

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 – 15.04.04 Автоматизация технологических процессов и производств
 Отделение школы (НОЦ) – Отделение автоматизации и робототехники

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Разработка клиента OPC UA сервера

УДК 004.896:004.455.2

Студент

Группа	ФИО	Подпись	Дата
8ТМ91	Арбузов Роман Александрович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Курганов Василий Васильевич	к.т.н.		

Консультант

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ОАР ИШИТР	Цавнин Алексей Владимирович	-		

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Гончарова Наталья Александровна	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД ШБИП	Сечин Андрей Александрович	к.т.н.		

Консультант – лингвист отделения иностранных языков ШБИП:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИЯ ШБИП	Сидоренко Татьяна Валерьевна	к.п.н., доцент		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Гайворонский Сергей Анатольевич	к.т.н., доцент		

Томск – 2021 г.

Планируемые результаты освоения ООП

15.04.04 – Автоматизация технологических процессов и производств
(Киберфизическая автоматизация технологических процессов и производств)

Код компетенции	Наименование компетенции
Универсальные компетенции	
УК(У)-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий
УК(У)-2	Способен управлять проектом на всех этапах его жизненного цикла
УК(У)-3	Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели
УК(У)-4	Способен применять современные коммуникативные технологии, в том числе на иностранном (-ых) языке (-ах), для академического и профессионального взаимодействия
УК(У)-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия
УК(У)-6	Способен определять и реализовывать приоритеты собственной
Общепрофессиональные компетенции	
ОПК(У)-1	Способен формулировать цели и задачи исследования, самостоятельно изучать научно-техническую документацию своей профессиональной деятельности
ОПК(У)-2	Способен определить математическую и техническую сущность задач и провести их качественно-количественный анализ
ОПК(У)-3	Способен на основании статистических методов участвовать в проведении корректирующих и превентивных мероприятий, направленных на улучшение качества, интерпретировать и представлять результаты
ОПК(У)-4	Способен анализировать полученные результаты измерений на основе их физической природы и принимать обоснованные решения в области профессиональной деятельности
Профессиональные компетенции выпускников	
ПК(У)-1	Обладает способностью разрабатывать технические задания на модернизацию и автоматизацию действующих производственных и технологических процессов и производств, технических средств и систем автоматизации, управления, контроля, диагностики и испытаний, новые виды продукции, автоматизированные и автоматические технологии ее производства, средства и системы автоматизации, управления процессами, жизненным циклом продукции и ее качеством;
ПК(У)-2	Обладает способностью проводить патентные исследования с целью обеспечения патентной чистоты и патентоспособности новых проектных решений и определения показателей технического уровня проектируемой продукции, автоматизированных и автоматических технологических процессов и производств, средств их технического и аппаратно-программного обеспечения;

ПК(У)-3	Обладает способностью: составлять описание принципов действия и конструкции устройств, проектируемых технических средств и систем автоматизации, управления, контроля, диагностики и испытаний технологических процессов и производств общепромышленного и специального назначения для различных отраслей национального хозяйства, проектировать их архитектурно-программные комплексы;
ПК(У)-4	Обладает способностью разрабатывать эскизные, технические и рабочие проекты автоматизированных и автоматических производств различного технологического и отраслевого назначения, технических средств и систем автоматизации управления, контроля, диагностики и испытаний, систем управления жизненным циклом продукции и ее качеством с использованием современных средств автоматизации проектирования, отечественного и зарубежного опыта разработки
ПК(У)-5	Обладает способностью разрабатывать функциональную, логическую и техническую организацию автоматизированных и автоматических производств, их элементов, технического, алгоритмического и программного обеспечения на базе современных методов, средств и технологий проектирования;

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 – 15.04.04 Автоматизация технологических процессов и производств

Отделение школы (НОЦ) – Отделение автоматизации и робототехники

Период выполнения – весенний семестр 2020/2021 учебного года

Форма представления работы:

Магистерской диссертации

**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	60
21.05.2021	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	20
17.05.2021	Социальная ответственность	20

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Курганов Василий Васильевич	К.Т.Н.		

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Гайворонский Сергей Анатольевич	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 – 15.04.04 Автоматизация технологических процессов и производств
 Отделение школы (НОЦ) – Отделение автоматизации и робототехники

УТВЕРЖДАЮ:
 Руководитель ООП

 (Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

магистерской диссертации (бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8ТМ91	Арбузову Роману Александровичу

Тема работы:

Разработка клиента OPC UA сервера	
Утверждена приказом директора (дата, номер)	№ 124-7/с от 04.05.2021

Срок сдачи студентом выполненной работы:

--

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p>Исходные данные к работе <i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	<p>Объектом исследования является клиент OPC UA сервера. Необходимо осуществлять сбор и архивирование данных технологического процесса, поступающих с OPC UA сервера</p>
<p>Перечень подлежащих исследованию, проектированию и разработке вопросов <i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов,</i></p>	<ol style="list-style-type: none"> 1. Анализ спецификаций OPC. 2. Анализ научных статей, в которых описывается назначение в их работах OPC сервера. 3. Выбор спецификации OPC сервера. 4. Выбрать язык разработки OPC клиента. 5. Выбрать СУБД, с помощью которой будет осуществляться запись данных в базу

<i>подлежащих разработке; заключение по работе).</i>	данных. б. Реализовать клиент OPC UA сервера на выбранном языке программирования.
Перечень графического материала <i>(с точным указанием обязательных чертежей)</i>	Скриншоты результата работы программы
Консультанты по разделам выпускной квалификационной работы <i>(с указанием разделов)</i>	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Гончарова Наталья Александровна, Доцент отделения социально-гуманитарных наук, кандидат экономических наук
Социальная ответственность	Сечин Андрей Александрович, доцент отделения общетехнических дисциплин, кандидат технических наук
Раздел на иностранном языке	Сидоренко Татьяна Валерьевна, доцент отделения иностранных языков, кандидат педагогических наук
Названия разделов, которые должны быть написаны на русском и иностранном языках:	
Текущая ситуация Описание структуры адресного пространства OPC UA	

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

Задание выдал руководитель / консультант (при наличии):

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Курганов Василий Васильевич	к.т.н.		

Консультант:

Ассистент ОАР ИШИТР	Цавнин Алексей Владимирович			
---------------------	-----------------------------	--	--	--

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ТМ91	Арбузов Роман Александрович		

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

Студенту:

Группа	ФИО
8ТМ91	Арбузову Роман Александровичу

Школа	ИШИТР	Отделение школы (НОЦ)	
Уровень образования	Магистр	Направление/специальность	15.04.04 «Автоматизация технологических процессов и производств»

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

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Бюджет – 568 591,39 руб. Затраты на заработную плату – 318 744.625 руб. Прочие расходы – 199,68 руб.
2. <i>Нормы и нормативы расходования ресурсов</i>	Тариф на электроэнергию 2,56 кВт*ч; Накладные расходы 16 %; Районный коэффициент 30%; Норма амортизации ПЭВМ 33,33%;
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Налог во внебюджетные фонды 30% Районный коэффициент – 1,3 Накладные расходы – 16%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Предпроектный анализ</i>	Оценка потенциальных потребителей исследования, анализ конкурентных решений, SWOT – анализ, оценка готовности разработки к коммерциализации.
2. <i>Инициация проекта</i>	Цели и результат проекта, организационная структура проекта.
3. <i>Планирование управления научно-техническим проектом</i>	Иерархичная структура работ, планирование этапов работ, определение трудоемкости и построение календарного графика, формирование бюджета.
4. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	Интегральный показатель эффективности-4.05 Сравнительная эффективность проекта – 1,107

Перечень графического материала (с точным указанием обязательных чертежей)

1. Оценка конкурентоспособности для проекта
2. Матрица SWOT
3. Структура работ в рамках научного исследования Рабочая группа проекта
4. Календарный план-график выполнения проекта
5. Материальные затраты
6. Бюджет НИИ
7. Основные показатели эффективности ИП

Дата выдачи задания для раздела по линейному графику

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Гончарова Наталья Александровна	К.Э.Н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ТМ91	Арбузов Роман Александрович		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8ТМ91	Арбузов Роман Александрович

Школа	ИШИТР	Отделение (НОЦ)	ОАР
Уровень образования	Магистратура	Направление/специальность	15.04.04 «Автоматизация технологических процессов и производств

Тема ВКР:

Разработка клиента OPC UA сервера	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	<i>Объектом исследования является клиент OPC UA сервера. Данный клиент предназначен для использования при разработке SCADA систем и мониторинга текущих параметров значений сервера</i>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности: – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны.	<i>"Трудовой кодекс Российской Федерации" от 30.12.2001 N 197-ФЗ (ред. от 30.04.2021) ГОСТ 12.1.003-83 «ССБТ. Общие требования безопасности» ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования. СН 2.2.4/2.1.8.562-96 «Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки» СП 52.13330.2016 " Естественное и искусственное освещение" ГОСТ Р 50923-96 Дисплеи. Рабочее место оператора ГОСТ 12.1.019-2017 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты.</i>
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	<i>1. Отклонение показателей микроклимата 2. Превышение уровня шума 3. Недостаточная освещенность рабочей зоны 4. Повышенное значение электромагнитного излучения 5. Вероятность получения удара электрическим током</i>
3. Экологическая безопасность:	<i>Возможным воздействием разработки на окружающую среду является негативное воздействие на литосферу во время утилизации оргтехники.</i>
4. Безопасность в чрезвычайных ситуациях:	<i>Наиболее вероятной чрезвычайной ситуацией является возникновение пожара</i>

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

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД ШБИП	Сечин Андрей Александрович	К.Т.Н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ТМ91	Арбузов Роман Александрович		

Реферат

Пояснительная записка содержит 100 страниц машинописного текста, 23 таблицы, 27 рисунков, 1 список использованных источников из 34 наименований, 6 приложений.

Ключевые слова: OPC, OPC UA, MySQL, Python, OPC клиент

Целью данной выпускной квалификационной работы является упростить сбор, обработку и архивирование информации о параметрах протекания технологического процесса

Результатом работы является программное обеспечение

В данной работе был произведён обзор актуальности использования спецификации OPC и обзор различных спецификаций OPC. Было разработано программное обеспечение, позволяющее опрашивать OPC UA сервер, архивировать полученные данные и выводить график их изменения в реальном времени.

Содержание

Введение.....	13
1 Текущая ситуация.....	14
1.1 Предпроектный анализ.....	14
1.2 Обзор существующих спецификаций OPC серверов.....	16
1.3 Обзор литературы.....	18
2 Описание спецификации OPC UA.....	21
2.1 Описание принципа работы OPC UA сервера.....	21
2.2 Описание структуры адресного пространства OPC UA.....	23
3 Разработка клиента OPC UA сервера.....	28
3.1 Стек используемых технологий.....	28
3.2 Разработка базы данных.....	29
3.3 Описание используемых библиотек.....	30
3.4 Описание алгоритма работы программы.....	31
3.5 Апробация программы.....	35
4 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения.....	41
4.1 Потенциальные потребители результатов исследования.....	41
4.2 Анализ конкурентных технических решений.....	41
4.3 Оценка готовности разработки к коммерциализации.....	42
4.4 SWOT – анализ.....	44
4.5 Инициация проекта.....	46
4.6 Цели и результат проекта.....	46
4.7 Организационная структура проекта.....	47
4.8 Планирование научно – исследовательских работ.....	48
4.8.1 Иерархическая структура работ.....	48
4.8.2 План проекта.....	48
4.8.3 Определение трудоемкости выполнения работ.....	50
4.9 Бюджет научно-технического исследования (НТИ).....	53
4.9.1 Расчет материальных затрат НТИ.....	53

4.9.2	Расчет амортизационных отчислений	54
4.9.3	Расчет затрат на специальное оборудование для научных (экспериментальных) работ	54
4.9.4	Основная заработная плата исполнителей темы	55
4.9.5	Дополнительная заработная плата исполнителей темы	57
4.9.6	Отчисление во внебюджетные фонды.....	57
4.9.7	Прочие прямые затраты	58
4.9.8	Накладные расходы	58
4.9.9	Формирование бюджета затрат научно-исследовательского проекта	59
4.10	Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	59
5	Социальная ответственность.....	63
5.1	Правовые и организационные вопросы обеспечения безопасности	63
5.2	Производственная безопасность.	66
5.3	Анализ опасных и вредных производственных факторов.....	67
5.3.1	Микроклимат.....	67
5.3.2	Превышение уровня шума.....	68
5.3.3	Расчёт параметров освещённости.	69
5.3.4	Повышенное значение электромагнитного излучения.....	72
5.3.5	Вероятность получения удара электрическим током	73
5.4	Обоснование мероприятий по снижению уровней воздействия опасных и вредных факторов на исследователя.	73
5.5	Экологическая безопасность	74
5.6	Безопасность в чрезвычайных ситуациях	74
5.6.1	Анализ вероятных ЧС, которые может инициировать объект исследований	74

5.6.2 Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований	75
Выводы по разделу	77
Заключение	78
Список используемых источников литературы.....	79
Приложение А. (обязательное) Функция подключения к базе данных	82
Приложение Б. (обязательное) Функция подключения к OPC UA серверу.....	83
Приложение В. (обязательное) Программу запуска графического интерфейса	84
Приложение Г. (обязательное) Листинг используемых в проекте функций.....	86
Приложение Д. (обязательное) Таблица тэгов.....	87
Приложение Е. (справочное) OPC UA server client development	88

Введение

Наиболее важной, современной и широко распространённой спецификацией, определяющей передачу данных между оборудованием и автоматизированным рабочим местом оператора, является унифицированная архитектура OPC. Данная спецификация наиболее часто используется для передачи технологической информации от уровня программируемого логического контроллера к верхнему уровню автоматизированной системы управления технологическим процессом и в отдельности к SCADA-системам [1]. В частности, одной из наиболее популярных спецификаций OPC является OPC UA.

Объектом разработки является клиент, способный опрашивать OPC UA сервер и записывать полученные данные в базу данных MySQL. Данная разработка будет полезна при проведении исследований с малыми автоматизированными системами или в ходе отладки технологического процесса.

Целью данной работы является автоматизация процесса сбора и архивирования данных технологических процессов с использованием спецификации OPC.

Для достижения поставленной цели были поставлены следующие задачи, которые необходимы для разработки OPC клиента:

1. изучить спецификации OPC;
2. провести анализ научных статей, в которых описывается назначение в их работах OPC сервера;
3. выбрать спецификацию OPC сервера;
4. выбрать язык разработки OPC клиента;
5. выбрать СУБД, с помощью которой будут записываться данные в базу данных;
6. реализовать клиент OPC UA сервера на выбранном языке программирования.

1 Текущая ситуация

1.1 Предпроектный анализ

В ходе разработки автоматизированной системы управления установкой термохимической переработки резины появилось необходимость снимать показания датчиков температуры и сохранять историю их изменения. Вывод данных замеров на ПК производился с помощью программы TermodatNET [2], которая опрашивала самописец Термодат 25М/5 [3]. На рисунке 1 представлен интерфейс программы TermodatNET.

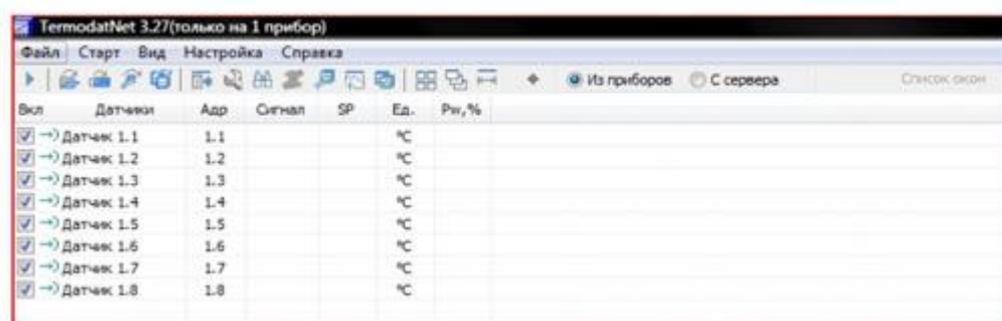


Рисунок 1 – Интерфейс программы ТермодатNET

TermodatNET также обладает функцией архивирования данных, экспорта сохранённых данных в БД MSSQL и вывода линии трендов на экран.

Функций данной программы было достаточно для поставленной задачи. Главным ограничением данной программы является работа только с оборудованием фирмы Термодат. В добавок к этому, данная программа – платная, и имеет ограничение на количество подключенных одновременно устройств.

Измерительный блок подключается при помощи собственного интерфейса к самописцу Термодат 25М/5. Данный самописец обладает следующими свойствами:

- архивирование данных во внутреннюю память самописца;
- настройка уставки температуры;
- настройка предупредительной и аварийной сигнализации;

- преобразование цифрового сигнала в значение температуры в зависимости от выбранного типа датчика температуры;
- вывод на экран самописца текущее показание температур по каждому измеряемому каналу;
- вывод графиков температуры по каждому датчику температуры непосредственно на экране самописца;
- передача данных самописца при помощи интерфейса RS-485 с помощью протоколов Modbus-RTU, Modbus-ASCII и проприетарного протокола Termodat.

Для подключения к ПК используется конвертер RS-485 – USB.

Данный самописец представлен на рисунке 2.



Рисунок 2 – Термодат-25М

В качестве тестового OPC сервера было выбрано ПО производства компании ИнСАТ «Multi-Protocol MasterOPC Server» [4]. Данный выбор был сделан на основе следующих соображений:

- поддержка протоколов Modbus RTU/ASCII/TCP;
- бесплатная неограниченная по времени работы версия, но ограниченная по количеству тэгов (32 тэга) версия;
- удобный интерфейс. Работа в режиме OPC DA (Data Access) и OPC UA (Unified Architecture) сервера.

В дальнейшем предполагается использование других датчиков, опрос которых программа TermodatNET осуществить не позволяет.

Данная проблема может быть решена при помощи использования SCADA системы. Но установка SCADA системы и её освоение требует достаточно много времени, а её возможности избыточны. В добавок к этому, большинство SCADA систем работает исключительно под управлением ОС Windows.

По причине того, что большинство SCADA систем получает данные с OPC серверов, было принято решение разработать OPC клиент, аналогичный по своим функциональным возможностям программе ТермодатNET. Данный клиент должен исключать недостаток в работе исключительно с оборудованием фирмы Термодат и не должен быть сложным в освоении.

1.2 Обзор существующих спецификаций OPC серверов

На сегодняшний день существует две основные спецификации технологии Open Platform Communication (OPC) – Classic [5], который включает в себя OPC Data Access (OPC DA), OPC Historical Data Access (OPC HDA), OPC Alarms & Events (OPC AE) и OPC Unified Architecture (OPC UA) [6].

OPC Classic в качестве основы использует технологию DCOM [7].

DCOM (Distributed Component Object Model), или распределенная модель компонентных объектов – это расширение модели COM фирмы Microsoft. Данное расширение ориентированно на поддержку и интеграцию функционирующих в сети распределенных объектных приложений.

Ключевым аспектом COM является обеспечение данной технологией связь между клиентами и серверами посредством интерфейсов. Именно интерфейс предоставляет клиенту способ опередить поддерживаемые на этапе выполнения технологии сервера. Для расширения возможностей сервера необходимо просто добавить новый интерфейс к существующим.

