.hr_adr.text)
        int(self.hr_num.text)
    except ValueError:
        self.error_popup('Неправильно указаны настройки')
    else:
        self.rebuild()

def rebuild_set(self, instance):
    try:
        self.clear_all()
        global do_par, di_par, ai_par, hr_par, send_info
        b2 = BoxLayout(orientation='vertical')
        b3 = BoxLayout(orientation='horizontal', size_hint=(1, 0.08))

        g1 = GridLayout(cols=2, spacing=30, padding=25, size_hint=(1, .175))
        g2 = GridLayout(cols=2, spacing=30, padding=25, size_hint=(1, .175))
        g3 = GridLayout(cols=2, spacing=30, padding=25, size_hint=(1, .175))
        g4 = GridLayout(cols=2, spacing=30, padding=25, size_hint=(1, .175))

        g1.add_widget(Label(text='Количество', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
        self.do_num = (TextInput(text=do_par.split(',')[2], multiline=False, font_size=20))
        g1.add_widget(self.do_num)
        g1.add_widget(Label(text='Адрес', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
        self.do_adr = (TextInput(text=do_par.split(',')[1], multiline=False, font_size=20))
        g1.add_widget(self.do_adr)

        g2.add_widget(Label(text='Количество', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
        self.di_num = (TextInput(text=di_par.split(',')[2], multiline=False, font_size=20))
        g2.add_widget(self.di_num)
        g2.add_widget(Label(text='Адрес', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
        self.di_adr = (TextInput(text=di_par.split(',')[1], multiline=False, font_size=20))
        g2.add_widget(self.di_adr)

```

```

g3.add_widget(Label(text='Количество', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
self.ai_num = (TextInput(text=ai_par.split(',')[2], multiline=False, font_size=20))
g3.add_widget(self.ai_num)
g3.add_widget(Label(text='Адрес', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
self.ai_adr = (TextInput(text=ai_par.split(',')[1], multiline=False, font_size=20))
g3.add_widget(self.ai_adr)

g4.add_widget(Label(text='Количество', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
self.hr_num = (TextInput(text=hr_par.split(',')[2], multiline=False, font_size=20))
g4.add_widget(self.hr_num)
g4.add_widget(Label(text='Адрес', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
self.hr_adr = (TextInput(text=hr_par.split(',')[1], multiline=False, font_size=20))
g4.add_widget(self.hr_adr)

but_back = Button(on_release=self.save_set, font_size=24, background_color=[0, 0.6,
0, 1],
background_normal="",
size_hint=(0.2, 1))
I1 = Image(source='E:\images\menu2.png')
but_back.add_widget(I1)
b3.add_widget(but_back)
b3.add_widget(
Button(text='Настройки', bold=True, font_size=28, background_color=[0, 0.6, 0, 1],
background_normal="",
background_down=""))
b3.add_widget(
Button(font_size=28, background_color=[0, 0.6, 0, 1], background_normal="",
background_down="", size_hint=(0.2, 1)))

b2.add_widget(b3)
b2.add_widget(
Label(text='Цифровые выходы:', font_size=24, halign='center', size_hint=(1, 0.05),
color=[0, 0, 0, 1]))
b2.add_widget(g1)
b2.add_widget(
Label(text='Цифровые входы:', font_size=24, halign='center', size_hint=(1, 0.05),
color=[0, 0, 0, 1]))
b2.add_widget(g2)
b2.add_widget(
Label(text='Аналоговые входы:', font_size=24, halign='center', size_hint=(1, 0.05),
color=[0, 0, 0, 1]))
b2.add_widget(g3)
b2.add_widget(
Label(text='Регистры временного хранения информации:', font_size=24,
halign='center',
size_hint=(1, 0.05), color=[0, 0, 0, 1]))
b2.add_widget(g4)

```

