

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

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Развитие графических возможностей программы MasterSCADA

УДК 004.42:004.92

Студент

Группа	ФИО	Подпись	Дата
8Т5Б	Алеутдинов Кирилл Александрович		

Руководитель ВКР

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

Нормоконтроль

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

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

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

По разделу «Социальная ответственность»

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

ДОПУСТИТЬ К ЗАЩИТЕ:

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

Томск – 2019 г.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

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

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

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

Уровень образования – Бакалавриат

Отделение (НОЦ) – Отделение автоматизации и робототехники

Период выполнения – осенний / весенний семестр 2018/2019 учебного года

Форма представления работы:

бакалаврская работа

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	
--	--

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

СОСТАВИЛ:

Руководитель ВКР

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

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Громаков Евгений Иванович	к.т.н., доцент		

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

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

УТВЕРЖДАЮ:
Руководитель ООП

(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ на выполнение выпускной квалификационной работы

В форме:

бакалаврской работы

Студенту:

Группа	ФИО
8Т5Б	Алеутдинову Кириллу Александровичу

Тема работы:

Развитие графических возможностей программы MasterSCADA	
Утверждена приказом директора (дата, номер)	№1095/ от 12.02.2019

Срок сдачи студентом выполненной работы:	13.06.2019
--	------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

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>	Цель работы: расширение графических возможностей программы MasterSCADA путём разработки и интеграции в программу библиотеки функциональных блоков. Объектом исследования является SCADA-система. В состав системы входят: контроллер, персональный компьютер, дисплей, соединительные провода.
---	--

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, подлежащих разработке; заключение по работе).</i>	1. Исследование целей и задач, решаемых при разработке SCADA-систем. 2. Определение и выбор основных параметров и характеристик для сравнения SCADA-систем. 3. Выбор средств и методов разработки. 4. Изучение методологии создания и реализация функционального блока для MasterSCADA на платформе Microsoft .Net Framework.
Перечень графического материала <i>(с точным указанием обязательных чертежей)</i>	1. Базовая схема SCADA-системы. 2. Таблица описывающая необходимые для отображения и использования свойства функционального блока. 3. Листинг кода функционального блока.
Консультанты по разделам выпускной квалификационной работы <i>(с указанием разделов)</i>	
Раздел Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Консультант Меньшикова Екатерина Валентиновна
Социальная ответственность	Винокурова Галина Федоровна
Названия разделов, которые должны быть написаны на русском и иностранном языках:	
-	

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	22.01.2019
--	-------------------

Задание выдал руководитель / консультант (при наличии):

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

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Т5Б	Алеутдинов Кирилл Александрович		

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

Студенту:

Группа	ФИО
8Т5Б	Алеутдинову Кириллу Александровичу

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

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

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Бюджет – 225 665,51 руб. Затраты на заработную плату – 79 054,38 руб., на специальное оборудование – 139 008 руб.
2. Нормы и нормативы расходования ресурсов	Тариф на электроэнергию – 5,8 кВт/ч Норма амортизации – 20%
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Налог во внебюджетные фонды 27,1 Районный коэффициент – 1,3 Накладные расходы – 16%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Оценка коммерческого потенциала инженерных решений (ИР)	Оценка потенциальных потребителей результатов исследования, анализ конкурентных решений, SWOT-анализ
2. Формирование календарного плана и бюджета инженерного проекта (ИП)	Планирование этапов работ, определение трудоемкости и построение календарного графика, формирование бюджета
3. Оценка ресурсной, финансовой, социальной, бюджетной эффективности ИР и потенциальных рисков	Оценка сравнительной эффективности исследования. Интегральный показатель ресурсоэффективности – 5 Интегральный показатель эффективности – 12,048 Сравнительная эффективность проекта – 2,563

Перечень графического материала (с точным указанием обязательных чертежей)

1. Оценка конкурентоспособности ИР
2. Матрица SWOT
3. График разработки ИР
4. Материальные затраты
5. Инвестиционный план. Бюджет ИП
6. Основные показатели эффективности ИП

Дата выдачи задания для раздела по линейному графику	12.03.2019
--	------------

Задание выдал консультант:

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

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Т5Б	Алеутдинов Кирилл Александрович		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8Т5Б	Алеутдинову Кириллу Александровичу

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

Тема ВКР:

Развитие графических возможностей программы MasterSCADA	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объектом исследования является SCADA-система. Рабочая зона оператора SCADA-системы располагается в специально оборудованном помещении, где работник занимается непосредственно своими обязанностями. Область применения объекта - любое производство, использующее автоматизированную систему управления технологическим процессом.
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.	Определены специальные правовые нормы в соответствии с СанПИН 2.2.2/2.4.1340-03, регламентирующие деятельность работника с использованием ПЭВМ. На основе СанПиН 2.2.542-96 определены основные требования к организации рабочего места.
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	Проанализировать выявленные вредные факторы при разработке и эксплуатации проектируемого решения: – опасности, связанные с использованием локальных вычислительных сетей; – безопасность элементов системы управления; – электробезопасность.
3. Экологическая безопасность:	Рассмотрены средства обеспечения экологической безопасности на производствах, использующих автоматизированные системы управления технологическим процессом.
4. Безопасность в чрезвычайных ситуациях:	Безопасность в чрезвычайных ситуациях: – Возникновение пожара.