Несмотря на то, что OPC Classic приобрёл огромный успех, данная спецификация является устаревшей и обладает следующими недостатками:

- доступность данной технологии только на операционных системах семейства Windows;
- связь с технологией DCOM. Исходные коды данной технологии являются закрытыми, что не позволяет решать вопросы надежности ПО. Также это не позволяет выявлять и устранять возникающие программные отказы;
- возникновение трудностей конфигурирования, которые связаны с DCOM;
- неточные сообщения DCOM о прерываниях связи;
- непригодность DCOM для обмена данными через интернет;
- непригодность DCOM для обеспечения информационной безопасности.

В связи с указанными выше недостатками консорциумом OPC Foundation было принято решение разработать новую спецификацию, лишённую их. Новая спецификация получила унифицированную архитектуру и получила название OPC UA. Основными её достоинствами:

- реализация на языке программирования ANSI C для обеспечения переносимости на другие платформы, включая встраиваемые системы. Доступны также версии на .NET и Java, разработанные OPC Foundation;
- данная спецификация ориентирована на сервисы, а не на объекты. Это позволяет использовать спецификацию OPC UA на любых устройствах, использующих веб-сервисы;
- позволяет осуществить масштабирование OPC UA, т.е. изменение объема программы в зависимости от вычислительных ресурсов процессора и требуемой функциональности. Может быть выполнена также компиляция в виде однопоточного или многопоточного приложения;

- поддержка надежного и современного транспортного механизма SOAP (Simple Object Access Protocol) на базе XML (eXtensible Markup Language) с применением HTTP протокола;
- обеспечение высокой степени информационной безопасности;
- конфигурируемый таймаут для каждого сервиса;
- использование открытых стандартов World Wide Web Consortium (W3C) вместо закрытого стандарта COM/DCOM.

Стоит отметить, что на современных контроллерах установлена операционная система на базе ядра Linux, что и позволяет запускать на них OPC UA сервер.

Помимо этого, ряд производителей предлагает контроллеры, на которых уже установлен OPC UA сервер. Например, контроллер производства Овен ПЛК200, Siemens S7-1500 и другие. OPC UA сервер конфигурируется во время разработки проекта, после чего его можно опросить при помощи разработанного клиента.

Разрабатываемое программное обеспечение создаётся из расчёта того, что, как и OPC UA сервер, оно сможет запускаться на любой современной операционной системе.

1.3 Обзор литературы

В данном обзоре литературы рассматривались публикации и статьи, в которых применяются решения на основе OPC UA. В качестве источников информации были выбраны только зарубежные статьи, которые были опубликованы не позже 8 лет назад.

Рассматривая опыт применения OPC UA в зарубежных странах, был сделан вывод о разностороннем применении данной спецификации OPC. Например, в статье 2020 года описывается построение нейронной сети глубокого обучения на основе базы данных, полученной при помощи данных, собранных SCADA-системой на основе OPC сервера. Так как

прогнозирование в ветроэнергетике [8] представляет собой сложную задачу и зависит от многих параметров, прогностическая модель основывается на данные SCADA-системы с частотой измерения мгновенных значений в 1 с.

Также рассматривая OPC UA, стоит отметить роль данной спецификации в цифровизации производства и собственно Индустрии 4.0 [9], в которой явно выражена тенденция транспортировки информации при помощи Интернета. Также спецификация OPC UA является основным инструментом для обмена данных в сфере интернет вещей.

В дополнение к этому, спецификация OPC UA нашла своё применение в качестве человеко-машинного интерфейса на основе веб-платформы [10]. Архитектура разработанной системы состоит из клиента OPC UA, который позволяет веб-приложению взаимодействовать с сервером OPC UA. Сервер OPC UA отвечает за обмен данными с контроллером. Веб-интерфейс платформы позволяет создавать различные варианты человеко-машинных интерфейсов для мониторинга разных процессов.

Преимущества спецификации OPC UA как средства удаленного доступа [11] описываются и в статье по автоматизации сельскохозяйственной техники. Преимущество использования OPC UA заключается в том, что данные для удаленного доступа представляются в структурированном и согласованном виде, а не только в виде списка необработанных переменных. Результаты рассмотренной статьи опытным путем демонстрируют, что накладные расходы на использование данной спецификации относительно невелики. В свою очередь, небольшие накладные расходы имеют решающее значение для сохранения низкой стоимости передачи данных, более того, небольшая задержка обеспечивает поддержку различных приложения телеметрии реального времени.

Помимо этого, были рассмотрены статьи по характеристикам и возможностям спецификации OPC UA [12]. Ключевой информацией, полученной при обзоре статей данной направленности, является соответствие рассматриваемой спецификации большинству запросов, которые к нему

предъявляется согласно требованиям Индустрии 4.0. Более того, изучаются характеристики данной спецификации при различных режимах и параметрах настройки OPC сервера, что позволяет сделать выбор в сторону либо безопасности, либо производительности разрабатываемой системы на базе OPC UA.

Даты выхода статей подчеркивают актуальность рассматриваемой темы и показывает, что OPC серверы находят применения в различных проектах и системах. Особенно OPC серверы снискали популярность в качестве инструмента для формирования баз данных для машинного обучения глубокого обучения, где необходимо обеспечить большой объем информации с низкой частотой измерения параметров.

Исходя из всего вышеперечисленного, было принято решение использовать унифицированную архитектуру OPC UA в качестве основы.

2 Описание спецификации OPC UA

2.1 Описание принципа работы OPC UA сервера

В системе на базе OPC UA способно находиться одновременно несколько серверов и клиентов. Каждому клиенту предоставляется возможность работать одновременно с несколькими серверами. Серверу также даётся возможность отвечать на запросы одновременно нескольких клиентов [13]. На рисунке 3 представлена структура клиентской программы в OPC UA.

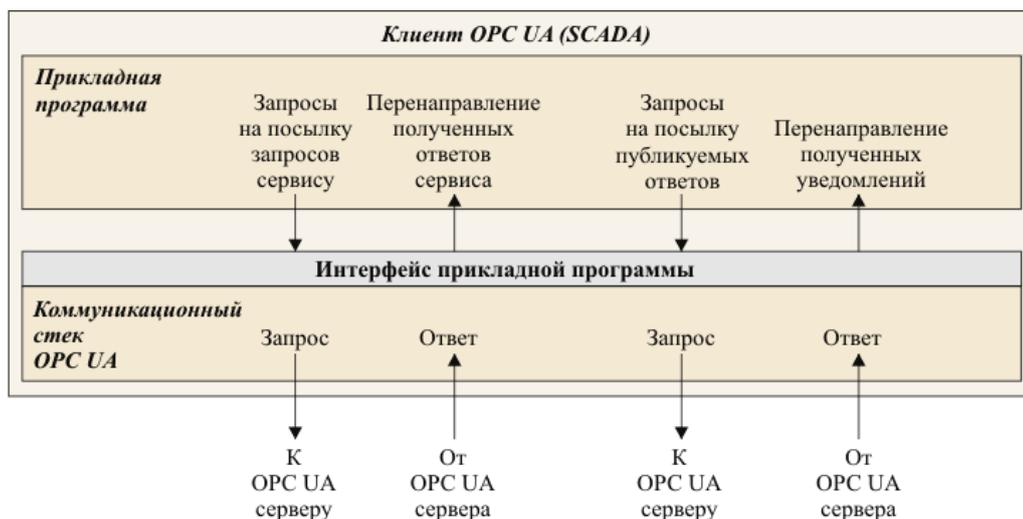


Рисунок 3 – Структура архитектуры клиентской программы в OPC UA

В качестве клиента OPC UA сервера выступает программа, опрашивающая его. Например, SCADA система. Отправка запросов на OPC сервер происходит с использованием внутреннего интерфейса, являющимся изолирующей прослойкой между клиентской программой и коммутационным стекком. Коммутационный стек выполняет роль преобразования запросов клиента в сообщения для вызова необходимого сервиса, посылаемого серверу. После получения ответа на отправленные запросы коммутационный стек передаёт результат опроса обратно в клиентскую программу.

Также архитектура OPC UA допускает обмен данными между двумя серверами. Для этого один из серверов становится клиентом, а другой выступает в роли сервера. Это позволяет объединить несколько серверов в

цепочку. При этом каждый из них будет выступать с одной стороны цепочки в качестве клиента, с другой стороны - в качестве сервера. На рисунке 4 представлена архитектура OPC UA сервера.

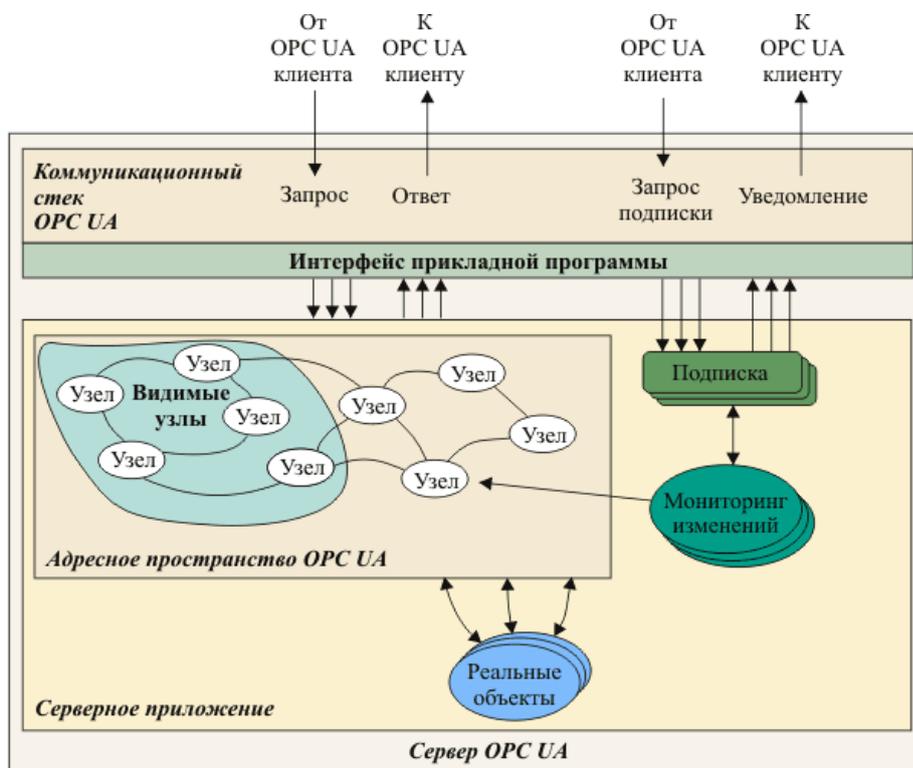


Рисунок 4 – Структура OPC UA сервера

Реальными объектами на данном рисунке обозначаются ПЛК, устройства ввода/вывода и другие, способные передавать данные с помощью OPC сервера.

Серверное приложение, представляет собой программную реализацию функций, которые должен выполнять сервер. Взаимодействие OPC UA сервера с клиентом выполняется через интерфейс прикладной программы, путем отправления запросов и получения ответов.

Обмен данными между сервером и клиентом может выполняться двумя способами:

- ответ на мгновенные запросы;
- по схеме «издатель – подписчик».

При обмене данными по схеме «издатель – подписчик» клиентская программа осуществляет «подписку» на получение определенных данных.

Сервер должен будет предоставить по мере их появления, т.е. по изменению этих значений. Для реализации режима подписки сервер осуществляет мониторинг узлов и соответствующих им реальных объектов с целью обнаружения изменений. При обнаружении изменений в данных, событиях или аварийных сигналах сервер генерирует уведомление, которое передается клиенту по каналу подписки.

2.2 Описание структуры адресного пространства OPC Unified Architecture

Основная цель адресного пространства OPC UA [14] - стандартизировать представление объектов сервера клиентам, которые осуществляют к нему запрос. Для достижения этой цели была разработана объектная модель OPC UA. Эта модель определяет объекты сервера в виде переменных и их методов. Она также описывает взаимодействие ними.

Кроме того, объекты могут быть типизированы. Другими словами, спецификация OPC UA предоставляет возможность определять и указывать как типы объектов (классы и переменные и их методы), так и их объектов.

При этом использование данного объектно-ориентированного подхода не является обязательным. Как и в OPC Data Access, простая модель адресного пространства может быть реализована с помощью объектов, папок и переменных. Но наличие дополнительных функций, присущих объектно-ориентированному подходу, значительно упрощает реализацию объектно-ориентированных систем с использованием спецификации OPC UA. На рисунке 5 представлена модель объекта OPC UA сервера.

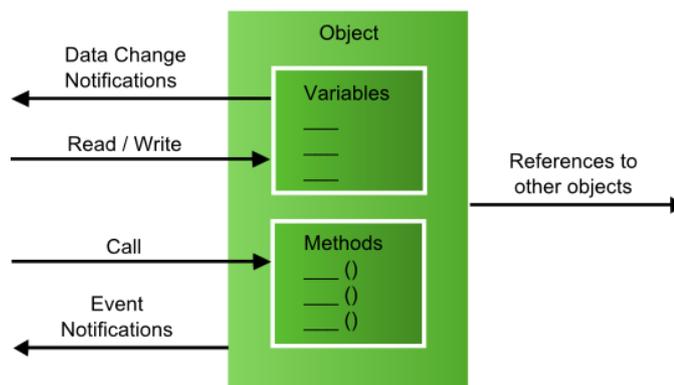


Рисунок 5 – Модель объекта OPC UA сервера

Модель объекта OPC UA

Службы UA используются для доступа к объектам и их компонентам. Например, они используются для чтения или записи значения переменной, вызова метода объекта или получения событий от него (например, события об изменении значения). Для анализа отношений между объектами и их компонентами возможно использовать службы просмотра. Данные службы выступают в качестве клиента OPC UA сервера.

Элементами данной модели в адресном пространстве являются узлы. Каждому узлу назначается класс, например, объект, переменная или метод объекта и представляет собой ещё один элемент модели объекта.

Модель узла

Объекты и связанная с ними информация, предоставляемые OPC UA сервером клиентам, называется его адресным пространством.

Объекты и их компоненты представлены в адресном пространстве как набор узлов, описываемые с помощью атрибутов и связанных ссылок. Модель узла представлена на рисунке 6.

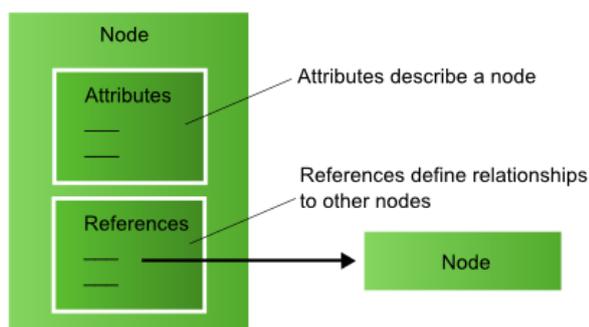


Рисунок 6 – Модель узла

Классы узлов

Для разных узлов атрибуты и ссылки определяются с помощью классов узлов. В спецификации OPC UA определено восемь различных классов узлов [15]:

- объект. Данный класс используется для отображения систем, их компонентов, реальных объектов и программного обеспечения;
- переменная. Данный класс служит для отображения содержания объектов;
- метод. Данный класс необходим для отображения различных методов в адресном пространстве;
- класс «Отображение» предназначен для указания видимости узлов и отношений между ними в адресном пространстве;
- класс «Тип объекта» описывает типы узлов объектов в адресном пространстве сервера;
- тип переменной. Описывает тип переменной, хранимой в адресном пространстве;
- тип отношения. Данный класс описывает типы отношений, используемых сервером;
- тип данных. Данный тип отображает абстрактность данных. Все типы данных представлены как узлы классов узлов в адресном пространстве. Имеет один атрибут – значения абстрактно или нет.

Каждый узел в адресном пространстве является экземпляром одного из классов узлов. Клиентам и серверам не разрешается определять дополнительные классы узлов или расширять определения этих классов узлов (например, список атрибутов для класса узла).

У каждого из классов есть свои атрибуты. Атрибуты — это элементы данных, описывающие узлы. Клиенты могут получить доступ к значениям атрибутов с помощью средств чтения, записи, запроса и подписки на изменения / отслеживания.

Атрибуты — это элементарные компоненты классов узлов. Каждое определение атрибута состоит из его идентификатора, имени, описания, типа данных и индикатора, отображающего, является ли данный атрибут обязательным. Определения атрибутов не видны в адресном пространстве, так как они являются частью определения класса узлов и известны клиенту. Набор атрибутов, определенных для каждого класса узла, является неизменным и не может быть расширен. При создании узла в адресном пространстве ему необходимо указать значения обязательных атрибутов класса данного узла.

Атрибуты предоставляют метаданные OPC UA, доступные для всех узлов. Атрибуты являются общими для всех узлов класса узла и определяются только спецификацией OPC UA.

Ссылки используются для связи узлов между собой. Доступ к ссылкам осуществляется с помощью служб просмотра и запросов.

Как и атрибуты, ссылки являются основным компонентом узлов. В отличие от атрибутов, ссылки определяются как экземпляры узлов класса «тип отношений». Узлы класса «тип отношений» видны в адресном пространстве и определяются с помощью класса «тип отношений».

Содержащий ссылку узел называется исходным узлом. Узел, на который делается ссылка — целевым. Комбинация исходного узла, типа отношений и целевого узла используется в службах OPC UA для уникальной идентификации ссылок. Таким образом, каждый узел может ссылаться на другой узел с тем же типом отношений только один раз.

Целевой узел ссылки может находиться как в адресном пространстве исходного узла, так и в адресном пространстве другого OPC UA сервера. Целевые узлы, расположенные на других серверах, идентифицируются в службах OPC UA при помощи комбинации имени удаленного сервера и присвоенного узлу удалённым сервером идентификатора.

Переменные используются для отображения значений. Существует два типа переменных: свойства и переменные данных. Они различаются

типом представляемых данных и возможностью содержать другие переменные.

Свойства содержат определенные сервером метаданные объектов, их переменных и других узлов. Свойства отличаются от атрибутов возможностью самостоятельного их добавления. Свойства характеризуют то, что представляет собой узел. Например, устройство или порядок опроса устройств.

Например, атрибут определяет тип данных переменной, тогда как свойство может использоваться для указания единиц измерения данной переменной.

С целью исключения рекурсии, свойствам запрещено использовать свойства, которые для них определены. Чтобы легко идентифицировать свойства, имя свойства должно быть уникальным в контексте узла, содержащего эти свойства.

Узел и свойства этого узла должны находиться на одном сервере.

Переменные данных определяют содержимое объекта. Объект — это контейнер для переменных и методов. Узел объекта не содержит значения в отличие от узлов переменных. Следовательно, узлы переменных данных используются для представления данных объекта.

Например, файл использует данные переменных для отображения содержимого файла в виде массива байтов. Его свойства можно использовать, чтобы указать время его создания и того, кто его создал.

3 Разработка клиента OPC UA сервера

3.1 Стек используемых технологий

В качестве языка разработки было принято решение использовать интерпретируемый язык Python [16]. Данный выбор сделан на основе того, что программа, написанная на этом языке, не требует предварительной компиляции и будет одинаково выполняться в любой ОС, для которой разработан интерпретатор Python. Также на языке Python написано большое количество библиотек, упрощающих разработку приложений.

Python написан на языке программирования C, что позволяет скомпилировать интерпретатор для любой операционной системы.

На сайте, с которого можно скачать интерпретатор, находятся скомпилированные интерпретаторы под операционные системы Windows, MacOS и операционные системы на базе ядра Linux.

В качестве сервера базы данных используется MySQL сервер. Он, также как и Python, является кроссплатформенным. MySQL сервер разработан под операционные системы Windows, MacOS и наиболее популярные операционные системы на базе ядра Linux.