```

        I1.x = 0 - I1.width * 0.125
        I1.y = Window.height - I1.height * 0.9
        self.root.add_widget(b2)
    finally:
        pass

def restart(self, instance):
    python = sys.executable
    os.execl(python, python, *sys.argv)

def rebuild(self):
    self.clear_all()
    b2 = BoxLayout(orientation='horizontal', size_hint=(1, 0.1))
    b3 = BoxLayout(orientation='horizontal', size_hint=(1, 0.1))

    but_off = Button(font_size=24, background_color=[0, 0.6, 0, 1], background_normal="",
                    size_hint=(0.2, 1), on_release=self.restart)
    I1 = Image(source='E:\images\off2.png')
    but_off.add_widget(I1)
    b2.add_widget(but_off)
    self.titl = Button(text='Индикация параметров', halign='left', bold=True, font_size=24,
                    background_color=[0, 0.6, 0, 1], background_normal="",
                    background_down="")
    b2.add_widget(self.titl)

    but_set = Button(on_release=self.rebuild_set, font_size=24, background_color=[0, 0.6, 0,
1],
                    background_normal="",
                    size_hint=(0.2, 1))
    I2 = Image(source='E:\images\set2.png')
    but_set.add_widget(I2)
    b2.add_widget(but_set)

    self.list_button = self.list_but # ListItemButton
    self.list_adapter = ListAdapter(data=[], cls=self.list_button,
allow_empty_selection=False)

    self.list_view = ListView(adapter=self.list_adapter)
    self.list_button.background_color = [1, 1, 1, 1]
    self.list_button.background_normal = ""
    self.list_button.on_release = self.show_popup
    self.list_view.container.spacing = 10
    self.list_button.font_size = 24
    self.list_button.font_name = 'arialbd'
    self.list_button.bold = True
    self.list_button.color = [0, 0, 0, 1]
    self.list_button.padding = 30
    self.list_button.halign = 'left'
    self.list_button.text_size = (Window.width, 0)
    self.list_adapter.data = []

    I1.x = 0 - I2.width * 0.125

```

```

I1.y = Window.height - I2.height * 0.9
I2.x = Window.width - I2.width * 0.9
I2.y = Window.height - I2.height * 0.9
self.root.add_widget(b2)
self.root.add_widget(self.list_view)

self.but_0x = Button(on_release=self.do_set, text='0x', bold=True, font_size=28,
                    background_color=[0, 0.6, 0, 1], background_normal="")

self.but_1x = Button(on_release=self.di_set, text='1x', bold=True, font_size=28,
                    background_color=[0, 0.6, 0, 1], background_normal="")

self.but_3x = Button(on_release=self.ai_set, text='3x', bold=True, font_size=28,
                    background_color=[0, 0.6, 0, 1], background_normal="")

self.but_4x = Button(on_release=self.hr_set, text='4x', bold=True, font_size=28,
                    background_color=[0, 0.6, 0, 1], background_normal="")
b3.add_widget(self.but_0x)
b3.add_widget(self.but_1x)
b3.add_widget(self.but_3x)
b3.add_widget(self.but_4x)
self.root.add_widget(b3)
self.hr_set()
Clock.schedule_interval(self.update, 1)

class SocketManager:

    def __init__(self, address):
        self.address = address

    def __enter__(self):
        self.sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        self.sock.connect(self.address)
        return self.sock

    def __exit__(self, *ignore):
        self.sock.close()

    def handle_request(self):
        try:
            with self.SocketManager(tuple(Address)) as sock:
                global send_info
                sock.send(bytes(send_info, 'utf-8'))
                data = sock.recv(1024)

        except socket.error as err:
            print("Connection Error".format(err))
        else:
            return (data)