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

Задание выдал консультант:

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

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Т5Б	Алеутдинов Кирилл Александрович		

РЕФЕРАТ

Выпускная квалификационная работа содержит 73 страницы, 12 рисунков, 17 таблиц, 22 источника.

Ключевые слова: Supervisory Control and Data Acquisition, MasterSCADA, функциональный блок, control, Component Object Model, .NET Framework.

Объектом исследования является SCADA-система.

Цель работы – расширение графических возможностей программы MasterSCADA путём разработки и интеграции в программу библиотеки функциональных блоков.

В данном проекте была разработана библиотеки функциональных блоков для программного пакета MasterSCADA.

Разработанная библиотека может применяться в системах контроля, управления и сбора данных на различных промышленных предприятиях.

Экономическая эффективность и значимость работы, себестоимость исследуемой системы представлена в главе финансовый менеджмент.

Выявлены основные опасные и вредные факторы, которые могут возникнуть в процессе работы с системой, в которой задействована разрабатываемая библиотека, разработаны меры по снижению воздействий системы на экологию, что представлено в главе социальная ответственность.

СОДЕРЖАНИЕ

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ.....	11
ВВЕДЕНИЕ.....	13
1 ОБЗОР ЛИТЕРАТУРЫ	15
1.1 Общие сведения о SCADA-системах	15
1.2 Цели и задачи, решаемые при разработке SCADA-систем.....	16
1.3 Основные параметры и характеристики для сравнения SCADA-систем.....	18
1.4 Оценка нескольких SCADA-систем	19
1.5 Выводы по обзору литературы	21
2 СРЕДСТВА И МЕТОДЫ РАЗРАБОТКИ.....	22
2.1 MasterSCADA.....	22
2.1.1 Функциональные блоки в MasterSCADA	22
2.2 Технологический стандарт Component Object Model	22
2.3 Программная платформа .NET Framework.....	24
2.4 Среда разработки программного обеспечения Visual Studio.....	24
2.5 Объектно-ориентированный язык программирования C#.....	25
3 РАЗРАБОТКА БИБЛИОТЕКИ ДЛЯ MASTERSCADA	27
3.1 Создание и настройка файла проекта.....	27
3.2 Выбор набора компонентов и их свойств для реализации в библиотеке	29
3.3 Описание компонентов на языке C#	30
4 РЕЗУЛЬТАТЫ ПРОВЕДЁННОЙ РАБОТЫ.....	34
4.1 Оценка результатов работы.....	34
4.2 Преимущества разработанных компонентов над предустановленными.....	34
4.3 Возможность использования разработанной библиотеки для других SCADA-систем... ..	36
5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ.....	38
5.1 Потенциальные потребители результатов исследования.....	38
5.2 Анализ конкурентных технических решений	39
5.3 SWOT-анализ	41
5.4 Планирование научно-исследовательских работ.....	43
5.5 Бюджет научно-технического исследования (НТИ).....	49
5.6 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.....	54
6 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ	57

ЗАКЛЮЧЕНИЕ	65
CONCLUSION	66
СПИСОК ЛИТЕРАТУРЫ.....	67
ПРИЛОЖЕНИЕ А Листинг кода функционального блока	69

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В данной работе применены следующие термины с соответствующими определениями:

автоматизированная система (АС): комплекс аппаратных и программных средств, предназначенный для управления различными процессами в рамках технологического процесса.

SCADA (Supervisory Control And Data Acquisition), СКАДА: архитектура системы управления, которая использует компьютеры, сетевую передачу данных и графические пользовательские интерфейсы для высокоуровневого управления процессами;

технологический процесс (ТП): последовательность технологических операций, необходимых для выполнения определенного вида работ.

Мнемосхема: представление технологической схемы в упрощенном виде на экране АРМ.

ERP (Enterprise Resource Planning): организационная стратегия интеграции производства и операций, управления трудовыми ресурсами, финансового менеджмента и управления активами.

ODBC (Open Database Connectivity): программный интерфейс (API) доступа к базам данных, разработанный компанией Microsoft.

SQL (Structured Query Language): декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных.

FCL (Framework Class Library): библиотека классов программной платформы .NET Framework.

CLR (Common Language Runtime): виртуальная машина приложения, которая обеспечивает такие функции, как безопасность, управление памятью и обработка исключений.

IDE (Integrated Development Environment): система программных средств, используемая программистами для разработки программного

обеспечения.

HRM (human resources management, HRM, HR-менеджмент): область знаний и практической деятельности, направленная на обеспечение организации качественным персоналом, способным выполнять возложенные на него трудовые функции, и оптимальное его использование.