Структурная схема опроса устройств ввода/вывода с помощью контроллера представлена на рисунке 7.

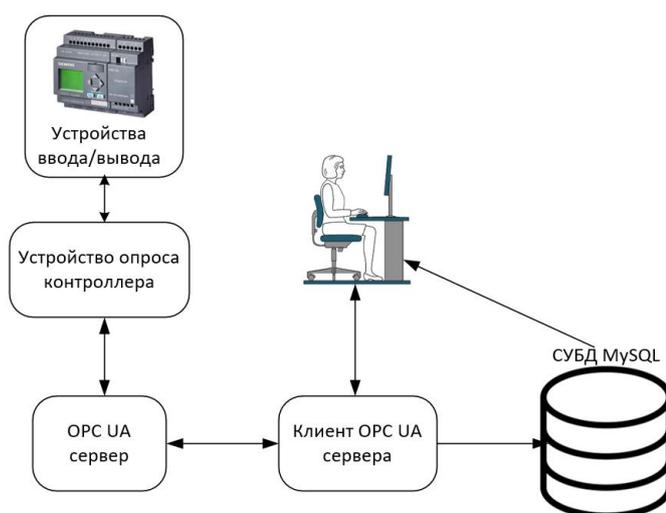


Рисунок 7 – Структурная схема

Основным же применением данного клиента можно описать в виде структурной схемы, представленной на рисунке 8.

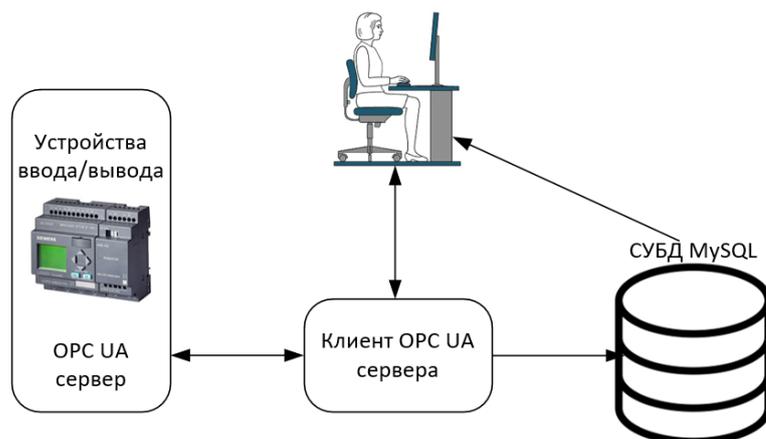


Рисунок 8 – Структурная схема

3.2 Разработка базы данных

Важным аспектом разрабатываемого программного обеспечения является работа с базой данных. Структура используемой базы данных представлена на рисунке 9.

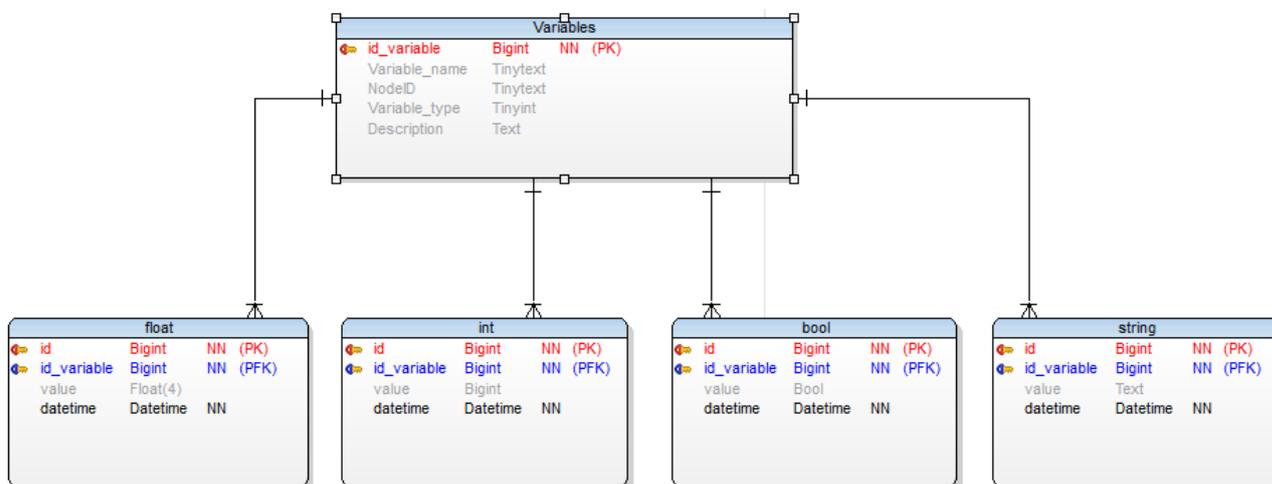


Рисунок 9 – Структура базы данных

Данная структура была выбрана в связи с тем, что сервер OPC UA поддерживает работу с различными типами данных. Изначальный вариант с использованием только одной таблицы и записью полученных данных в формате «string» является неоптимальным в связи с большим объёмом занимаемой информации у данного типа данных.

Также в разных узлах могут располагаться тэги с одинаковыми именами, что может привести к усложнению дальнейшего анализа собранной информации.

Выбор таблицы, в которую будет производиться запись, зависит от типа данных на сервере. В данную таблицу можно записать данные типа `integer`, `float`, `bool`. Данные типа `string` и остальные записываются в таблицу «string».

Сущность `Variables` содержит информацию о каждом уникальном записанном тэге. Атрибуты данной сущности:

- `id_variable` является первичным ключом и показывает порядковый номер тэга, хранимого в данной таблице;
- `Variable_name` – название тэга, значение которого записывается;
- `NodeID` – путь, по которому данный тэг хранится на сервере;
- `Variable_type` – тип переменной, установленный сервером;
- `Description` – описание переменной, установленное сервером.

Сущности `float`, `int`, `bool` и `string` имеют одинаковые атрибуты:

- `id` – первичный ключ, который указывает на номер записанного значения. Для него установлено свойство автоинкремент.
- `id_value` – внешний ключ, с помощью которого из сущности `Variable` можно получить информацию о тэге записываемого значения.
- `value` – непосредственно значение тэга, которое хранится в базе данных. В зависимости от сущности (`float`, `int`, `bool` или `string`) имеет свой тип данных.
- `datetime` – дата и время записи значения. Необходимо для дальнейшего анализа истории изменения записываемого значения.

3.3 Описание используемых библиотек

Для работы программы необходимо наличие следующих библиотек:

- `time`

- `mysql.connector`;
- `opcua`;
- `tkinter`;
- `stdiomask`;
- `matplotlib`.

Библиотека `time` [17] позволяет работать со временем.

Библиотека `mysql-connector-python` [18] позволяет осуществлять работу с MySQL сервером. Данная библиотека разработана непосредственно разработчиками MySQL. С помощью неё возможно осуществлять подключение к базам данных MySQL и выполнение SQL запросов с отображением полученных результатов.

Библиотека `opcua` [19] позволяет подключаться к OPC UA серверу и получать с него данные. Данная библиотека включает в себя как классы для разработки сервера OPC UA, так и классы разработки OPC UA клиента. В разработанном программном обеспечении используется только классы клиента OPC UA.

Стандартная библиотека `tkinter` [20] используется для создания графического интерфейса программы. С помощью неё на экран выводится структура сервера, данные отслеживаемых тэгов.

Библиотека `stdiomask` [21] используется для сокрытия вводимого пароля с целью недопущения его распространения.

Библиотека `matplotlib` [22] служит для вывода на экран графика изменения значений выбранного тэга.

3.4 Описание алгоритма работы программы

Перед запуском клиента необходимо выполнить следующий порядок действий:

1. Установить интерпретатор Python.

2. Установить и сконфигурировать MySQL сервер при его отсутствии.

3. Установить необходимые библиотеки при помощи менеджера пакетов `pip`. Данный менеджер пакетов устанавливается вместе с интерпретатором и позволяет устанавливать библиотеки при помощи команды «`pip install “название библиотеки”`» в командной строке или терминале (зависит от используемой операционной системы).

Сначала происходит подключение к OPC UA серверу. Листинг программы указан в приложении А. Данная часть программы отвечает за подключение к OPC UA серверу. От пользователя требуется ввести адрес расположения OPC UA сервера. При невозможности подключиться к нему программа выдаст ошибку и попросит ввести расположение сервера ещё раз.

Далее программа требует ввода логина, пароля, а также указать расположение MySQL сервера. После чего программа предлагает выбрать, к какой из существующих баз данных необходимо выполнить подключение. Если содержание выбранной базы данных не соответствует структуре приведённой выше таблицы, то произойдёт их автоматическая генерация. Листинг функции подключения к базам данных представлен в приложении Б

После указанных выше действий появляется окно с пользовательским интерфейсом, в котором и происходит вся работа пользователя. Листинг программы, запускающей графический интерфейс, представлен в приложении В. В приложении Г приведён листинг функций, отвечающий за наполнение раскрываемых элементов списка. Пользовательский интерфейс представлен на рисунке 10.

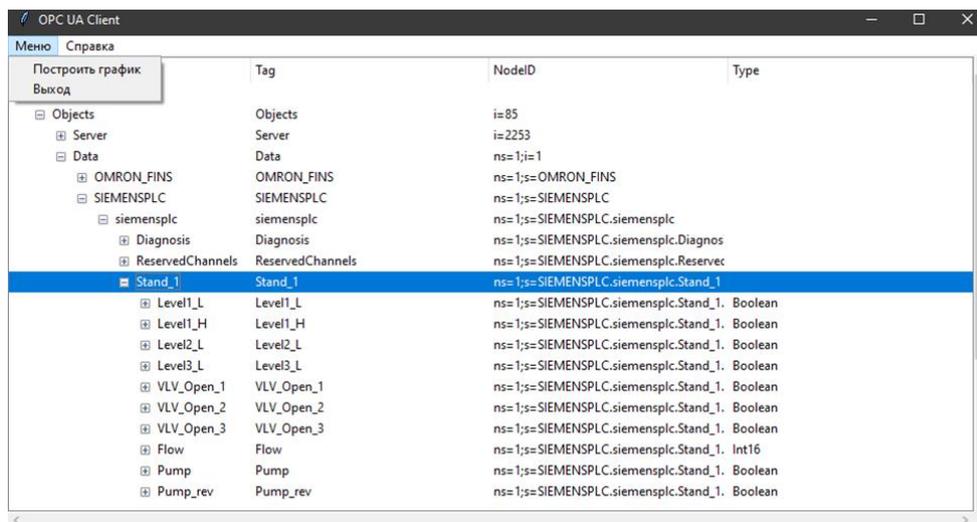


Рисунок 10 – Пользовательский интерфейс

После завершения работы программы выполняется безопасное отключение от OPC UA сервера и БД MySQL с помощью следующих команд:

```
cnx.close()
client.disconnect()
```

В приложении В представлен основной код программы, который запускает графический интерфейс.

На рисунке 11 продемонстрирован пример выводимых данных в командную строку при выполнении программы.

```
insert into bd_test VALUES('QualifiedName(1:Pump_rev)', 'False', '2021-03-21 15:58:17')
QualifiedName(1:Current_1) 0
insert into bd_test VALUES('QualifiedName(1:Current_1)', '0', '2021-03-21 15:58:17')
QualifiedName(1:Current_2) 0
insert into bd_test VALUES('QualifiedName(1:Current_2)', '0', '2021-03-21 15:58:17')
QualifiedName(1:Current_3) 0
insert into bd_test VALUES('QualifiedName(1:Current_3)', '0', '2021-03-21 15:58:17')
QualifiedName(1:Ust_1) 0
insert into bd_test VALUES('QualifiedName(1:Ust_1)', '0', '2021-03-21 15:58:17')
QualifiedName(1:Ust_2) 0
insert into bd_test VALUES('QualifiedName(1:Ust_2)', '0', '2021-03-21 15:58:17')
QualifiedName(1:Ust_3) 0
insert into bd_test VALUES('QualifiedName(1:Ust_3)', '0', '2021-03-21 15:58:17')
```

Рисунок 11 – Вывод данных программы

На рисунке 12 продемонстрированы экспортированные данные в базу данных.

QualifiedName(1:Start)	False	2021-03-21 15:58:27
QualifiedName(1:Level1_L)	False	2021-03-21 15:58:27
QualifiedName(1:Level1_H)	False	2021-03-21 15:58:27
QualifiedName(1:Level2_L)	False	2021-03-21 15:58:27
QualifiedName(1:Level3_L)	False	2021-03-21 15:58:27
QualifiedName(1:VLV_Open_1)	False	2021-03-21 15:58:27
QualifiedName(1:VLV_Open_2)	False	2021-03-21 15:58:27
QualifiedName(1:VLV_Open_3)	False	2021-03-21 15:58:27
QualifiedName(1:Flow)	0	2021-03-21 15:58:27
QualifiedName(1:Pump)	False	2021-03-21 15:58:27
QualifiedName(1:Pump_rev)	False	2021-03-21 15:58:27
QualifiedName(1:Current_1)	0	2021-03-21 15:58:27
QualifiedName(1:Current_2)	0	2021-03-21 15:58:27
QualifiedName(1:Current_3)	0	2021-03-21 15:58:27
QualifiedName(1:Ust_1)	0	2021-03-21 15:58:27
QualifiedName(1:Ust_2)	0	2021-03-21 15:58:27
QualifiedName(1:Ust_3)	0	2021-03-21 15:58:27
QualifiedName(1:Start)	False	2021-03-21 15:58:27
QualifiedName(1:Level1_L)	False	2021-03-21 15:58:27
QualifiedName(1:Level1_H)	False	2021-03-21 15:58:27
QualifiedName(1:Level2_L)	False	2021-03-21 15:58:27
QualifiedName(1:Level3_L)	False	2021-03-21 15:58:27
QualifiedName(1:VLV_Open_1)	False	2021-03-21 15:58:27
QualifiedName(1:VLV_Open_2)	False	2021-03-21 15:58:27
QualifiedName(1:VLV_Open_3)	False	2021-03-21 15:58:27
QualifiedName(1:Flow)	0	2021-03-21 15:58:27

Рисунок 12 – Экспортированные данные

На рисунке 13 продемонстрирована структура опрашиваемого OPC UA сервера, полученная при использовании клиента UaExpert [23].

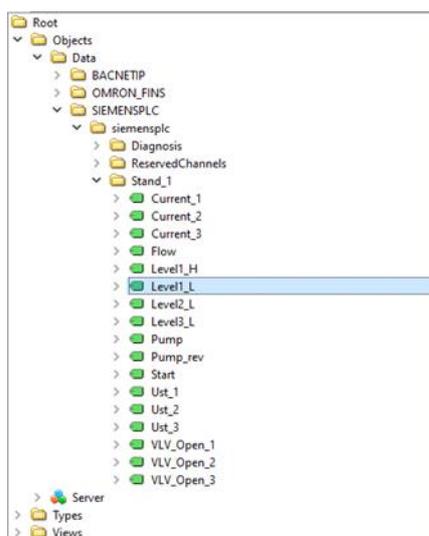


Рисунок 13 – Структура OPC сервера

С целью оценки скорости работы программы была написана функция, рекурсивно опрашивающая OPC UA сервер и выводит список расположенных на нём объектов.

Данный список содержит не только список тэгов, но и диагностическую информацию, которую хранит в себе сервер.

В качестве эксперимента были взяты OPC серверы от разработчиков «Инсат» и «UA Expert». OPC сервер «ИнСат» полностью опрашивается за 2 минуты 18 секунд. Тестовый сервер OPC «UA Expert» с намного большим количеством данных опрашивается за 9 секунд. Отсюда сделан вывод о том, что скорость работы программы ограничена скоростью работы OPC UA сервера.

3.5 Апробация программы

В качестве контроллера, на котором запускается OPC UA сервер, используется контроллер Овен ПЛК 200 [24], который представлен на рисунке 15.



Рисунок 14 – Овен ПЛК200

Ниже представлены его характеристики:

- Процессор ARM® Cortex-A8 с частотой 800 МГц.
- ROM 512 Мбайт (NAND).
- RAM 256 Мбайт (DDR3).
- RETAIN 64 Кбайт (MRAM).
- Операционная система Linux с RT-патчем.
- Поддержка быстрых входов/выходов до 95 кГц на выделенном

PRU

В качестве персонального компьютера используется системный блок. На нём установлена операционная система Windows 10 с установленным интерпретатором Python 3.9.5.

На данном компьютере также установлен MySQL Community Server 8.0.25.

Для проверки работы разработанного программного обеспечения была написана программа для ПЛК, регулирующая уровень жидкости в баке с помощью изменения уровня открытия задвижки, используя ПИД регулятор. Для программирования данного контроллера используется программное обеспечение CODESYS V3.5 SP14 Patch 3.

На рисунке 15 представлен функциональный блок основной программы.

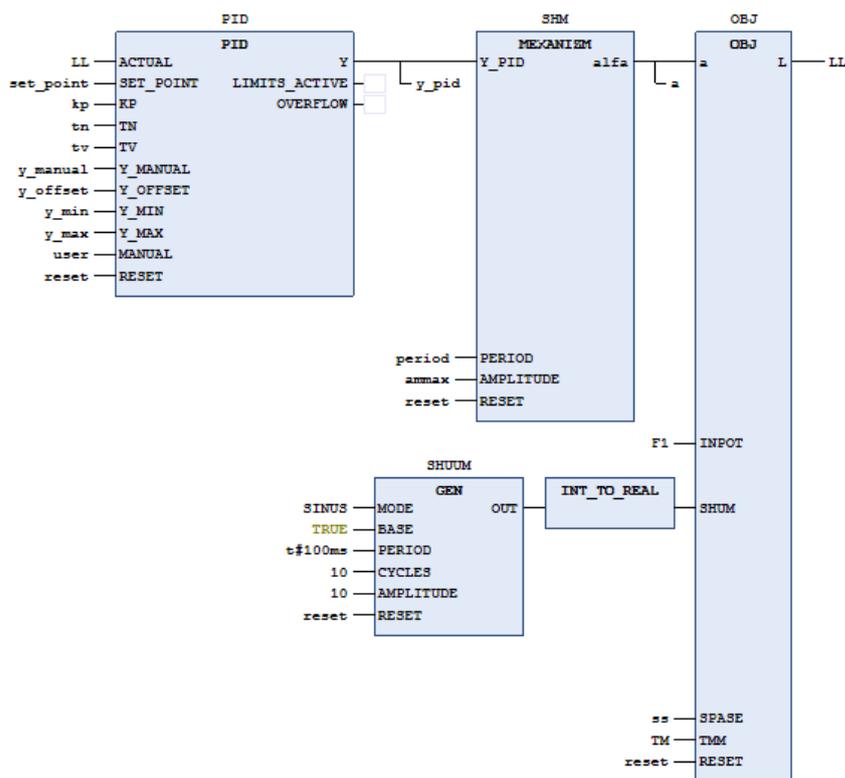


Рисунок 15 – Функциональный блок основной программы

На рисунке 16 представлена программа, регулирующая положение задвижки.

```

IF NOT RESET THEN
  IF OPEN=TRUE THEN
    a:=a-0.1;
  ELSIF CLOSE=TRUE THEN
    a:=a+0.1;
  END_IF;
  IF a<0 THEN
    a:=0;
  ELSIF a>1 THEN
    a:=1;
  END_IF;
ELSE
  a:=0;
END_IF;

```

Рисунок 16 – Программа управления положением открытия задвижки

На рисунке 17 представлен функциональный блок ШИМ, управляющий положением задвижки.