    def update(self, *args):
        global send_info

```

```

recv = self.handle_request()
recv = recv.decode("utf-8")
recv = recv[1:-1].split(',')
parameters = []
send = send_info.split(',')
for i in range(len(recv)):
    param = send_info[0] + '0' * (4 - len(str(i + int(send[1])))) + str(i + int(send[1]))
    s = ' ' + param + ' = ' + recv[i]
    for j in alarm_list:
        if (param == j[0]):
            if j[1] == ":":
                max = 10000
            else:
                max = int(j[1])
            if j[2] == ":":
                min = -10000
            else:
                min = int(j[2])
            s += (' ' * (14 - 2 * len(recv[i]))) + 'max = ' + j[1] + (' ' * (10 - 2 * len(j[1]))) + 'min
= ' + \
                j[2]
            if (int(recv[i]) < max) and (int(recv[i]) > min) and (j[3] == 1):
                j[3] = 0
            if ((int(recv[i]) >= max) or (int(recv[i]) <= min)) and (j[3] == 0):
                if int(recv[i]) >= int(j[1]):
                    mes = param + ' превысил верхнюю границу!'
                else:
                    mes = param + ' ниже нижней границы!'
                self.alarm_popup(mes)
                j[3] = 1
            if ((int(recv[i]) >= max) or (int(recv[i]) <= min)):
                if int(recv[i]) >= int(j[1]):
                    s += (' ' * (10 - 2 * len(j[2]))) + '▲'
                else:
                    s += (' ' * (10 - 2 * len(j[2]))) + '▼'
    parameters.append(s)
for i in range(10): parameters.append("")
self.list_adapter.data = parameters

def connect(self, instance):
    try:
        Address.append(self.ip_in.text)
        Address.append(int(self.port_in.text))
        s = self.handle_request()
        if s == None: raise ConnectionError
    except ValueError:
        self.error_popup('Неправильно указан порт')
    except ConnectionError:
        self.error_popup('Не удалось подключиться к серверу')
    else:
        self.rebuild()

```

```

def build(self):

    b1 = BoxLayout(orientation='vertical')
    b2 = BoxLayout(orientation='vertical', padding=25, size_hint=[1, 0.8])
    g1 = GridLayout(cols=2, spacing=20, padding=25)

    g1.add_widget(Label(text='IP адрес', font_size=20, halign='left', size_hint=(.2, .1),
color=[0, 0, 0, 1]))
    self.ip_in = (TextInput(multiline=False, text='192.168.0.101', font_size=20))
    g1.add_widget(self.ip_in)
    g1.add_widget(Label(text='Порт', font_size=20, halign='left', size_hint=(.2, .1), color=[0,
0, 0, 1]))
    self.port_in = (TextInput(multiline=False, text='5005', font_size=20))
    g1.add_widget(self.port_in)

    b1.add_widget(
        Button(text='Индикация параметров', bold=True, font_size=28, background_color=[0,
0.6, 0, 1],
            background_normal="",
            background_down="", size_hint=[1, 0.5]))
    b1.add_widget(BoxLayout(orientation='vertical'))
    b1.add_widget(BoxLayout(orientation='vertical'))
    b1.add_widget(g1)
    b1.add_widget(BoxLayout(orientation='vertical'))
    b1.add_widget(BoxLayout(orientation='vertical'))
    b2.add_widget(
        Button(text='Подключение', bold=True, font_size=28, background_color=[0, 0.6, 0, 1],
background_normal="",
            on_release=self.connect))
    b1.add_widget(b2)
    return b1

if __name__ == "__main__":
    App.icon = 'E:\images\logo1_256.png'
    App().run()

```

Приложение В

(обязательное)

Календарный план-график выполнения проекта

№ работ	Вид работ	Исполнители	T _{кi} , кал. дн.	Продолжительность выполнения работ																	
				Янв.			февр.			март			апр.			май			июнь		
				1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
1	Составление и утверждение технического задания	Руководитель	6				▨														
2	Подбор и изучение материалов по теме	Инженер	24				■	■	■												
3	Изучение существующих систем	Инженер	6							■											
4	Календарное планирование работ по теме	Руков., инж.	1										▨								
5	Настройка и установка сервера	Инженер	12										■								
6	Создание Android приложения	Инженер	12										■								
7	Подключение к ПЛК	Инженер	17										■	■	■						
8	Тестирование и отладка	Руков., инж.	6																▨		
9	Оценка полученных результатов	Руководитель	1																▨		
10	Проведение технико-экономических расчетов и оценка безопасности проекта	Инженер	12																■		
11	Составление пояснительной за-писки	Инженер	12																		■



- руководитель



- инженер