В данной работе применены следующие обозначения и сокращения со соответствующей расшифровкой:

АРМ – Автоматизированное рабочее место;

PLC (Programmable Logic Controllers) – Программируемые логические контроллеры (ПЛК);

RTU (Remote Terminal Unit) – Программируемые логические контроллеры (ПЛК);

HMI (Human Machine Interface), ЧМИ – Человеко-машинный интерфейс;

OPC (Object Protocol Control) – протокол для управления процессами;

ФБ – Функциональный блок.

ВФБ – Визуальный функциональный блок.

MES (Manufacturing Execution System) – система управления производственными процессами.

EAM (Enterprise Asset Management System) — система управления основными фондами.

ПЭВМ – Персональная электронно-вычислительная машина.

ПК – Персональный компьютер.

ЛВС – Локальные вычислительные сети.

ЧС - Чрезвычайная ситуация.

ВВЕДЕНИЕ

Сейчас трудно представить реализацию масштабного технологического процесса без внедрения автоматизированной системы управления. Её внедрение позволяет обеспечить более эффективный контроль качества продукции и оптимизировать экономические затраты. Для того, чтобы разработать автоматизированную систему управления технологическим процессом нанимается штат высококвалифицированных сотрудников, специализирующихся на проектировании и наладке таких систем. Кроме того, в штате должны быть специалисты, обладающие навыками программирования и визуализации технологического процесса с помощью SCADA-систем.

Современные автоматизированные системы управления технологическим процессом являются многоуровневыми. На верхнем уровне находятся системы диспетчерского управления и сбора данных, также известные как SCADA-системы.

SCADA – это архитектура системы управления, которая использует компьютеры, сетевую передачу данных и графические пользовательские интерфейсы для высокоуровневого управления процессами, также для взаимодействия с технологической установкой используются другие периферийные устройства, такие как программируемый логический контроллер (ПЛК) и дискретные ПИД-контроллеры.

В настоящее время на рынке представлено множество различных программных продуктов, на базе которых может быть реализована SCADA-система. У каждого из них есть свои достоинства и недостатки и выбор системы в каждом конкретном случае становится неоднозначным решением. Поэтому сейчас как никогда важно разобраться в ключевых критериях для оценки качества SCADA-систем таких как: функциональные и графические возможности, технические и эксплуатационные характеристики, открытость системы и ценовая политика производящей компании.

Интересным моментом в выборе SCADA-системы является возможность самостоятельной модификации программной среды при помощи экспорта

компонентов, разработанных в других средах. Это позволяет существенно расширить возможности программы без значительной переплаты за программный продукт. Самостоятельная доработка программной среды – это неочевидное и зачастую наиболее выгодное решение.

Целью данной работы является развитие графических возможностей программы MasterSCADA путём разработки и интеграции в программу собственной библиотеки компонентов.

1 ОБЗОР ЛИТЕРАТУРЫ

1.1 Общие сведения о SCADA-системах

SCADA (от англ. Supervisory Control And Data Acquisition) система— это система программных и аппаратных элементов, которая позволяет промышленным организациям:

- Локально или удаленно управлять расположенными производственными процессами;
- Осуществлять мониторинг, сбор и обработку данных в реальном времени;
- Настраивать непосредственное взаимодействие с такими устройствами, как датчики, клапаны, насосы, двигатели и многими другими, через программное обеспечение человеко-машинного интерфейса (ЧМИ);
- Записывать и архивировать событий в файл журнала (log) [1].

SCADA-системы имеют решающее значение для промышленных организаций, поскольку они помогают увеличить эффективность производства, быстро обрабатывать данные для принятия более обоснованных решений и способны оперативно сообщать о системных проблемах и ошибках, помогая уменьшить время простоя.

Базовая архитектура SCADA начинается с программируемых логических контроллеров (ПЛК) или удаленных терминалов (RTU). ПЛК и RTU - это микрокомпьютеры, которые взаимодействуют с массивом объектов, таких как фабричные машины, датчики и исполнительные устройства, а затем направляют информацию от этих объектов на компьютеры с программным обеспечением SCADA (см. рисунок 1). Программное обеспечение SCADA обрабатывает, распределяет и отображает данные, помогая операторам и другим сотрудникам анализировать данные и принимать решения [1].