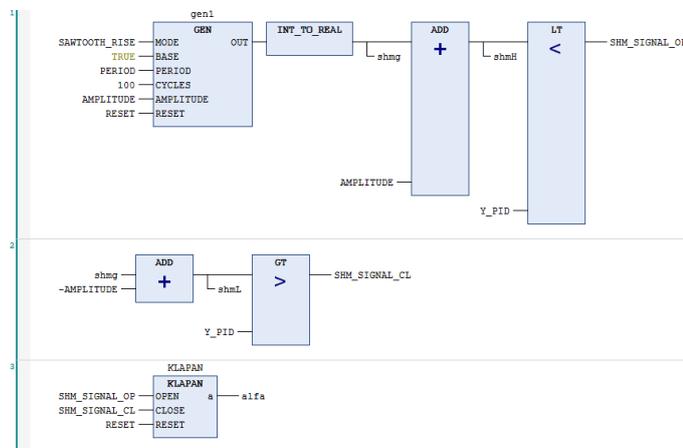


Рисунок 17 – Функциональный блок ШИМ

На рисунке 18 представлена основная программа.

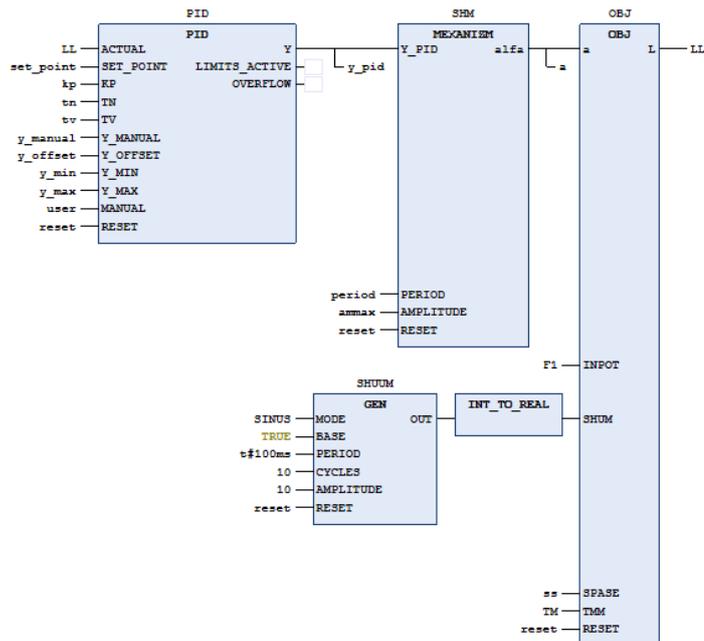


Рисунок 18 – Основная программа

При загрузке программы в контроллер также генерируется адресное пространство OPC UA сервера, установленного на нём. Данная программа запишет на сервер тэги, приведённые на рисунке 15.

Символы	Права ...	Мак...	Ат...	Тип	Члены	Комментарий
Constants						
IoConfig_Globals						
PLC_PRG						
<input checked="" type="checkbox"/> F1				REAL		
<input checked="" type="checkbox"/> LL				REAL		
<input type="checkbox"/> OBJ				OBJ	...	
<input type="checkbox"/> PID				PID	...	
<input type="checkbox"/> SHM				MEXANIZM	...	
<input type="checkbox"/> SHUUM				GEN	...	
<input checked="" type="checkbox"/> TM				DWORD		
<input checked="" type="checkbox"/> a				REAL		
<input checked="" type="checkbox"/> ammax				INT		
<input type="checkbox"/> close				BOOL		
<input checked="" type="checkbox"/> kp				REAL		
<input checked="" type="checkbox"/> open				BOOL		
<input checked="" type="checkbox"/> period				TIME		
<input checked="" type="checkbox"/> reset				BOOL		
<input checked="" type="checkbox"/> set_point				REAL		
<input checked="" type="checkbox"/> ss				REAL		
<input checked="" type="checkbox"/> tn				REAL		
<input checked="" type="checkbox"/> tv				REAL		
<input checked="" type="checkbox"/> user				BOOL		
<input checked="" type="checkbox"/> y_manual				REAL		
<input checked="" type="checkbox"/> y_max				REAL		
<input checked="" type="checkbox"/> y_min				REAL		
<input checked="" type="checkbox"/> y_offset				REAL		
<input checked="" type="checkbox"/> y_pid				REAL		

Рисунок 19 – Записываемые на OPC UA сервер тэги

В приложении Г приведена таблица всех используемых тэгов в проекте.

Была разработана экранная форма, отображающая основные параметры регулирования и текущий уровень жидкости. С помощью данной экранной формы можно указать требуемый уровень жидкости, который будет обработан программой. Данная мнемосхема представлена на рисунке 20.

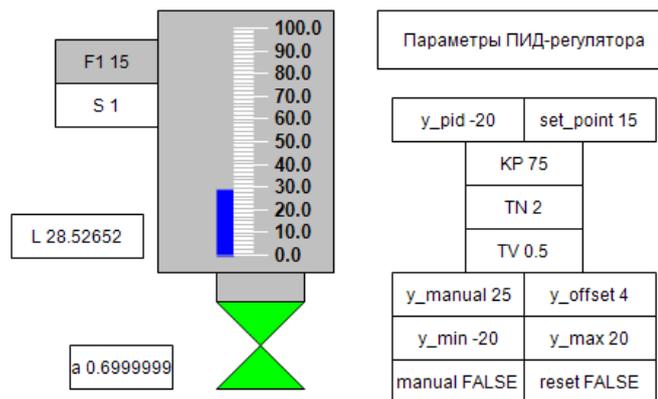


Рисунок 20 – Мнемосхема уровня жидкости в баке

Теперь произведём подключение к OPC UA серверу, сгенерированному на контроллере. На рисунке 21 представлено адресное пространство сервера.

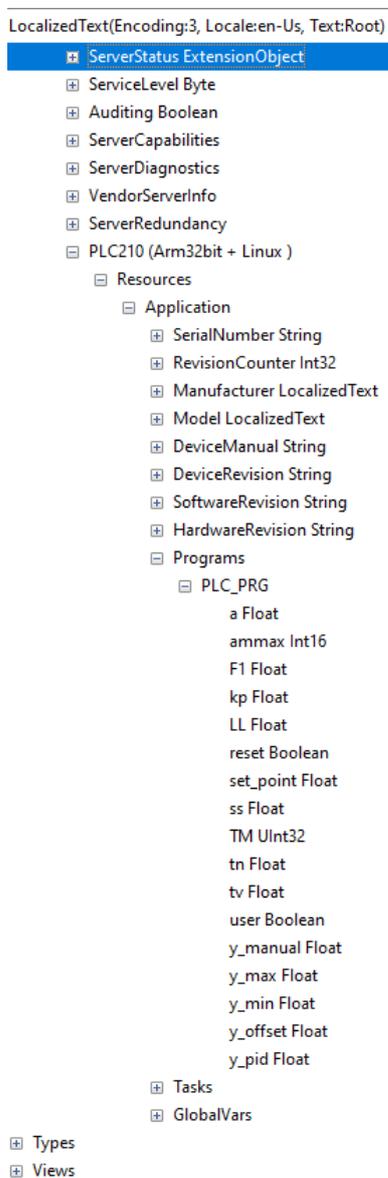


Рисунок 21 – Адресное пространство сервера

Как мы видим, типы данных на сервере и на контроллере различаются. Типы данных в ПЛК были приведены к типам данных, используемых на OPC UA сервере.

Теперь произведём запись значений в базу данных.

Запишем в таблицу переменные, используемых в проекте. Результат записи переменных представлен на рисунке 22.

🔑 id_variable	Variable_name	NodeID	Variable_type	Description
1	a	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.a	Float	
2	ammax	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.ammax	Int16	(NULL)
3	F1	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.F1	Float	(NULL)
4	kp	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.kp	Float	(NULL)
5	LL	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.LL	Float	(NULL)
6	reset	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.reset	Boolean	(NULL)
7	set_point	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.set_point	Float	(NULL)
8	ss	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.ss	Float	(NULL)
9	TM	ns=4;s= var PLC210 (Arm32bit + Linux).Application.PLC_PRG.TM	UInt32	(NULL)

Рисунок 22 – Результат записи полученных значений

Как мы можем увидеть, регистрируемые показания записываются в различные таблицы в зависимости от их типа данных.

На рисунке 23 представлен график изменения значений, который строится в разработанном клиенте OPC UA сервера.

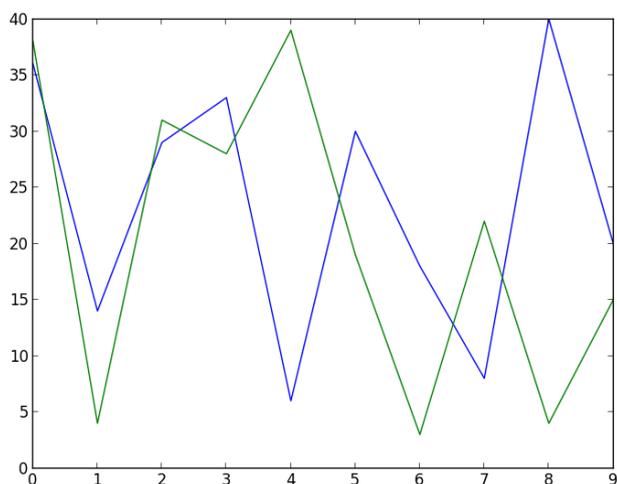


Рисунок 23 – График выбранного значения

Во время работы клиент потребляет 60 Мбайт оперативной памяти при выполнении программы в операционной системе Windows 10.

4 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения

4.1 Потенциальные потребители результатов исследования

Потенциальными потребителями результатов исследований являются организации, специализирующиеся на разработке SCADA систем, а также предприятия, их эксплуатирующие. Данная разработка также полезна при сборе данных о работе технологических установок при проведении научных испытаний. Карта сегментирования представлена в таблице 1.

Таблица 1 – Карта сегментирования

		Область применения		
		Научные лаборатории	Разработка SCADA систем	Эксплуатация SCADA систем
Размер компании	Крупные			
	Средние			
	Мелкие			

Согласно карте сегментирования, можно выбрать следующие сегменты рынка: использование клиента OPC UA сервера для научных лабораторий малого размера, а также малых, средних и крупных предприятий, проектирующих и эксплуатирующих SCADA системы.

4.2 Анализ конкурентных технических решений

Детальный анализ конкурирующих разработок, существующих на рынке, необходимо проводить систематически, поскольку рынки пребывают в постоянном движении. Такой анализ помогает вносить коррективы в научное исследование, чтобы успешнее противостоять своим соперникам.

Важно реалистично оценить сильные и слабые стороны разработок конкурентов.

Среди конкурентных разработок можно выделить клиент сервера от компании ИнСат (конкурент 1) и UaExpert (конкурент 2). Данный анализ приведён в таблице 2.

Таблица 2 – Оценочная карта для сравнения конкурентных технических решений

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б _ф	Б _{к1}	Б _{к2}	К _ф	К _{к1}	К _{к2}
1	2	3	4	5	6	7	8
Технические критерии оценки ресурсоэффективности							
Удобство эксплуатации	0,15	5	3	2	0,75	0,45	0,3
Быстродействие	0,1	5	2	4	0,5	0,2	0,4
Стабильность работы	0,2	5	3	3	1	0,6	0,6
Потребность в ресурсах памяти	0,1	4	5	4	0,4	0,5	0,4
Функциональная пригодность	0,1	5	3	4	0,5	0,3	0,4
Качество интерфейса	0,1	5	3	2	0,5	0,3	0,2
Экономические критерии оценки ресурсоэффективности							
Конкурентоспособность продукта	0,1	5	2	4	0,5	0,2	0,4
Цена	0,1	5	5	5	0,5	0,5	0,5
Послепродажное обслуживание	0,05	4	4	2	0,2	0,2	0,1
Итого	1				4,85	3,25	3,3

Исходя из анализа технических решений и сравнения их с конкурентами, был сделан вывод, что разработка данного клиента OPC UA сервера имеет ряд преимуществ перед конкурентами. К основным плюсам можно отнести удобство в эксплуатации, качество интерфейса и стабильность работы.

4.3 Оценка готовности разработки к коммерциализации

Одной из важных задач в ходе выполнения данного раздела является оценка готовности разработки к коммерциализации. Оцениваемыми параметрами являются как научная, так и коммерческая составляющая.

Таблица 3 представляет собой бланк оценки степени готовности разработки к коммерциализации.

Таблица 3 – Бланк оценки степени готовности научного проекта к коммерциализации

		Степень проработанности разработки	Уровень имеющихся знаний у разработчика
1	Определен имеющийся научно-технический задел	4	4
2	Определены перспективные направления коммерциализации научно-технического задела	2	2
3	Определены отрасли и технологии (товары, услуги) для предложения на рынке	5	5
4	Определена товарная форма научно-технического задела для представления на рынок	4	3
5	Определены авторы и осуществлена охрана их прав	3	5
6	Проведена оценка стоимости интеллектуальной собственности	4	5
7	Проведены маркетинговые исследования рынков сбыта	3	4
8	Разработан бизнес-план коммерциализации научной разработки	2	3
9	Определены пути продвижения научной разработки на рынок	3	4
10	Разработана стратегия (форма) реализации научной разработки	5	5
11	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	1	1
12	Проработаны вопросы использования услуг инфраструктуры поддержки, получения льгот	1	1
13	Проработаны вопросы финансирования коммерциализации научной разработки	4	3
14	Имеется команда для коммерциализации научной разработки	4	3
15	Проработан механизм реализации разработки	5	5
	Итоговый балл	50	53

Данный программный продукт предполагается коммерциализировать посредством продажи лицензии на его использование, поскольку он не является разработкой под конкретный проект и имеет универсальное применение в области АСУ ТП. Продукты конкурентов распространяются аналогичным способом.

4.4 SWOT – анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – представляет собой комплексный анализ научно-исследовательского проекта. SWOT-анализ применяют для исследования внешней и внутренней среды проекта. Он проводится в несколько этапов. Результирующая матрица SWOT представлена в таблице 4.

Таблица 4 – SWOT анализ

	<p>Сильные стороны:</p> <p>С1. Возможность модификации.</p> <p>С2. Работа на различных операционных системах.</p> <p>С3. Использование удобных функций мониторинга и сохранения значений технологических параметров.</p> <p>С4. Тестирование работы на реальном объекте.</p> <p>С5. Использование последней спецификации программного обеспечения</p>	<p>Слабые стороны:</p> <p>Сл1. Требование навыков программирования для запуска проекта.</p> <p>Сл2. Отсутствие длительной проверки работоспособности программы.</p> <p>Сл3. Скорость работы программы зависит от скорости работы сервера, к которой она подключается</p>
<p>Возможности:</p> <p>В1. Переход на использование OPC UA серверов большего числа разработчиков.</p> <p>В2. Повышение стоимости конкурентных разработок.</p> <p>В3. Внедрение OPC UA серверов в большее количество проектов. установленным OPC UA сервером.</p>	<p>Разрабатываемое программное обеспечение позволяет работать на большинстве используемых сегодня операционных системах. Дополнительный набор возможностей (сохранение результатов в базу данных и вывод графиков на экран) делает его наиболее универсальным предложением из существующих на рынке.</p>	<p>Переход разработчиков автоматизированных систем на более спецификацию OPC UA, обучение базовым навыкам программирования и использование современного оборудования позволит доработать программу и использовать её максимально эффективно</p>

Продолжение таблицы 4 – SWOT анализ

<p>В4. Применение контроллеров с В5. Выход на международный рынок В6. Усиление роли автоматизации.</p>		
<p>Возможности: В1. Переход на использование OPC UA серверов с OPC Classic среди большего числа разработчиков SCADA систем. В2. Повышение стоимости конкурентных разработок. В3. Внедрение SCADA систем в большее количество проектов. В4. Использование контроллеров, внутри которых уже установлен OPC UA сервер В5. Выход на международный рынок В6. Усиление роли автоматизации при проведении исследований технологических процессов</p>	<p>Разрабатываемое программное обеспечение позволяет работать на большинстве используемых сегодня операционных системах. Дополнительный набор возможностей (сохранение результатов в базу данных и вывод графиков на экран) делает его наиболее универсальных предложением из всех на рынке.</p>	<p>Переход разработчиков автоматизированных систем на более спецификацию OPC UA, обучение базовым навыкам программирования и использование современного оборудования позволит доработать программу и использовать её максимально эффективно</p>
<p>Угрозы: У1. Нежелание разработчиков переходить с устаревшей спецификации OPC Classic на более современную OPC UA. У2. Устаревшее ПО на устройствах заказчика. У3. Низкая производительность оборудования, неспособного обработать запросы клиента с заданной частотой.</p>	<p>Переход большего числа разработчиков SCADA систем на более современную спецификацию снизит затраты на его внедрение, что приведёт к росту его популярности и увеличению числа пользователей, которые будут использовать разрабатываемое программное обеспечение. Также более количество пользователей позволит протестировать программное обеспечение в различных условиях и указать на недостатки программы и легко их доработать благодаря открытому коду программы.</p>	<p>Повышение квалификации персонала сведёт риск возникновения неполадок программы к минимум. Удобство разрабатываемой программы подтолкнёт разработчиков использовать современную спецификацию OPC, что также повлечёт за собой обновление оборудования, на котором программа будет работать.</p>

4.5 Инициация проекта

Группа процессов инициации состоит из процессов, которые выполняются для определения нового проекта или новой фазы существующего. В рамках процессов инициации определяются изначальные цели и содержание и фиксируются изначальные финансовые ресурсы. Определяются внутренние и внешние заинтересованные стороны проекта, которые будут взаимодействовать и влиять на общий результат научного проекта. Данная информация закрепляется в Уставе проекта.

4.6 Цели и результат проекта

В данном пункте приведена информация о заинтересованных сторонах проекта. В таблице 5 представлены цели и результаты проекта.

В таблице 6 представлены заинтересованные стороны проекта.

Таблица 5 – Цели и результаты проекта

Цели проекта:	Разработка клиента OPC UA сервера
Ожидаемый результат проекта:	1) Подключение и опрос тэгов OPC UA сервера. 2) Реализация алгоритма экспорта данных в базу данных и вывод графиков изменения значения параметра. 3) Описание работы программы.
Критерии приемки результата разработки:	Система должна включать в себя: 1) Удобный графический интерфейс. 2) Программа выполняет подключение к OPC UA серверу и серверу MySQL 3) Данные архивируются и выводятся на экран
Требования к результату разработки:	Требования:
	Возможность запуска программы на различных операционных системах.
	Возможность в реальном времени выводить график изменения состояния технологического параметра.
	Возможность архивирования собранных данных для дальнейшего анализа в распространённом формате, который позволит с лёгкостью эти данные извлечь.

Таблица 6 – Заинтересованные стороны проекта

Заинтересованные стороны	Ожидания заинтересованных сторон
Научные лаборатории	Возможность сбора и анализа данных в удобном для исследователя формате.
Предприятия, разрабатывающие SCADA системы	Разработанная программа, позволяющая упростить отладку SCADA систем
Предприятия, эксплуатирующие SCADA системы	Разработанная программа, позволяющая отследить текущее состояние технологического параметра и историю его изменения

4.7 Организационная структура проекта

На данном этапе работы решены следующие вопросы: кто будет входить в рабочую группу данного проекта, а также определена роль каждого участника в данном проекте и прописаны функции, выполняемые каждым из участников и их трудозатраты в проекте.

В таблице 7 приведена рабочая группа проекта.