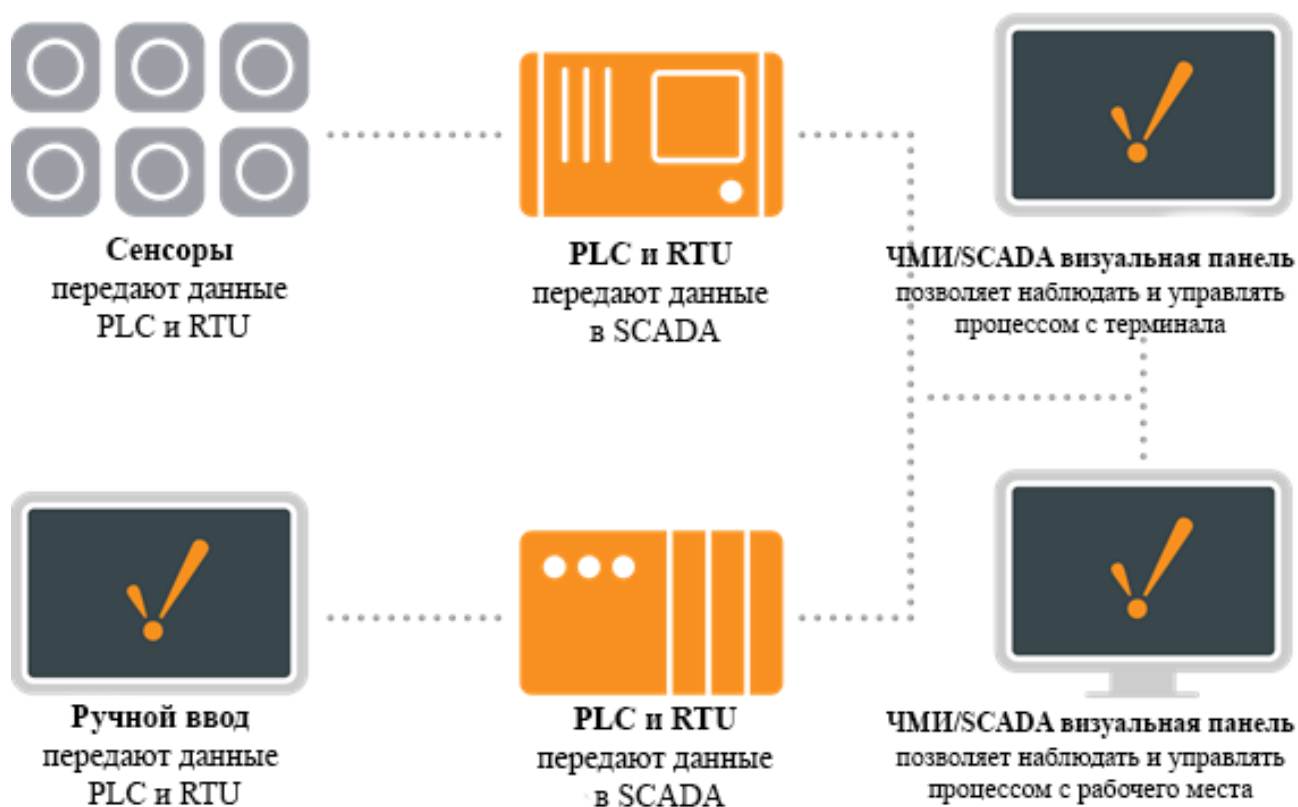


Рисунок 1 – Базовая схема SCADA-системы

1.2 Цели и задачи, решаемые при разработке SCADA-систем

Стивен Гарбрехт, главный цифровой архитектор компании AchieveDE, выделил 10 основных целей и задач, решаемых при помощи внедрения SCADA-систем:

1. **Вычислительные операции.** Обеспечение координации и видимости всех параметров производства. Возможность написания сложных сценариев работы оборудования, обнаружения аварий и создания событий при помощи SCADA обеспечивают эффективную среду управления.

2. **Прослеживаемость.** Связывание сырья с готовыми партиями продукта и проблемами с качеством. Такие возможности необходимы в некоторых отраслях. Например, при производстве продуктов питания и фармацевтики.

3. **Автоматизированное составление отчетностей и ведение журнала.** Для фиксации всего материала, который вошёл в ваш продукт. Пакетный анализ также дает возможность обнаружить при каких исходных данных получается

идеальная партия, и приблизить к ним все операции для получения продукта наилучшего качества.

4. Нормативные отчеты. Предоставление записей о выбросах углерода или сбросов сточных вод для государственных и федеральных организаций. Также эти данные могут быть использованы, чтобы уменьшить количество отходов.

5. Управление коммунальными услугами. Если вы можете визуализировать использование энергии в предприятии, вы можете сэкономить примерно 10% энергии, просто зная, что и когда отключить. Тогда вы можете сэкономить еще 15%, автоматизируя процесс отключения приборов.

6. Экскурсии по заводу. Система SCADA - отличный способ дать им представление туристическим группам о том, что происходит на заводе. Многие организации создают экраны именно для этой цели, чтобы новичок мог понять, что происходит на заводе.

7. Контроль качества. Большинство систем SCADA предоставляют возможность архивирование истории. Это позволяет отслеживать изменение показателей качества с течением времени и, когда вы чувствуете себя хорошо, как работает ваш процесс, предоставляя диспетчерский контроль над процессом, обеспечивая согласованность действий для операторов. Ключевые показатели (Key Performance Indicator) также предоставляются через экраны SCADA для персонала завода, чтобы иметь возможность оценить правильность работы завода.

8. Непрерывная эксплуатация. Благодаря автоматическому управлению и удаленному мониторингу не нужно, чтобы на заводе 24 часа в сутки работали сотрудники. Оперативный персонал может находиться в режиме ожидания дома или удаленно, чтобы устранять проблемы, с которыми система не может справиться без их участия.

9. Управление алгоритмами - в системе SCADA может храниться библиотека алгоритмов и загружаться в программируемые логические контроллеры для выполнения. Это обеспечивает согласованность действий.

Система SCADA также может быть механизмом получения заказов на продукцию от систем ERP (Enterprise resource planning) и преобразования их в подробный рецепт, который будет производиться в нескольких технологических единицах.

10. Управление производством нескольких продуктов - Программирование SCADA может координировать операции, которые производят более одного продукта на одной конвейерной линии, обеспечивая более быструю замену и внедрение нового продукта. Это дает гибкость, необходимую для того, чтобы оставаться конкурентоспособным на рынке, где клиентам ежедневно требуются новые виды продуктов [2].

Вот некоторые из основных причин, по которым люди сегодня используют системы SCADA для управления своим предприятием. Технология SCADA очень гибкая и может решить многие проблемы, с которыми не могут справиться стандартные программы.

1.3 Основные параметры и характеристики для сравнения SCADA систем

SCADA-системы можно рассматривать и оценивать по многим параметрам. Ниже приведены некоторые из них, используемые при сравнении в большинстве рассмотренных русскоязычных и иностранных источников:

- Поддерживаемые операционные системы;
- Языки интерфейса;
- Масштабируемость (максимальное поддерживаемое количество переменных, объектов, станций оператора);
- Возможность автоматического создания отчетов, поддерживаемые форматы для их формирования;
- Возможности по построению трендов;
- Поддержка алармов, OPC, ODBC / SQL, событий, контролов;
- Наличие демоверсии программы;

- Возможности для организации Web интерфейса;
- Графические возможности программы;
- Цена;
- Поддерживаемые языки программирования [3].

1.4 Оценка нескольких SCADA-систем

MasterSCADA 3.X

- Поддерживаемые операционные системы – Microsoft Windows;
- Языки интерфейса – многоязыковая поддержка;
- Масштабируемость (максимальное поддерживаемое количество переменных, объектов, станций оператора) – неограниченно;
- Возможность автоматического создания отчетов – присутствует;
- Возможности по построению трендов – присутствует;
- Поддержка алармов, OPC, ODBC / SQL – звуковые и цветовые варианты сигнализации, поддержка OPC, OLE, DCOM, ActiveX, OLE DB, ODBC, MS SQL, Oracle, InterBase, Sybase, MySQL, MS Access;
- Наличие демоверсии программы – присутствует;
- Возможности для организации Web интерфейса – интернет клиент присутствует;
- Графические возможности программы – библиотеки готовых комплексных объектов проекта, включающих графику, алгоритмы и сообщения, трехмерная графика и мультимедиа;
- Цена – 1195\$/1000 points, free 32 points demo;
- Поддерживаемые языки программирования – языки стандарта МЭК 6131-3 (ST, IL, LD, SFC, FBD) [4].

InTouch

- Поддерживаемые операционные системы – Microsoft Windows;
- Языки интерфейса – Многоязыковая поддержка;

- Масштабируемость (максимальное поддерживаемое количество переменных, объектов, станций оператора) – до 1 миллиона входов и выходов;
- Возможность автоматического создания отчетов – присутствует;
- Возможности по построению трендов – присутствует;
- Поддержка алармов, OPC, ODBC / SQL – отображение распределенных алармов (сигналов тревог и событий), поддержка OPC, DDE, MS SQL Server 2000/2005 или MSDE 2000;
- Наличие демоверсии программы – присутствует;
- Возможности для организации Web интерфейса – присутствует;
- Графические возможности программы – расширенный набор готовых мастер-объектов с индустриальной графикой, которые позволяют разработчику создавать сложные и мощные экраны операторского интерфейса быстро и легко;
- Цена – ~2000\$
- Поддерживаемые языки программирования – C, язык сценариев InTouch, Visual Basic (VBScript) [5].

TRACE MODE

- Поддерживаемые операционные системы – Microsoft Windows;
- Языки интерфейса – Многоязыковая поддержка;
- Масштабируемость (максимальное поддерживаемое количество переменных, объектов, станций оператора) – до 64000 точек;
- Возможность автоматического создания отчетов – присутствует;
- Возможности по построению трендов – присутствует;
- Поддержка алармов, OPC, ODBC / SQL – Система управления тревогами MPB, приоритеты тревог, поддержка OPC, ODBC;
- Наличие демоверсии программы – присутствует;
- Возможности для организации Web интерфейса – присутствует;
- Графические возможности программы – встроенная 3D-графика, графический редактор SCADA TRACE MODE предоставляет 132000 встроенных

3D-инструментов изображения трубопроводной арматуры и емкостей (400 базовых 3D-форм).;

- Цена – 1024 точек / 31744 руб., 64 точки / 8160 руб.;
- Поддерживаемые языки программирования – Алгоритмы управления на языках стандарта IEC 61131-3 [6].