Таблица 7 – Рабочая группа проекта

№ п/п	ФИО, основное место работы, должность	Роль в разработке	Функции	Трудозатраты, час.
1	Курганов Василий Васильевич, ТПУ доцент ОАР	Научный руководитель	Постановка технического задания, установка сроков выполнения работ, координация деятельности исполнителя	35
2	Арбузов Роман Александрович, ТПУ ОАР ИШИТР, магистрант гр. 8ТМ91	Исполнитель	Выполнение поставленных задач, разработка системы	280
Итого:				315

Исходя из приведенных выше данных, видно, что выполняемая работа требует больших трудозатрат. Заинтересованные стороны проекта ожидают готовую систему с высокими показателями.

4.8 Планирование научно – исследовательских работ

4.8.1 Иерархическая структура работ



Рисунок 24 – Иерархическая структура работ по проведению разработки

Данный проект разделен на 3 основных блока:

- предварительные исследования;
- реализация клиента OPC UA сервера;
- отладка и тестирование проекта.

Основная часть работы заключается в разработке алгоритмов опроса OPC UA сервера и создании графического интерфейса программы.

4.8.2 План проекта

В рамках планирования научного проекта построим календарный и сетевой графики проекта.

Календарный план проекта представлен в таблице 8.

Таблица 8 – Календарный план проекта

Код работы	Название	Длительность, дни	Дата начала работ	Дата окончания работ	Состав участников
1	Выбор направления исследования	5	20.10.2020	24.10.2020	Арбузов Р.А.
2	Выбор темы научного исследования	10	25.10.2020	03.11.2020	Курганов В.В. Арбузов Р.А.
3	Подбор и изучение материалов по теме	25	04.11.2020	28.11.2020	Арбузов Р.А.
4	Анализ отобранного материала	15	29.11.2020	13.12.2020	Курганов В.В. Арбузов Р.А.
5	Подготовка среды разработки	15	14.12.2020	28.12.2020	Арбузов Р.А.
6	Подбор необходимых библиотек для разработки	20	29.12.2020	17.01.2021	Арбузов Р.А.
7	Разработка алгоритмов опроса OPC UA сервера	40	18.01.2021	26.02.2021	Арбузов Р.А.
8	Разработка графического интерфейса программы	40	27.02.2021	07.04.2021	Арбузов Р.А.
9	Отладка программы	15	08.04.2021	22.04.2021	Курганов В.В. Арбузов Р.А.
10	Оценка полученных результатов	15	23.04.2021	07.05.2021	Курганов В.В. Арбузов Р.А.
11	Составление пояснительной записки	15	08.05.2021	22.05.2021	Арбузов Р.А.
12.	Подготовка презентации дипломного проекта	5	23.05.2021	27.05.2021	Арбузов Р.А.
Итого:		220			

4.8.3 Определение трудоемкости выполнения работ

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

Трудоемкость выполнения научного исследования оценивается экспертным путем в человеко-днях и носит вероятностный характер, т.к. зависит от множества трудно учитываемых факторов. Для определения ожидаемого (среднего) значения трудоемкости $t_{ожі}$ используется следующая формула (1):

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

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

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

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

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях T_p , учитывающая параллельность выполнения работ несколькими исполнителями. Она вычисляется по формуле (2)

$$T_{p_i} = \frac{t_{ожі}}{Ч_i}, \quad (2)$$

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

$t_{ожі}$ – ожидаемая трудоемкость выполнения одной работы, чел.-дн.

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

Представим 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 (3):

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

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 (4):

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

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 9.

Таблица 9 – Временные показатели проведения научного исследования

Название работы	Трудоёмкость работ						Длительность работ в рабочих днях T_{pi}		Длительность работ в календарных днях T_{ki}	
	t_{min} , чел-дни		t_{max} , чел-дни		$t_{ож}$, чел-дни					
	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель
Выбор направления исследования	3		7		4,6	0	4,6		6	
Выбор темы научного исследования	8	8	12	12	9,6	9,6	4,8	4,8	6	6
Подбор и изучение материалов по теме	22		30	30	25,2	12	12,6	6	16	8
Анализ отобранного материала	13	13	20	20	15,8	15,8	7,9	7,9	10	10
Подготовка среды разработки	10		20		14	0	14		18	

Продолжение таблицы 9 – Временные показатели проведения научного исследования

Название работы	Трудоёмкость работ						Длительность работ в рабочих днях T_{pi}		Длительность работ в календарных днях T_{ki}	
	t_{min} , чел-дни		t_{max} , чел-дни		$t_{ож}$, чел-дни					
	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель
Подбор необходимых библиотек	15		23		18,2	0	18,2		23	
Разработка алгоритмов опроса OPC UA сервера	30		50		38	0	38		47	
Разработка графического интерфейса программы	30		50		38	0	38		47	
Отладка программы	12	12	20	20	15,2	15,2	7,6	7,6	10	10
Оценка полученных результатов	7	7	20	20	12,2	12,2	6,1	6,1	8	8
Составление пояснительной записки	7		20		12,2	0	12,2		15	
Подготовка презентации дипломного проекта	2		7		4	0	4		5	
Итого							168	32,4	211	42

На основе полученной таблицы 9 строится календарный план-график. График строится для максимального по длительности исполнения работ с разбивкой по месяцам и декадам. График работ приведен на рисунке 25.

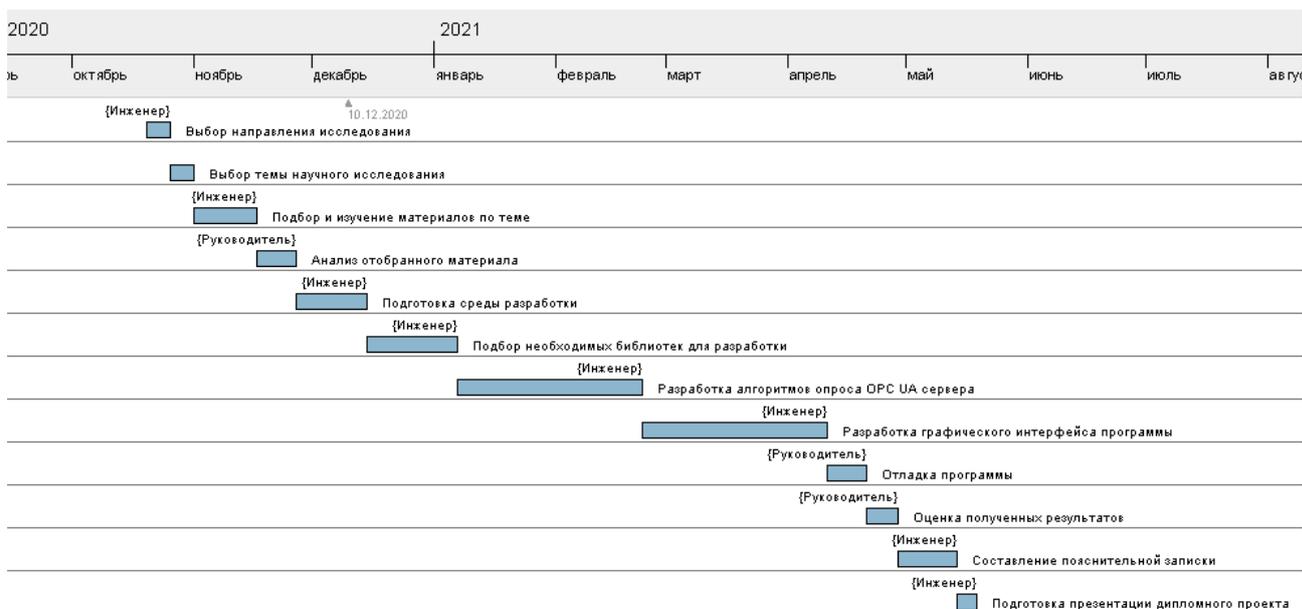


Рисунок 25 – График проведения разработки

4.9 Бюджет научно-технического исследования (НТИ)

4.9.1 Расчет материальных затрат НТИ

Расчет материальных затрат осуществляется по формуле (7):

$$Z_m = (1 + k_T) \cdot \sum_{i=1}^m C_i \cdot N_{расхi} , \quad (7)$$

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

$N_{расхi}$ – количество материальных ресурсов i -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м, м² и т.д.);

C_i – цена приобретения единицы i -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м, руб./м² и т.д.);

k_T – коэффициент, учитывающий транспортно-заготовительные расходы.

В таблице 10 представлены материальные затраты на проведение научного исследования

Таблица 10 – Затраты на научные исследования

Наименование	Количество	Цена за ед., руб.	Стоимость
Ноутбук Dell Vostro 5568	1	30000	30000
Мышь	1	1500	1500
Коврик для мыши	1	300	300
Итого:		31800	

4.9.2 Расчет амортизационных отчислений

Произведем расчет амортизационных отчислений. Первоначальная стоимость ноутбука магистранта, который используется для работы, составляет 30 000 рублей. Срок полезного использования данной машины – 3 года, из которых 10 месяцев машина использовалась для написания ВКР и написания программного обеспечения.

Норма амортизации рассчитывается по формуле (6):

$$A_n = \frac{1}{n} \cdot 100\% = \frac{1}{3} \cdot 100\% = 33,33\%. \quad (6)$$

Годовые амортизационные отчисления вычисляются по формуле (7):

$$A_n = 30\,000 \cdot 0,33 = 10\,000 \text{ рубль}. \quad (7)$$

Ежемесячные амортизационные отчисления рассчитываются по формуле (8):

$$A_n = \frac{10\,000}{12} = 833 \text{ руб.} \quad (8)$$

Таким образом затраты на амортизацию ПК составляют 10 000 рублей.

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

В данную статью включают все затраты, связанные с приобретением специального оборудования (приборов, контрольно-измерительной аппаратуры, стендов, устройств и механизмов), необходимого для проведения работ по конкретной теме. Определение стоимости

спецоборудования производится по действующим прейскурантам, а в ряде случаев по договорной цене. Расчет затрат по данной статье заносится в таблицу 11.

Таблица 11 – Затраты на оборудование для научных работ

№ п/п	Наименование оборудования	Кол-во единиц оборудования	Цена единицы оборудования, руб.	Общая стоимость оборудования, руб.
1.	Microsoft Office	1	5 990	5 990
2.	Microsoft Windows	1	9 390	9 390
Итого:				15 380

4.9.4 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 от 20 до 30 % от тарифа или оклада.

Статья включает основную заработную плату работников, непосредственно занятых выполнением НТИ, (включая премии, доплаты) и дополнительную заработную плату, которые считаются по формуле (9):

$$Z_{зп} = Z_{осн} + Z_{доп}, \quad (9)$$

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

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

Основная заработная плата ($Z_{осн}$) руководителя (лаборанта, инженера) от **предприятия** (при наличии руководителя от предприятия) рассчитывается по формуле (10):

$$Z_{осн} = Z_{дн} \cdot T_p, \quad (10)$$

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

T_p – продолжительность работ, выполняемых научно-техническим работником, раб. дн. (табл. 8);

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

Среднедневная заработная плата рассчитывается по формуле (11):

$$Z_{дн} = \frac{Z_m \cdot M}{F_d}, \quad (11)$$

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, рабочих дней. Данные приведены в таблице 12.

Таблица 12 – Баланс рабочего времени

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

Месячный должностной оклад работника считается по формуле (12):

$$Z_m = Z_{тс} \cdot (1 + k_{пр} + k_d) \cdot k_p, \quad (12)$$

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

$k_{пр}$ – премиальный коэффициент, равный 0,3 (т.е. 30% от $Z_{тс}$);

k_d – коэффициент доплат и надбавок составляет примерно 0,2 – 0,5 (в НИИ и на промышленных предприятиях – за расширение сфер обслуживания, за профессиональное мастерство, за вредные условия: 15-20 % от $Z_{тс}$);

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

Основная заработная плата **руководителя** (от ТПУ) рассчитывается на основании отраслевой оплаты труда. Отраслевая система оплаты труда в ТПУ предполагает следующий состав заработной платы:

1. оклад – определяется предприятием. В ТПУ оклады распределены в соответствии с занимаемыми должностями, например, ассистент, ст. преподаватель, доцент, профессор. Базовый оклад Z_6 определяется исходя из размеров окладов, определенных штатным расписанием предприятия;

2. стимулирующие выплаты – устанавливаются руководителем подразделений за эффективный труд, выполнение дополнительных обязанностей и т.д.;

3. иные выплаты; районный коэффициент.

Таблица 13 – Расчёт основной заработной платы

Исполнители	Оклад	кпр	кд	кр	Зм, руб	Здн, руб.	Тр, раб. дн.	Зосн, руб.
Руководитель	33 664	-	-	1.3	43 763.2	1 813.3	33,9	61 470,87
Инженер	21 760	-	-	1.3	28 288	1 172.09	219.5	257 273,75
Итого								318 744,62

4.9.5 Дополнительная заработная плата исполнителей темы

Расчет дополнительной заработной платы ведется по формуле (13):

$$Z_{доп} = k_{доп} \cdot Z_{осн}, \quad (13)$$

где $k_{доп}$ – коэффициент дополнительной заработной платы (0,12).

Получим:

$$Z_{доп НР} = 0,12 \cdot 61\,470,87 = 7\,376,5 \text{ рублей.}$$

$$Z_{доп И} = 0,12 \cdot 257\,273,75 = 30\,872,85 \text{ рублей.}$$

4.9.6 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 (14):

$$Z_{внеб} = k_{внеб} * (Z_{осн} + Z_{доп}), \quad (14)$$

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

Расчет отчислений приведен в таблице 14.

Таблица 14 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.	Дополнительная заработная плата, руб.	Коэффициент отчислений	Отчисления
Руководитель	61 470,87	7 376,5	0,3	20 654,21
Инженер	257 273,75	30 872,85		86 443,98
Итого				107 098,19

4.9.7 Прочие прямые затраты

К прочим прямым затратам относятся затраты на электроэнергию. Стоимость 1 кВт составляет 2.56 р. Используемый ноутбук потребляет в среднем 65 Вт в час. Всего за этим ноутбуком было проведено 200 дней по 6 часов.

$$Z_{эн} = 200 \cdot \frac{2.56}{1000} \cdot 6 \cdot 65 = 199,68 \text{ руб.}$$

4.9.8 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии, почтовые и телеграфные расходы, размножение материалов и т.д. Их величина определяется по следующей формуле (15):

$$C_{накл} = k_{накл} \cdot (Z_{осн} + Z_{доп}), \quad (15)$$

где $k_{\text{накл}}$ – коэффициент накладных расходов, 16 %.

Получим:

$$Z_{\text{накл}} = 0,16 \cdot (318\,744,625 + 38\,249,85) = 57\,119,036 \text{ рублей}$$

4.9.9 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 на научно-исследовательский проект по каждому варианту исполнения приведен в таблице 15.

Таблица 15 –Расчёт бюджета затрат НИИ

Наименование статьи	Сумма, руб.
Материальные затраты НИИ	31 800
Затраты на специальное оборудование для научных (экспериментальных) работ	15 380
Затраты по основной заработной плате исполнителей темы	318 744,63
Затраты по дополнительной заработной плате исполнителей темы	38 249,85
Отчисление во внебюджетные фонды	107 098,19
Прочие расходы	199,68
Накладные расходы	57 119,04
Бюджет затрат НИИ	511 472,35

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

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его

нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный показатель финансовой эффективности научного исследования получают в ходе оценки бюджета затрат трех (или более) вариантов исполнения научного исследования, который приведён в таблице 16. Для этого наибольший интегральный показатель реализации технической задачи принимается за базу расчета (как знаменатель), с которым соотносятся финансовые значения по всем вариантам исполнения.

Интегральный финансовый показатель разработки определяется по формуле (16):

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}}, \quad (16)$$

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 (в т.ч. аналоги).

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в разах (значение больше единицы), либо соответствующее численное удешевление стоимости разработки в разах (значение меньше единицы, но больше нуля).

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить по формуле (17):

$$I_{pi} = \sum a_i \cdot b_i, \quad (17)$$

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.

Расчет интегрального показателя ресурсоэффективности рекомендуется проводить в форме таблицы. Данный показатель рассчитан в таблице 16.

Таблица 16 – Сравнительная оценка характеристик вариантов исполнения проекта

Критерии	Весовой коэф.	Исп. 1	Исп. 2	Исп. 3	I_{p1}	I_{p2}	I_{p3}
1.Объём программы	0,05	2,00	3,00	5,00	0,1	0,15	0,25
2.Надежность	0,10	4,00	4,00	3,00	0,4	0,4	0,3
3.Скорость работы	0,25	5,00	2,00	3,00	1,25	0,5	0,75
4.Удобство интерфейса	0,15	5,00	3,00	4,00	0,75	0,45	0,6
5.Использование вычислительных ресурсов	0,25	3,00	5,00	5,00	0,75	1,25	1,25
6. Удобство сопровождения	0,20	4,00	5,00	2,00	0,8	1	0,4
Итого	1,00	26,00	22,00	21,00	4,05	3,75	3,55

Интегральный показатель эффективности вариантов исполнения разработки определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле (18):

$$I_{исп.1} = \frac{I_{p-исп1}}{I_{финр}^{исп.1}} \quad (18)$$

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}) рассчитывается по формуле (19):

$$\mathcal{E}_{cp} = \frac{I_{исп.1}}{I_{исп.2}} \cdot \quad (19)$$

В таблице 17 приведены рассчитанные показатели эффективности.

Таблица 17 – Сравнительная эффективность разработки

Показатель	Исп. 1	Исп. 2	Исп. 3
Интегральный финансовый показатель	1	0,93	0,97
Интегральный показатель ресурсоэффективности	4,05	3,75	3,55
Интегральный показатель эффективности	4,05	4,03	3,66
Сравнительная эффективность вариантов исполнения	1	1,005	1,107

5 Социальная ответственность

Ни одна область человеческой деятельности не может быть не сопряжена с воздействием вредных и опасных факторов. Одной из главных задач на сегодняшний день является снижение влияния этих факторов на организм человека и окружающую природу.

В настоящей работе представлено решение по разработке OPC UA клиента. Данная разработка позволяет получить доступ к OPC UA серверу, опросить его и сохранить полученные данные в базу данных

Цель работы – разработка клиента OPC UA сервера, позволяющего упростить мониторинг технологических параметров. Объект исследования – клиент OPC UA сервера.

В качестве рабочего места, рассматривается рабочее место разработчика SCADA систем, оснащенное различной техникой, такой как монитор, системный блок, клавиатура, мышь, и т.д.

5.1 Правовые и организационные вопросы обеспечения безопасности

Трудовые отношения между работником и работодателем регулируются Трудовым кодексом Российской Федерации [25]. В нём описываются основные нормы и правила, которые обязаны соблюдать рабочий и работодатель в случае возникновения трудовых отношений.

График работы персонала должен соответствовать 16 главе трудового кодекса, регламентирующей режим рабочего времени.