1.5 Выводы по обзору литературы

При помощи SCADA-систем решают множество задач и немалую роль в их решении играют графические возможности программы и поэтому они становятся весомым фактором при выборе программного обеспечения для реализации SCADA-системы. Для разработки наглядной и презентабельной мнемосхемы требуется гибкий редактор и обширная база интерактивных компонентов с проработанным набором свойств, отображающих состояние процессов визуально и обеспечивающих различные способы взаимодействия с ними.

При этом системы, обладающие продвинутыми графическими возможностями, стоят значительно дороже программ с более скромными редакторами мнемосхем. Хорошим решением этой дилеммы является самостоятельная модификации программной среды при помощи экспорта компонентов, разработанных в других средах. Это позволяет существенно расширить возможности программы без значительной переплаты за программный продукт. Самостоятельная доработка программной среды зачастую является наиболее выгодным решением.

Чтобы разобраться в механизме дополнения базы компонентов программной среды и получить соответствующие навыки, целью работы было выбрано расширение графических возможностей программы MasterSCADA путём разработки и интеграции в программу функциональных блоков.

2 СРЕДСТВА И МЕТОДЫ РАЗРАБОТКИ

2.1 MasterSCADA

MasterSCADA — программный пакет для проектирования систем диспетчерского управления и сбора данных (SCADA). Основными свойствами является модульность, масштабируемость и объектный подход к разработке. Система предназначена для сбора, архивирования, отображения данных, а также для управления различными технологическими процессами. Помимо создания т. н. верхнего уровня, система позволяет программировать контроллеры с открытой архитектурой. Таким образом MasterSCADA позволяет создавать единый комплексный проект автоматизации (SCADA система + ПЛК). Вся система, включая все компьютеры и все контроллеры, конфигурируется в едином проекте, за счет этого не требуется конфигурировать внутренние связи в системе [4].

2.1.1 Функциональные блоки в MasterSCADA

В поставку MasterSCADA входит набор библиотек, содержащий более ста различных функциональных блоков, расположенных в Палитре ФБ. Функциональные блоки позволяют решать необходимые разработчику задачи — проводить математическую и логическую обработку, работать со временем и архивом, управлять исполнительными механизмами, осуществлять операции с документами.

Наряду с чисто алгоритмическими функциональными блоками, в MasterSCADA используются визуальные функциональные блоки (ВФБ), которые наряду с алгоритмом имеют отображение на мнемосхемах и окна управления. Каждый ВФБ полностью реализует все необходимое для контроля и управления тем типом исполнительного механизма или объекта, для которого он был создан [7].

2.2 Технологический стандарт Component Object Model

Component Object Model (COM) - это стандарт бинарного интерфейса для

программных компонентов, представленный Microsoft в 1993 году. Он используется для создания объектов межпроцессного взаимодействия между множеством языков программирования. COM является основой для нескольких других технологий и платформ Microsoft, включая OLE, OLE Automation, Browser Helper Object, ActiveX, COM +, DCOM, Windows shell, DirectX, UMDF и Windows Runtime. Суть технологического стандарта COM - это независимый от языка способ реализации объектов, который можно использовать в средах, отличных от той, в которой они были созданы, даже на другом устройстве. Для написанных проверенными авторами компонентов COM позволяет повторно использовать объекты, не зная об их внутренней реализации, так как вынуждает разработчиков компонентов предоставлять четко определенные интерфейсы, которые отделены от реализации. Различная семантика распределения языков обеспечивается тем, что объекты отвечают за свое создание и уничтожение посредством считывания ссылок (reference-counting). Преобразование типов между различными интерфейсами объекта достигается с помощью метода QueryInterface. Предпочтительным методом «наследования» в COM является создание подобъектов, которым делегируются «вызовы» метода [8].

COM - это интерфейсная технология, определенная и реализованная в качестве стандарта только в Microsoft Windows и Apple Core Foundation 1.3 и более поздних интерфейсах прикладного программирования (API). Последний реализует только подмножество всего COM-интерфейса. Для некоторых приложений COM был заменен, по крайней мере, до некоторой степени платформой Microsoft .NET и поддержкой веб-служб через Windows Communication Foundation (WCF). Однако COM-объекты можно использовать со всеми языками .NET через .NET COM Interop. Сетевой DCOM использует собственные двоичные форматы, а WCF поощряет использование сообщений SOAP на основе XML. COM очень похож на другие технологии интерфейса программных компонентов, такие как CORBA и Enterprise JavaBeans, хотя у каждого есть свои сильные и слабые стороны. В отличие от C ++, COM обеспечивает стабильный двоичный интерфейс приложения (ABI), который не

меняется между выпусками компилятора. Это делает СОМ-интерфейсы привлекательными для объектно-ориентированных библиотек С ++, которые должны использоваться клиентами, скомпилированными с использованием разных версий компилятора.

2.3 Программная платформа .NET Framework

NET Framework (произносится как «дот нэт») представляет собой программную платформу, разработанную Microsoft, которая работает в основном на Microsoft Windows. Она включает в себя большую библиотеку классов, которая называется Framework Class Library (FCL) и обеспечивает взаимодействие языков. Программы, написанные для .NET Framework, называются Common Language Runtime (CLR). CLR - это виртуальная машина приложения, которая обеспечивает такие функции, как безопасность, управление памятью и обработка исключений. Таким образом, компьютерный код, написанный с использованием .NET Framework, называется «управляемый код» ("managed code"). FCL и CLR вместе составляют .NET Framework [9].

FCL обеспечивает пользовательский интерфейс, доступ к данным, подключение к базе данных, криптографию, разработку веб-приложений, численные алгоритмы и сетевые коммуникации. Программисты создают программное обеспечение для объединения своих библиотек. Фреймворк предназначен для платформы Windows. Microsoft также предоставляет интегрированную среду разработки для программного обеспечения .NET, которая называется Visual Studio.

2.4 Среда разработки программного обеспечения Visual Studio

Microsoft Visual Studio - это интегрированная среда разработки (IDE) от Microsoft. Он используется для разработки компьютерных программ, а также веб-сайтов, веб-приложений, веб-сервисов и мобильных приложений. Visual Studio использует платформы разработки программного обеспечения Microsoft, такие как Windows API, Windows Forms, Windows Presentation Foundation,

Windows Store и Microsoft Silverlight. Он может производить как собственный код, так и управляемый код (managed code) [10].

Visual Studio включает в себя редактор кода, поддерживающий IntelliSense (вспомогательный компонент, упрощающий написание кода), а также рефакторинг кода. Интегрированный отладчик работает как отладчик на уровне исходного кода и отладчик машинного уровня. Другие встроенные инструменты включают профилировщик кода, конструктор форм для создания приложений с графическим интерфейсом, веб-дизайнер, дизайнер классов и конструктор схем базы данных. Он поддерживает плагины, которые расширяют функциональные возможности практически на каждом уровне, включая добавление поддержки систем контроля версий (таких как Subversion и Git) и добавление новых наборов инструментов, таких как редакторы и визуальные дизайнеры, для языков, специфичных для предметной области, или наборов инструментов для других аспектов разработки программного обеспечения. жизненный цикл (например, клиент Team Foundation Server: Team Explorer).

Visual Studio поддерживает 36 различных языков программирования и позволяет редактору кода и отладчику поддерживать (в разной степени) практически любой язык программирования при условии, что существует специальная языковая служба. Встроенные языки включают C, C ++, C ++ / CLI, Visual Basic .NET, C#, F#, JavaScript, TypeScript, XML, XSLT, HTML и CSS. Поддержка других языков, таких как Python, Ruby, Node.js и М среди других, доступна через плагины. Java (и J #) поддерживались в прошлом.

2.5 Объектно-ориентированный язык программирования C#

C# - это универсальный мультипарадигмальный язык программирования, включающий строгую типизацию, лексическую область видимости, императив, декларативный, функциональный, универсальный, объектно-ориентированный (на основе классов) и компонентно-ориентированный дисциплины программирования. Он был разработан Microsoft примерно в 2000 году в рамках инициативы .NET, а затем утвержден в качестве стандарта Ecma (ECMA-334) и

ISO (ISO / IEC 23270: 2018). C# является одним из языков программирования, разработанных для инфраструктуры общего языка.

C# был разработан Андерсом Хейлсбергом, а его командой разработчиков в настоящее время руководит Мадс Торгерсен. Самая последняя версия - C # 7.3, выпущенная в 2018 году вместе с Visual Studio 2017 версии 15.7.2 [11].

3 РАЗРАБОТКА БИБЛИОТЕКИ ДЛЯ MASTERSCADA

3.1 Создание и настройка файла проекта

Был создан файл проекта, в котором планировалось реализовать функциональный блок: Файл-> Создать-> Проект... Был выбран тип проекта библиотека классов (.Net Standart) (см. рисунок 2).