Данная разработка, как правило, применяется разработчиками SCADA систем. Разработчики данных систем работают по пятидневной рабочей неделе

Согласно ТК РФ, N 197-ФЗ ст. 219 работник имеет право на:

– получение достоверной информации от работодателя, соответствующих государственных органов и общественных организаций об условиях и охране труда на рабочем месте, о существующем риске

повреждения здоровья, а также о мерах по защите от воздействия вредных и (или) опасных производственных факторов;

- обязательное социальное страхование от несчастных случаев на производстве и профессиональных заболеваний в соответствии с федеральным законом;

- обеспечение средствами индивидуальной и коллективной защиты в соответствии с требованиями охраны труда за счет средств работодателя;

- обучение безопасным методам и приемам труда за счет средств работодателя;

- личное участие или участие через своих представителей в рассмотрении вопросов, связанных с обеспечением безопасных условий труда на его рабочем месте, и в расследовании происшедшего с ним несчастного случая на производстве или профессионального заболевания;

- внеочередной медицинский осмотр в соответствии с медицинскими рекомендациями с сохранением за ним места работы (должности) и среднего заработка во время прохождения указанного медицинского осмотра;

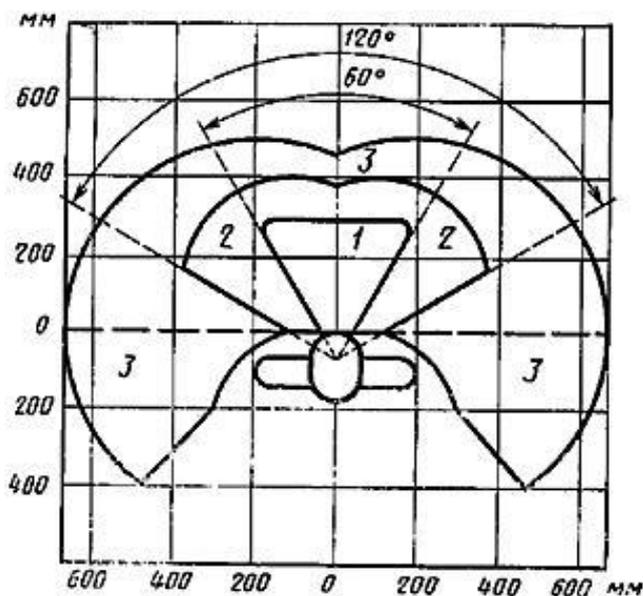
- отказ от выполнения работ в случае возникновения опасности для его жизни и здоровья вследствие нарушения требований охраны труда, за исключением случаев, предусмотренных федеральными законами, до устранения такой опасности;

- гарантии и компенсации, установленные в соответствии с настоящим Кодексом, коллективным договором, соглашением, локальным нормативным актом, трудовым договором, если он занят на работах с вредными и (или) опасными условиями труда.

- повышенные или дополнительные гарантии и компенсации за работу на работах с вредными и (или) опасными условиями труда могут устанавливаться коллективным договором, локальным нормативным актом с учетом финансово-экономического положения работодателя.

Безопасность труда работника регламентируется нормативным документом под названием «Система стандартов безопасности труда» (ССБТ). Использование разработанного программного обеспечения подразумевает работу с персональным компьютером. В данном случае рабочее место сотрудника регулируется ГОСТ 12.2.032-78 [26]. Оптимальным является размещение предметов труда и документации следующим образом:

- Моноблок со встроенным дисплеем располагается в центральной части зоны 3;
- Клавиатура располагается в зоне 1;
- Мышь располагается в зоне 2 справа;
- документация, необходимая при работе – в зоне легкой досягаемости ладони – 3, а в выдвижных ящиках стола – литература, неиспользуемая постоянно.



1 – зона выполнения ручных операций 1; 2 – зона выполнения ручных операций 2; 3 – зона выполнения ручных операций 3.

Рисунок 26 – Зоны для выполнения ручных операций и размещения органов управления

Конструкция кресла позволяет регулировать положение сиденья по вертикали.

5.2 Производственная безопасность.

Основная работа при проведении исследования проводилась за компьютером. Данный вид работы связан с воздействием на человека вредных и опасных факторов труда. Классификация вредных и опасных факторов проведена в соответствии с ГОСТ 12.1.003-83 [27]. Возможные опасные и вредные факторы при выполнении магистерской диссертации на разных этапах работ отражены в таблице 18.

Таблица 18 – Возможные опасные и вредные факторы

Факторы (ГОСТ 12.0.003-2015)	Этапы работ			Нормативные документы
	Проектирование	Разработка	Эксплуатация	
1.Отклонение показателей микроклимата	+	+	+	ГОСТ 12.1.003-83 «ССБТ. Общие требования безопасности» [27] СНиП 23-03-2003 «Защита от шума» [28] СанПиН 2.2.4.548-96 «Гигиенические требования к микроклимату производственных помещений» [29]
2.Превышение уровня шума	+	+	+	СП 52.13330.2016 «Естественное и искусственное освещение» [30] ГОСТ Р 50923-96 Дисплеи. Рабочее место оператора [31]
3.Недостаточная освещенность рабочей зоны	+	+	+	
4.Повышенное значение электромагнитного излучения	+	+	+	ГОСТ 12.1.019-2017 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты. [32]
5.Вероятность получения удара электрическим током	+	+	+	

5.3 Анализ опасных и вредных производственных факторов.

5.3.1 Микроклимат

Микроклимат определяется действующими на организм человека сочетаниями температуры, относительной влажности и скорости движения воздуха. Отклонения показателей микроклимата может привести гипо- или гипертермии организма. Эти отклонения появляются вследствие изменения температуры окружающей среды и влажности воздуха.

В таблице 19 представлены оптимальные показатели микроклимата. Оптимальные микроклиматические условия установлены по критериям оптимального теплового и функционального состояния человека.

По степени физической тяжести работа разработчика SCADA систем относится к категории лёгких работ.

Таблица 19 – Оптимальные показатели микроклимата

Период года	Категория работ по уровням энергозатрат, Вт	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	Ia (до 139)	(22 – 24)	(21 – 25)	(40 – 60)	0,1
Тёплый	Ia (до 139)	(23 – 25)	(22 – 26)	(40 – 60)	0,1

Допустимые микроклиматические условия установлены по критериям допустимого теплового и функционального состояния человека на период 8-часовой рабочей смены. Они не вызывают повреждений или нарушений состояния здоровья, но могут приводить к возникновению общих и локальных ощущений теплового дискомфорта, напряжению механизмов терморегуляции, ухудшению самочувствия и понижению работоспособности. Допустимые величины показателей микроклимата устанавливаются в случаях, когда по технологическим требованиям, техническим и экономически обоснованным причинам не могут быть обеспечены оптимальные величины. Допустимые параметры микроклимата представлены в таблице 20.

Таблица 20 – Допустимые параметры микроклимата

Период года	Категория работ	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
		Ниже оптимальных не более	Выше оптимальных не более			Ниже оптимальных не более	Выше оптимальных не более
Холодный	Ia	(20,0 - 21,9)	(24,1 - 25,0)	(19,0 - 26,0)	(15 – 75)	0,1	0,1
Тёплый	Ia	(21,0 - 22,9)	(25,1 - 28,0)	(20,0 - 29,0)	(15 – 75)	0,1	0,2

С целью снижения воздействия отклонений показателей микроклимата необходимо установить кондиционеры для снижения температуры и увлажнитель воздуха для поддержания оптимальной влажности. При понижении температуры ниже допустимой необходимо включить обогреватели.

5.3.2 Превышение уровня шума

Источником возникновения является источники активного охлаждения ПЭВМ, серверного оборудования, работа вентилятора и т.д. Превышение уровня шума приводит к раздражительности, головным болям, головокружению, снижению памяти, повышенной утомляемости, понижению аппетита, боли в ушах и т. д.

Допустимые значения звукового давления представлены в таблице 21.

Таблица 21 – Допустимые уровни звукового давления

Помещения и рабочие места	Уровень звукового давления, дБ, в октавных полосах со среднегеометрическими частотами, Гц					Уровень звука, дБА
	63	12	26	10	4000	
Помещения управления, рабочие комнаты	79	70	68	55	50	60

Для снижения уровня шума применяют: подавление шума в источниках; звукоизоляция и звукопоглощение; увеличение расстояния от источника шума; рациональный режим труда и отдыха.

5.3.3 Расчёт параметров освещённости.

Недостаточная освещённость рабочей зоны приводит к снижению зрения рабочего и уменьшению внимания. Недостаточная освещённость может быть результатом неправильного расположения источников искусственного света и неправильной планировки окон.

На рабочей поверхности должны отсутствовать резкие тени, которые создают неравномерное распределение поверхностей с различной яркостью в поле зрения, искажает размеры и формы объектов различия.

Используются три вида освещения: естественное (от солнца), искусственное (от искусственных источников), совмещенное. При недостатке естественного света нарушаются обмен веществ и резистентность организма, поэтому используются газоразрядные источники, близкие по спектру к естественному свету.

Дано помещение с размерами: длина $A = 10$ м, ширина $B = 6$ м, высота $H = 3,5$ м. Высота рабочей поверхности $h_{рп} = 0,8$ м. Требуется создать освещённость $E = 300$ лк.

Исходя из того, что потолок в помещении чистый белый, а также побеленные стены, 3 окна, закрытых светлыми жалюзи от прямых солнечных лучей. Согласно таблице 22 примем коэффициенты отражения от стен $R_c=50\%$ и потолка $R_p=70\%$. Рассчитываем систему общего люминесцентного освещения.

Таблица 22 –Значение коэффициентов отражения потолка и стен

Состояние потолка	$\rho_n, \%$	Состояние стен	$\rho_{ст}, \%$
Свежепобелённый	70	Свежепобелённые с окнами, закрытыми шторами	70
Побелённый, в сырых помещениях	50	Свежепобелённые с окнами без штор	50
Чистый бетонный	50	Бетонные с окнами	30
Светлый деревянный (окрашенный)	50	Оклеенные светлыми обоями	30
Бетонный грязный	30	Грязные	10
Деревянный	30	Кирпичные неоштукатуренные	10

неокрашенный			
Грязный (кузница, склады)	10	С тёмными обоями	10

Выбираем светильники типа ОД, $\lambda = 1,4$.

Приняв $h_c = 0,05$ м, определяем расчетную высоту по формуле (20):

$$h = H - h_c - h_{\text{рп}} = 3,5 - 0,05 - 0,8 = 2,65 \text{ м.} \quad (20)$$

Расстояние между светильниками рассчитывается по формуле (21):

$$L = \lambda \cdot h = 1,4 \cdot 2,65 = 3,71 \text{ м.} \quad (21)$$

Расстояние от крайнего ряда светильников до стены:

$$L/3 = 1,24 \text{ м.}$$

Определим количество рядов светильников по формуле (22):

$$n_{\text{ряд}} = \frac{(B - \frac{2}{3}L)}{L} + 1 = \frac{(6 - \frac{2}{3} \cdot 3,71)}{3,71} + 1 \approx 2. \quad (22)$$

Определим количество светильников в ряду по формуле (23):

$$n_{\text{ряд}} = \frac{(A - \frac{2}{3}L)}{l_{\text{св}} + 0,5} = \frac{(10 - \frac{2}{3} \cdot 3,71)}{1,23 + 0,5} \approx 4. \quad (23)$$

Размещаем светильники в два ряда. В каждом ряду можно установить 4 светильника типа ОД мощностью 40 Вт (с длиной 1,23 м), при этом разрывы между светильниками в ряду составят 50 см. Изображаем в масштабе план помещения и размещения на нем светильников. Схема размещения представлена на рисунке 27. Учитывая, что в каждом светильнике установлено две лампы, общее число ламп в помещении $N = 8$.

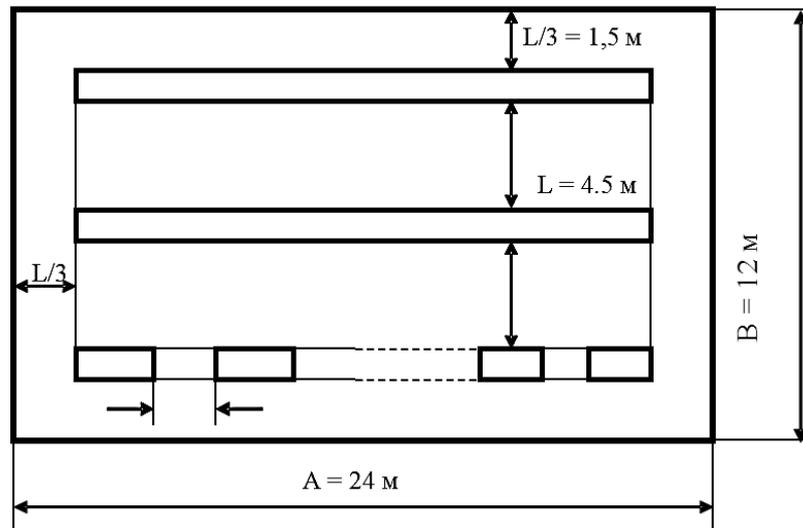


Рисунок 27 – Схема расположения светильников

Находим индекс помещения по формуле (25):

$$i = S / h(A+B) = 60 / (2,65(10+6)) = 1,4. \quad (25)$$

Определяем коэффициент использования светового потока:

$$\eta = 0,51.$$

Определяем потребный световой поток лампы в каждом из рядов по формуле (26):

$$\Phi = \frac{En * S * K_z * Z}{N_l * \eta} = \frac{300 * 60 * 1,5 * 1,1}{12 * 0,51} = 4853. \quad (26)$$

Выбираем ближайшую стандартную лампу – ЛБ 65 Вт с потоком 4600 лм. Делаем проверку выполнения условия по формуле (27):

$$-10\% \leq \frac{\Phi_{л.станд} - \Phi_{л.расч}}{\Phi_{л.расч}} * 100\% \leq +20\% \quad (27)$$

$$\text{Получаем: } -10\% \leq -5,21\% \leq +20\%$$

Определяем электрическую мощность осветительной установки:

$$P = N * P_{л} = 8 * 40 = 320 \text{ Вт,}$$

где: N – число светильников,

$P_{л}$ – мощность светильника.

Нормы освещенности определяются основными признаками зрительной работы: размер различаемого объекта (буквы, штрих рисунка и т.д.), коэффициент отражения фона, контраст между объектом и фоном. На

этой основе разработаны требования к естественному и искусственному, совмещенному освещению жилых и общественных зданий. Согласно им в кабинете разработчика SCADA систем норматив естественного освещения при боковом освещении выполняется.

5.3.4 Повышенное значение электромагнитного излучения

Электромагнитные излучения оказывают негативное влияние на сердечно-сосудистую, нервную и эндокринную систему, а также могут привести к раковым заболеваниям. Источниками электромагнитного излучения являются системный блок и кабели, соединяющие электрические цепи.

Временные допустимые уровни ЭМП, создаваемых ПЭВМ указаны в таблице 23.

Таблица 23 – Временные допустимые уровни ЭМП, создаваемых ПЭВМ

Наименование параметров		ВДУ ЭМП
Напряженность электрического поля	в диапазоне частот 5 Гц-2 кГц	25 В/м
	в диапазоне частот 2 кГц-400 кГц	2,5 В/м
Плотность магнитного поля	в диапазоне частот 5 Гц-2 кГц	250 нТл
	в диапазоне частот 2 кГц-400 кГц	25 нТл
Напряженность электростатического поля		500 В

Для снижения воздействия электромагнитного излучения применяют следующие меры:

- расстояние от монитора до работника должно составлять не менее 50 см;
- применение экранных защитных фильтров, а также средств индивидуальной защиты.

5.3.5 Вероятность получения удара электрическим током

Удар электрическим током возможно получить при соприкосновении с токоведущими частями оборудования.

Помещение, в котором расположено рабочее место, относится к категории без повышенной опасности и соответствует следующим условиям:

- напряжение питающей сети 220 В, 50 Гц;
- относительная влажность воздуха 50 %;
- средняя температура около 24 °С;
- наличие непроводящего полового покрытия.

Для снижения вероятности поражения электрическим током ПЭВМ должны быть оборудованы защитным занулением; подача электрического тока в помещение должна осуществляться от отдельного независимого источника питания; необходима изоляция токопроводящих частей и ее непрерывный контроль; должны быть предусмотрены защитное отключение, предупредительная сигнализация и блокировка.

5.4 Обоснование мероприятий по снижению уровней воздействия опасных и вредных факторов на исследователя.

Рабочий процесс находится в помещении, относящееся к категории «без повышенной опасности». Осмотр рабочего места перед началом работы позволит заранее увидеть оголённые провода и снизить риск поражения электрическим током.

Работа оператора ПЭВМ подразумевает напряжённую зрительную работу. Для снижения зрительного напряжения регулярно проводится комплекс упражнений для глаз.

Во время своей работы ПЭВМ нагревается. Для снижения влияния ПЭВМ на микроклимат регулярно проветривается помещение.

5.5 Экологическая безопасность

В ходе эксплуатации программного обеспечения возможно вредное воздействие на литосферу, которое объясняется из-за утилизации электронной техники: системных блоков, мониторов и т.п.

При рассмотрении влияния процесса утилизации персонального компьютера согласно ГОСТ Р 51768-2001 были выявлены особо вредные выбросы. В случае выхода из строя компьютеров, они списываются и отправляются на специальный склад, который при необходимости принимает меры по утилизации списанной техники и комплектующих. В настоящее время в Томской области утилизацией оргтехники занимаются компании: городской полигон и ООО «ЭкоСибирь».

Процесс эксплуатации объекта не подразумевает никаких вредных аспектов, которые влияют на окружающую среду.

5.6 Безопасность в чрезвычайных ситуациях

5.6.1 Анализ вероятных ЧС, которые может инициировать объект исследований

К объекту исследования ограничен доступ посторонних лиц на территорию объекта. Это могут быть любые преграждающие устройства: входные двери, ворота, шлагбаумы, установленные на многих предприятиях или частной собственности людей (квартир, домов, дач). Так же система представлена в виде ПО, анализирующего параметры работы оборудования, и не служит для выдачи команд управления. Поэтому сама система практически исключает возможность инициирования ЧС. Однако не стоит исключать возможность наличия ошибок или недоработок в проектировании систем контроля доступом, что в свою очередь может инициировать возникновение ЧС, связанных с пожароопасностью горючих строительных материалов.

5.6.2 Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований

Использование разработанного программного обеспечения подразумевает работу исключительно с ЭВМ. Наиболее типичной ситуацией чрезвычайной ситуации в данном случае является возникновение пожара. Пожар может возникнуть в результате неисправности электрооборудования.

В соответствии с правилами определения категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности, регламентируемыми правилами устройства электроустановок, помещение, в котором была написана данная работа, относится к категории В, так как в помещении находятся сгораемые вещества и материалы (шкафы, столы, стулья, документация), для питания вычислительной техники используется напряжение 220 В переменного тока.

Для предупреждения возникновения пожара в помещении необходимо проводить следующие пожарно-профилактические мероприятия:

- организационные мероприятия, касающиеся технического процесса с учетом пожарной безопасности объекта;
- эксплуатационные мероприятия, рассматривающие эксплуатацию имеющегося оборудования;
- технические и конструктивные, связанные с правильные размещением и монтажом электрооборудования и отопительных приборов.

Организационные мероприятия:

1. противопожарный инструктаж работников;
2. изучение правил техники безопасности;
3. издание инструкций, плакатов, планов эвакуации.

Эксплуатационные мероприятия:

- Соблюдение эксплуатационных норм оборудования.
- Обеспечение свободного подхода к оборудованию. В комнате рабочие места размещены так, что расстояние между рабочими местами с

видеотерминалами (от поверхности экрана одного, до поверхности экрана другого) составляет порядка 2,5 м, расстояния между боковыми поверхностями порядка 1,5 м, что соответствует нормам. Из вышесказанного следует, что дополнительных мер защиты не требуется.

– Содержание в исправном состоянии изоляции токоведущих проводников.

Технические мероприятия:

1. Соблюдение противопожарных мероприятий при устройстве электропроводок, оборудования, систем отопления, вентиляции и освещения. В коридоре квартиры на достигаемом расстоянии находится рубильник, обесточивающий всю квартиру.

2. Профилактический осмотр, ремонт и испытание оборудования. В коридоре помещения, в котором выполнялась магистерская диссертация, на достигаемом расстоянии, находится пожарный щит. Наиболее дешевым и простым средством пожаротушения является вода, поступающая из обычного водопровода. Для осуществления эффективного тушения огня используют пожарные рукава и стволы, находящиеся в специальных шкафах, расположенных в коридоре. В пунктах первичных средств огнетушения должны располагаться ящик с песком, пожарные ведра и топор. Помещение оборудовано, порошковым огнетушителем ОП-10(б)-АВСЕ МИГ Е [33]. При проектировании здания были запланированы места подъезда пожарных и указаны места расположения пожарных щитов.

В случае возникновения неконтролируемого горения необходимо:

- сообщить диспетчеру о возникновении пожара;
- по возможности обесточить рабочее помещение;
- покинуть помещение согласно плану эвакуации;
- при задымлении необходимо защитить органы дыхания, лечь на пол и как можно раньше покинуть опасную зону.

В соответствии с СП 12.13130.2009 «Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности» [34] помещение разработчика SCADA систем относится к наименее опасной категории (Д) с пониженной пожароопасностью. Само здание по взрывопожарной и пожарной опасности относится к категории (Д). Наружных установок здание не имеет.

Выводы по разделу

В ходе работы по разделу «Социальная ответственность» были рассмотрены правовые и организационные вопросы обеспечения безопасности.

Их суть заключалась в анализе основных нормативных документов, регулирующих и регламентирующих производственную деятельность диспетчера на рассматриваемом объекте.

Был выполнен анализ вредных факторов проектируемого решения, таких как микроклимат, шум, освещение, электромагнитное и электростатическое излучение.

Особое место занимают экологическая безопасность и безопасность в чрезвычайных ситуациях.

На предприятиях, где существует вероятность техносферных катастроф или гибели персонала, создаются целые отделы, занимающиеся техникой безопасности. Инженеры по охране труда и промышленной безопасности следят за выполнением всех норм и регламентов безопасности. За несоблюдение данных регламентов грозят различного рода санкции вплоть до увольнения или уголовной ответственности работников предприятия.

Заключение

В ходе данной выпускной квалификационной работы было разработано программное обеспечение, позволяющее выполнить опрос OPC UA сервера и сохранить полученные значения в сервер базы данных MySQL.

Выбор языка программирования и спецификации OPC был выбран из соображений кроссплатформенности. Python – интерпретируемый язык программирования. Программы, написанные на данном языке программирования, запускаются на любой операционной системе, на которой установлен его интерпретатор. OPC UA сервер также является кроссплатформенным, что позволяет запускать его на любой операционной системе. В том числе и непосредственно на современных промышленных контроллерах. MySQL сервер является бесплатным для некоммерческого использования и также написан под большинство популярных операционных систем.

Разработанное программное обеспечение обладает открытым исходным кодом, что позволяет дорабатывать его. На данное программное обеспечение была подана заявка на получение патента. Также по данной теме было написано 2 научные статьи и подана заявка №2021Э08346 на регистрацию патента программного кода для ЭВМ.

Список используемых источников литературы

1. Новицкая К.В., Цавнин А.В. Применение SCADA-систем для контроля и планирования производства / К.В. Новицкая, А.В. Цавнин / Молодёжь и современные информационные технологии: сб. статей. – Томск, 2019. – С. 221-222
2. TermodatNET [Электронный ресурс] – URL: https://termodat.ru/catalog/po/programma_termodattools/, свободный. (дата обращения: 28.05.2021)
3. Электронный регистратор температуры Термодат 25М/5 – URL: <http://www.termodat.ru/catalog/samopistsy/termodat-25m5/>, свободный (дата обращения: 28.05.2021)
4. Описание Multi-Protocol MasterOPC Server. [Электронный ресурс]. – URL: <https://insat.ru/products/?category=400> (дата обращения: 28.05.2021)
5. Описание стандарта OPC Classic. [Электронный ресурс]. – URL: <https://opcfoundation.org/about/opc-technologies/opc-classic/> (дата обращения: 28.05.2021)
6. Описание стандарта OPC UA. [Электронный ресурс]. – URL: <https://opcfoundation.org/about/opc-technologies/opc-ua/> (дата обращения: 28.05.2021)
7. Описание технологии DCOM [Электронный ресурс] – URL: <https://ru.wikipedia.org/wiki/DCOM> (дата обращения: 28.05.2021)
8. Wind power forecasting of an offshore wind turbine based on high-frequency SCADA data and deep learning neural network [Электронный ресурс] – URL: <https://doi.org/10.1016/j.energy.2020.117693> (дата обращения: 28.05.2021)
9. Model transformation between OPC UA and UML [Электронный ресурс] – URL: <https://doi.org/10.1016/j.csi.2016.09.004> (дата обращения: 28.05.2021)

10. Human Machine Interfaces Based on Open Source Web-Platform and OPC UA [Электронный ресурс] – URL: <https://doi.org/10.1016/j.promfg.2020.02.089> (дата обращения: 28.05.2021)
11. Remote access of ISO 11783 process data by using OPC Unified Architecture technology [Электронный ресурс] – URL: <https://doi.org/10.1016/j.compag.2015.08.002> (дата обращения: 28.05.2021)
12. Analysis of OPC UA performances [Электронный ресурс] – URL: <https://doi.org/10.1016/j.csi.2013.06.004> (дата обращения: 28.05.2021)
13. Описание спецификации OPC UA [Электронный ресурс] – URL: https://www.bookasutp.ru/Chapter9_2_4.aspx (дата обращения: 28.05.2021)
14. Описание устройства адресного пространства OPC UA сервера [Электронный ресурс] – URL: https://documentation.unified-automation.com/uasdkhp/1.4.1/html/_12_ua_address_space_concepts.html (дата обращения: 28.05.2021)
15. Описание типов классов, используемых OPC UA сервером [Электронный ресурс] – URL: https://documentation.unified-automation.com/uasdkhp/1.4.1/html/_12_ua_node_classes.html (дата обращения: 28.05.2021)
16. Язык программирования Python [Электронный ресурс] – URL: <https://ru.wikipedia.org/wiki/Python> (дата обращения: 28.05.2021)
17. Модуль time [Электронный ресурс] – URL: <https://pythonworld.ru/moduli/modul-time.html> (дата обращения: 28.02.2021)
18. Модуль подключения к серверу MySQL [Электронный ресурс] – URL: <https://dev.mysql.com/doc/connector-python/en/> (дата обращения: 28.05.2021)
19. Модуль OPC UA [Электронный ресурс] – URL: <https://python-opcua.readthedocs.io/en/latest/> (дата обращения: 28.05.2021)
20. Модуль создания графического интерфейса tkinter [Электронный ресурс] – URL: <https://docs.python.org/3/library/tkinter.html> (дата обращения: 28.05.2021)

21. Модуль stdiomask [Электронный ресурс] – URL: <https://pypi.org/project/stdiomask/> (дата обращения: 28.05.2021)
22. Модуль matplotlib [Электронный ресурс] – URL: <https://matplotlib.org/stable/users/index.html> (дата обращения: 28.05.2021)
23. Клиент OPC UA сервера UaExpert [Электронный ресурс] – URL: <https://www.unified-automation.com/products/development-tools/uaexpert.html> (дата обращения: 28.05.2021)
24. Программируемый логический контроллер Овен ПЛК200 [Электронный ресурс] – URL: <https://owen.ru/product/plk200> (дата обращения: 28.05.2021)
25. Трудовой кодекс Российской Федерации" от 30.12.2001 N 197-ФЗ (ред. от 30.04.2021)
26. ГОСТ 12.2.032-78. Рабочее место при выполнении работ сидя.
27. ГОСТ 12.1.003-83. ССБТ. Общие требования безопасности.
28. СНиП 23-03-2003. Защита от шума.
29. СанПиН 2.2.4.548-96. Гигиенические требования к микроклимату производственных помещений.
30. СП 52.13330.2016. Естественное и искусственное освещение.
31. ГОСТ Р 50923-96 Дисплей. Рабочее место оператора
32. ГОСТ 12.1.019-2017 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты.
33. Огнетушитель ОП-10(б)-АВСЕ МИГ Е [Электронный ресурс] – URL: <https://www.tinko.ru/catalog/product/266171/> (дата обращения: 28.05.2021)
34. СП 12.13130.2009 Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности (с Изменением N 1)

Приложение А

(Обязательное)

Функция подключения к базе данных

```
def connect_db(self):
    try:
        self.cnx =
mysql.connector.connect(user=input("user: "),
password=stdiomask.getpass(prompt="Password: "),

host=input("host: "),

auth_plugin='mysql_native_password')
        print("Connected")
    except mysql.connector.Error as err:
        if err.errno ==
errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your
user name or password")
            self.connect_db()
        else:
            print(err)
            self.connect_db()
```

Приложение Б

(Обязательное)

Функция подключения к OPC UA серверу

```
def connect_opc(self):
    url = "opc.tcp://" + input("opc.tcp://")
    self.client = Client(url)
    try:
        self.client.connect()
    except:
        print('Connection error. Please, try
again')
        self.connect_opc()
    else:
        print("Client connected")
```

Приложение В

(Обязательное)

Программу запуска графического интерфейса

```
import tkinter as tk
import tkinter.ttk as ttk
from opcua import Client
from mysql.connector import errorcode
import mysql.connector
import stdiomask
from tkinter import messagebox
import re
class App(tk.Tk):
    def __init__(self):
        super().__init__()
        self.connect_opc()
        self.connect_db()
        self.abstractmethod = self.client.get_root_node()
        cursor = self.cnx.cursor()
        databases = "show databases"
        cursor.execute(databases)
        count = 0
        databases = list(cursor)
        for i in databases:
            print(count, i[0])
            count += 1
        database = databases[int(input("Select
database: "))][0]
        cursor.execute(f"USE {database}")
if __name__ == "__main__":
    root = App()
    mainmenu = Menu(root)
    root.config(menu=mainmenu)
    filemenu = Menu(mainmenu, tearoff=0)
    filemenu.add_command(label="Построить график")
    filemenu.add_command(label="Выход",
command=root.on_closing)
    helpmenu = Menu(mainmenu, tearoff=0)
    helpmenu.add_command(label="Помощь")
    helpmenu.add_command(label="О программе")
    mainmenu.add_cascade(label="Меню", menu=filemenu)
    mainmenu.add_cascade(label="Справка",
menu=helpmenu)
```

```

root.title("OPC UA Client")
root.nodes = {}
heads = ['Tag', 'NodeID', 'Type']
root.tree = ttk.Treeview(root)
root.tree['columns'] = [0, 1, 2]
root.tree.heading("#0", text='Tree', anchor=tk.W)
root.tree.heading("#1", text=heads[0], anchor=tk.W)
root.tree.heading("#2", text=heads[1], anchor=tk.W)
root.tree.heading("#3", text=heads[2], anchor=tk.W)
ysb = ttk.Scrollbar(root, orient=tk.VERTICAL,
command=root.tree.yview)
xsb = ttk.Scrollbar(root, orient=tk.HORIZONTAL,
command=root.tree.xview)
root.tree.configure(yscroll=ysb.set,
xscroll=xsb.set)
root.tree.grid(row=0, column=0, sticky=tk.N + tk.S
+ tk.E + tk.W)
ysb.grid(row=0, column=1, sticky=tk.N + tk.S)
xsb.grid(row=1, column=0, sticky=tk.E + tk.W)
root.rowconfigure(0, weight=1)
root.columnconfigure(0, weight=1)

root.tree.bind("<<TreeviewOpen>>", root.open_node)
root.protocol("WM_DELETE_WINDOW", root.on_closing)
root_node = root.tree.insert(parent='',
index=tk.END, text='Root', open=False)
root.populate_node(root_node, root.abspath)
root.mainloop()

```

Приложение Г

(Обязательное)

Листинг используемых в проекте функций

```
def populate_node(self, parent, abspath):
#Наполнение раскрываемого элемента списка
    children = abspath.get_children()
    for entry in children:
        node_name = str(entry.get_display_name())
        m =
re.match(r'LocalizedText\(Encoding:(.*), Locale:(.*),
Text:(.*)\) ', node_name)
        try:
            node_type =
entry.get_data_type_as_variant_type()
        except:
            node_type = ''
            node_type = str(node_type).split(sep='.')[-
1]
        node = self.tree.insert(parent, tk.END,
text=str(f'{m.group(3)} {node_type}'), open=False)
        if children:
            self.nodes[node] = entry
            self.tree.insert(node, tk.END)

# Действие на открытие списка. Сначала выбирается
элемент списка, после чего происходит его наполнение
def open_node(self):
    item = self.tree.focus()
    print(item)
    abspath = self.nodes.pop(item, False)
    print(abspath)
    if abspath:
        children = self.tree.get_children(item)
        self.tree.delete(children)
        self.populate_node(item, abspath)
```

Приложение Д

(Обязательное)

Таблица тэгов

Название переменной	Тип данных	Начальное значение
y_pid	REAL	
LL	REAL	
set_point	REAL	15
kp	REAL	75
tn	REAL	2
tv	REAL	0,5
y_manual	REAL	15
y_offset	REAL	4
y_min	REAL	-20
y_max	REAL	20
user	BOOL	
reset	BOOL	
period	TIME	t#2s
amax	INT	20
close	BOOL	
open	BOOL	
a	REAL	
ss	REAL	1
TM	DWORD	10
F1	REAL	10
SHUUM	GEN	
a	REAL	
INPOT	REAL	
SHUM	REAL	
SPASE	REAL	
TMM	DWORD	
RESET	BOOL	
Y_PID	REAL	
PERIOD	TIME	
AMPLITUDE	INT	
OPEN	BOOL	
CLOSE	BOOL	
F1, F2, F	REAL	
INTEGRAL	INTEGRAL	
integ	REAL	

Приложение Е
(Справочное)

OPC UA server client development

Студент:

Группа	ФИО	Подпись	Дата
8ТМ91	Арбузов Роман Александрович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР	Курганов В.В.	К.Т.Н.		

Консультант

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ОАР	Цавнин А.В.	-		

Консультант – лингвист ШБИП ОИЯ

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИЯ	Сидоренко Т.В.	К.П.Н		

1 Current situation

1.1 Introduction

The most important, modern and widespread specification that defines the data transferring between hardware and an operator workstation is the unified OPC architecture. This specification is most often used to transfer technological data from the level of a programmable logic controller to the upper level of an automated process control system and, separately, to SCADA systems [1]. In particular, one of the most popular OPC specifications is OPC UA.

The object of study is a client capable of polling the OPC UA server and recording the received data to the MySQL database. This paper is useful for conducting research with small automated systems or during the debugging a technological process.

1.2 Pre-project analysis

It became necessary to screen temperature sensors and save the history of their changes during of the development of an automated control system for the thermochemical rubber processing unit. The measurement data were output to a PC using the TermodatNET program [2], which interrogated the Termodat 25M / 5 recorder [3]. Figure 1 shows the interface of the TermodatNET program.

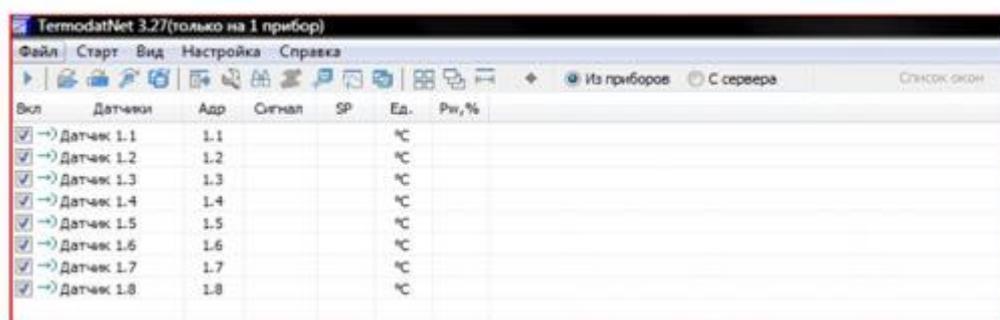


Figure 1 – TermodatNET interface

TermodatNET also includes function of archiving data, exporting saved data to the MSSQL database and displaying a trend line on the screen.

The functions of this program were sufficient for the task. The main limitation of this program is to work only with the equipment of the Termodat

Company. In addition, this program is paid, and has a limit on the number of devices connected at the same time.

The measuring unit is connected using its own interface to the Termodat 25M/5 recorder. This recorder has the following properties:

- Archiving data to the internal memory of the recorder.
- Setting the temperature setpoint.
- Configure warnings and alarms.
- Converting a digital signal to a temperature value depending on the temperature sensor selected type.
- Display of the current temperature reading for each measured channel on the screen of the recorder.
- Display of temperature graphs for each temperature sensor directly on the recorder screen.
- Data transmission of the recorder using the RS-485 interface using the Modbus-RTU, Modbus-ASCII protocols and the proprietary Termodat protocol.

An RS-485 - USB converter is used to connect to a PC.

This recorder is shown in Figure 2.

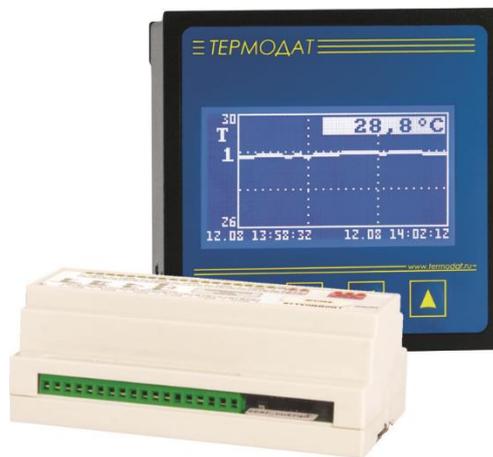


Figure 2 – Termodat-25M

As a test OPC server, we chose the InSAT software “Multi-Protocol MasterOPC Server” [4]. This choice was made based on the following considerations:

- It supports Modbus RTU / ASCII / TCP protocols.

- Free version is unlimited in time, but limited in the number of tags (32 tags).
- User-friendly interface. Work in OPC DA (Data Access) and OPC UA (Unified Architecture) server mode.

It is planned to use other sensors in the future, which the TermodatNET program does not allow to poll.

This problem can be solved by using a SCADA system. However, the installation of a SCADA system and its mastering takes a lot of time and its capabilities are redundant. In addition to this, most SCADA systems run exclusively under Windows OS.

Due to the fact that most SCADA systems receive data from OPC servers, it was decided to develop an OPC client similar in functionality to the TermodatNET program. This client should exclude the drawback of working exclusively with the equipment of the Termodat Company and should not be difficult to master.

Based on the received initial data, the following tasks were set, which are necessary for the development of an OPC client:

1. Study the OPC specifications.
2. Conduct an analysis of scientific articles that describe the purpose of the OPC server in their work.
3. Select the specification of the OPC server.
4. Select the development language for the OPC client.
5. Select the DBMS with which the data will be written to the database.
6. Implement the OPC UA server client in the selected programming language.

1.3 Existing specifications of OPC servers review

Today, there are two main specifications for Open Platform Communication (OPC) technology - Classic [5], which includes OPC Data Access

(OPC DA), OPC Historical Data Access (OPC HDA), OPC Alarms & Events (OPC AE) and OPC Unified Architecture (OPC UA) [6].

OPC Classic uses DCOM technology as a basis [7].

DCOM (Distributed Component Object Model) is an extension of Microsoft's COM. This extension is focused on supporting and integrating distributed object applications operating in the network.

A key aspect of COM is that it enables communication between clients and servers through interfaces. It is the interface that provides the client with a way to get ahead of the server technologies supported at runtime. To expand the capabilities of the server, you just need to add a new interface to the existing ones.

Although OPC Classic has been a huge success, this specification is outdated and has the following disadvantages:

- this technology is available only on operating systems of the Windows family;
- communication with DCOM technology. The source codes of this technology are closed, which does not allow solving software reliability issues. It also does not allow identifying and eliminating emerging software failures;
- the emergence of configuration difficulties that are associated with DCOM;
- inaccurate DCOM communications interruption messages;
- the inability of DCOM to exchange data over the Internet;
- the inability of DCOM to ensure information security.

In connection with the above shortcomings, the OPC Foundation consortium decided to develop a new specification, devoid of them. The new specification received a unified architecture and was named OPC UA. Its main advantages:

- implementation in the ANSI C programming language for portability to other platforms, including embedded systems. NET and Java versions developed by the OPC Foundation are also available;

- This specification is service oriented, not object oriented. This allows the OPC UA specification to be used on any device using web services;
- allows scaling of OPC UA, i.e. changing the size of the program depending on the computing resources of the processor and the required functionality. Compilation can also be performed as a single-threaded or multi-threaded application;
- support for a reliable and modern transport mechanism SOAP (Simple Object Access Protocol) based on XML (eXtensible Markup Language) using the HTTP protocol;
- ensuring a high degree of information security;
- configurable timeout for each service;
- using the open standards of the World Wide Web Consortium (W3C) instead of the closed standard COM / DCOM.

It should be noted that modern controllers have an operating system based on the Linux kernel, which allows them to run an OPC UA server on them.

In addition, a number of manufacturers offer controllers on which the OPC UA server is already installed. For example, a controller manufactured by Aries PLC200, Siemens S7-1500 and others. The OPC UA server is configured during project development, after which it can be interrogated using the developed client.

The developed software is created on the assumption that, like the OPC UA server, it can run on any modern operating system.

1.4 Literature review

This literature review looked at publications and articles that use OPC UA-based solutions. Only foreign articles that were published no later than 8 years ago were selected as sources of information.

Considering the experience of using OPC UA in foreign countries, it was concluded that this OPC specification is versatile. For example, a 2020 article describes how to build a deep learning neural network based on a database

obtained using data collected by a SCADA system based on an OPC server. Since forecasting in wind power [8] is a complex task and depends on many parameters, the forecast model is based on data from a SCADA system with a measurement frequency of instantaneous values of 1 s.

Also considering OPC UA, it is worth noting the role of this specification in the digitalization of production and Industry 4.0 itself [9], in which the trend of information transportation using the Internet is clearly expressed. Also, the OPC UA specification is the main tool for data exchange in the Internet of Things.

In addition to this, the OPC UA specification has found its use as a human-machine interface based on a web platform [10]. The architecture of the developed system consists of an OPC UA client, which allows a web application to interact with an OPC UA server. The OPC UA server is responsible for communicating with the controller. The platform's web interface allows you to create various versions of human-machine interfaces for monitoring different processes.

The advantages of the OPC UA specification as a means of remote access [11] are also described in the article on the automation of agricultural machinery. The advantage of using OPC UA is that remote access data is presented in a structured and consistent way, not just a list of raw variables. The results of the reviewed article empirically demonstrate that the overhead of using this specification is relatively small. In turn, low overhead is critical to keeping the cost of data transmission low, moreover, low latency allows support for various real-time telemetry applications.

In addition, articles were considered on the characteristics and capabilities of the OPC UA specification [12]. The key information obtained in the review of articles in this direction is the compliance of the specification under consideration with most of the requests that are presented to it in accordance with the requirements of Industry 4.0. Moreover, the characteristics of this specification are studied for various modes and settings of the OPC server, which allows you to make a choice in the direction of either security or performance of the developed system based on OPC UA.

The release dates of the articles emphasize the relevance of the topic under consideration and show that OPS servers are used in various projects and systems. Especially OPC servers have gained popularity as a tool for building databases for deep learning machine learning, where it is necessary to provide a large amount of information with a low frequency of parameter measurement.

Based on all of the above, it was decided to use the unified architecture of OPC UA as a basis.

2 Description of the OPC UA specification

2.1 Description of the principle of operation of the OPC UA server

An OPC UA-based system can accommodate several servers and clients at the same time. Each client is given the opportunity to work simultaneously with several servers. The server is also given the ability to respond to requests from several clients simultaneously [13]. Figure 3 shows the structure of a client program in OPC UA.

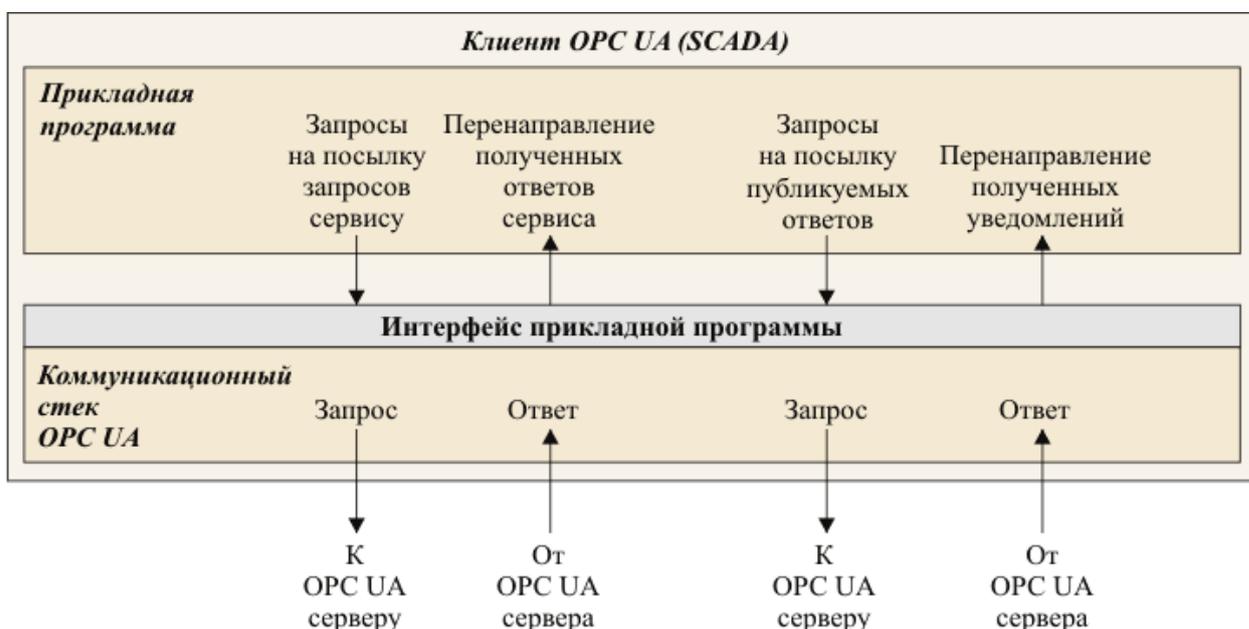


Figure 3 – Architecture of the client program in OPC UA structure

The client of the OPC UA server is a program that polls it. For example, a SCADA system. Sending requests to the OPC server is performed using the internal interface, which is an isolating layer between the client program and the switching stack. The switch stack serves the role of translating client requests into

messages to invoke the required service to be sent to the server. After receiving a response to the sent requests, the switching stack passes the polling result back to the client program.

Also, the OPC UA architecture allows data exchange between two servers. To do this, one of the servers becomes a client, and the other acts as a server. This allows multiple servers to be chained together. In this case, each of them will act on one side of the chain as a client, on the other side - as a server. Figure 4 shows the architecture of the OPC UA server.

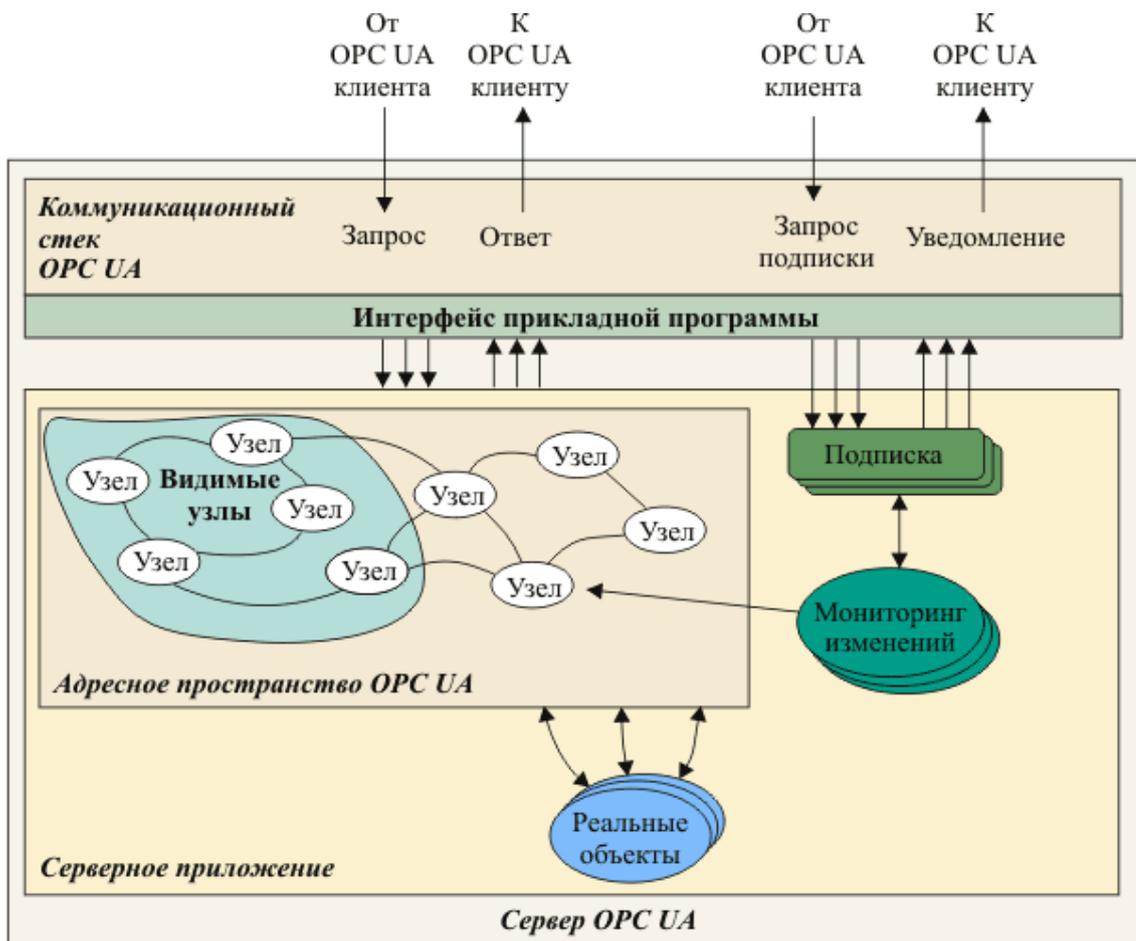


Figure 4 – OPC UA server architecture

The real objects in this figure are PLCs, I/O devices and others capable of transmitting data using the OPC server.

A server application is a software implementation of the functions that the server must perform. The interaction of the OPC UA server with the client is performed through the application program interface, by sending requests and receiving responses.

Data exchange between server and client can be done in two ways:

- answering instant requests;
- according to the "publisher-subscriber" scheme.

When exchanging data according to the "publisher-subscriber" scheme, the client program "subscribes" to receive certain data. The server will have to provide as soon as they become available, i.e. to change these values. To implement the subscription mode, the server monitors nodes and their corresponding real objects in order to detect changes. When changes in data, events or alarms are detected, the server generates a notification that is sent to the client via the subscription channel.

2.2 Structure of the OPC UA address space description

The main purpose of the OPC UA address space [14] is to standardize the presentation of server objects to clients who make a request to it. To achieve this goal, the OPC UA object model was developed. This model defines server objects as variables and their methods. It also describes how they interact.

In addition, objects can be typed. In other words, the OPC UA specification provides the ability to define and specify both the types of objects (classes and variables and their methods) and their objects.

However, the use of this object-oriented approach is optional. As with OPC Data Access, a simple address space model can be implemented using objects, folders, and variables. But the presence of additional features inherent in the object-oriented approach greatly simplifies the implementation of object-oriented systems using the OPC UA specification. Figure 5 shows the object model of the OPC UA server.

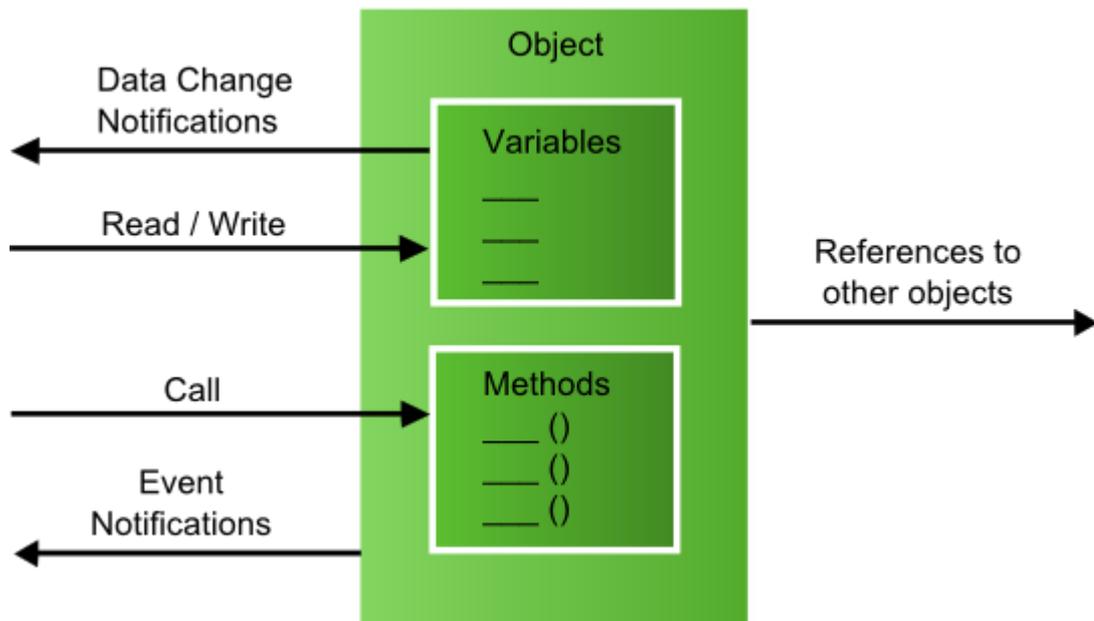


Figure 5 – OPC UA server's object model

OPC UA Object Model

The UA services are used to access objects and their components. For example, they are used to read or write the value of a variable, call a method on an object, or receive events from it (for example, a value change event). Browse services can be used to analyze the relationship between objects and their components. These services act as a client of the OPC UA server.

The elements of this model in the address space are nodes. Each node is assigned a class, such as an object, variable, or method of an object, and represents another element of the object model.

Node model

Objects and related information provided by the OPC UA server to clients are called its address space.

Objects and their components are represented in the address space as a collection of nodes, described using attributes and related links. The node model is shown in Figure 6.

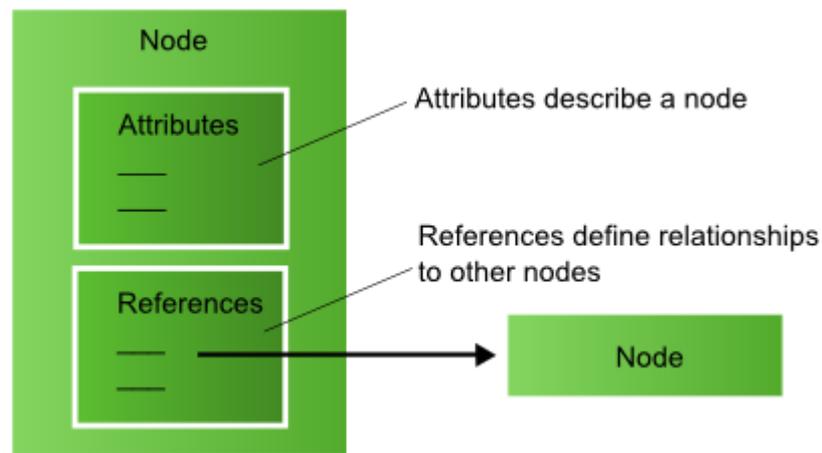


Figure 6 – Node model

Node classes

For different nodes, attributes and links are defined using node classes. The OPC UA specification defines eight different node classes [15]:

- Object. This class is used to display systems, their components, real objects and software.
- Variable. This class is used to display the content of objects.
- Method. This class is required to display various methods in the address space.
- Class "Display" is designed to indicate the visibility of nodes and the relationship between them in the address space.
- The Object Type class describes the types of object nodes in the server's address space.
- Variable type. Describes the type of the variable stored in the address space.
- Reference type. This class describes the types of relationships used by the server.
- Data type. This type represents the abstractness of the data. All data types are represented as nodes of node classes in the address space. Has one attribute - the values are abstract or not.

Each node in the address space is an instance of one of the node classes. Clients and servers are not allowed to define additional node classes or extend the definitions of these node classes (for example, a list of attributes for a node class).

Each of the classes has its own attributes. Attributes are data elements that describe nodes. Clients can access attribute values using reader, writer, query, and change / track subscription.

Attributes are the elementary components of node classes. Each attribute definition consists of its identifier, name, description, data type, and an indicator indicating whether the attribute is required. The attribute definitions are not visible in the address space because they are part of the node class definition and are known to the client. The set of attributes defined for each node class is immutable and cannot be extended. When creating a node in the address space, it must specify the values of the required class attributes of this node.

Attributes provide OPC UA metadata available to all nodes. Attributes are common to all nodes of a node class and are defined only by the OPC UA specification.

References are used to link nodes together. The References are accessed using the browse and query services.

Like attributes, links are the main component of nodes. Unlike attributes, links are defined as instances of nodes of the "reference type" class. Nodes of the "reference type" class are visible in the address space and are defined using the "reference type" class.

The node containing the reference is called the source node. The node that is referenced is referred to as the target node. The combination of source node, reference type, and target node is used in OPC UA services to uniquely identify links. Thus, each node can reference another node with the same reference type only once.

The target node of the link can be located both in the address space of the source node, and in the address space of another OPC UA server. Target nodes located on other servers are identified to OPC UA services using a combination of

the name of the remote server and the ID assigned to the node by the remote server.

Variables are used to display values. There are two types of variables: properties and data variables. They differ in the type of data they represent and the ability to contain other variables.

Properties contain server-defined metadata for objects, their variables, and other nodes. Properties differ from attributes by the ability to add them yourself. Properties characterize what a node is. For example, the device or the order in which devices are polled.

For example, an attribute defines the data type of a variable, while a property can be used to specify the units of measure for a given variable.

In order to avoid recursion, properties are prohibited from using the properties that are defined for them. To easily identify properties, the property name must be unique in the context of the node containing those properties.

The node and properties of this node must be on the same server.

Data variables define the content of an object. An object is a container for variables and methods. An object node does not contain a value, unlike variable nodes. Hence, variable data nodes are used to represent object data.

For example, a file uses variable data to display the contents of the file as an array of bytes. Its properties can be used to indicate when it was created and who created it.