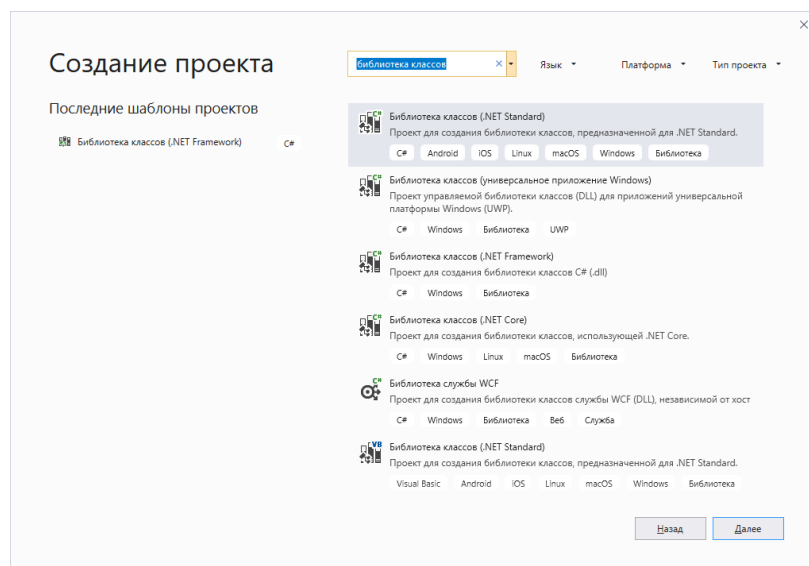


Рисунок 2 – Создание файла проекта

На закладке «Сборка» было необходимо отметить пункт «Зарегистрировать для взаимодействия COM» и указать «Выходной путь» на директорию, в которую установлена MasterSCADA (см. рисунки 3 и 4).

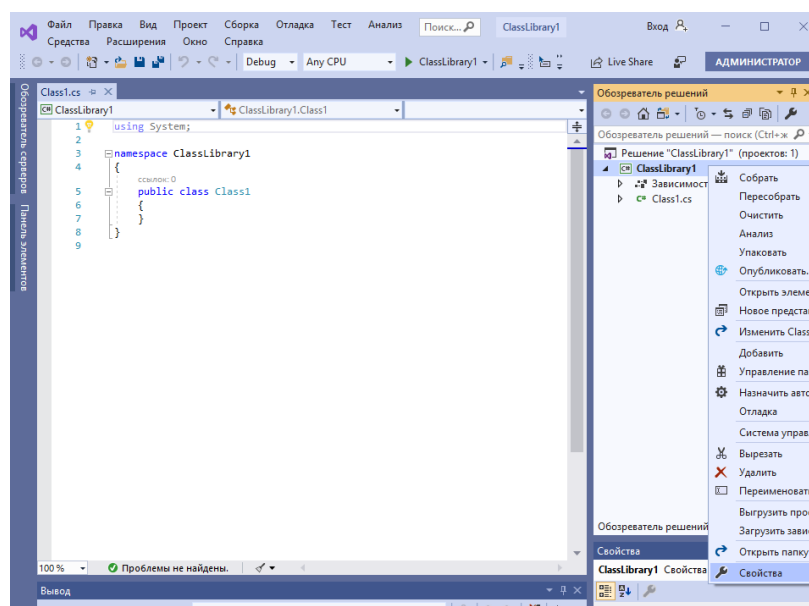


Рисунок 3 – Настройка сборки проекта

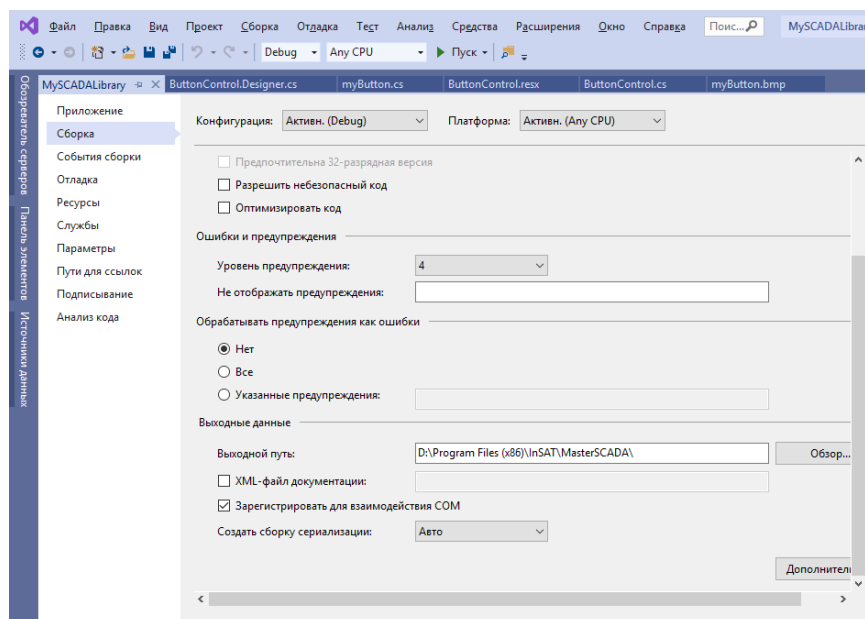


Рисунок 4 – Настройка сборки проекта

Далее были добавлены ссылки на сборки, расположенные в директории с MasterSCADA. Для этого правой кнопкой мыши было открыто контекстное меню элемента “Ссылки” проекта и выбран пункт “Добавить ссылку...” и в открывшемся диалоге выбрана закладка “Обзор”, нажата кнопка “Обзор...” и в папке, в которую установлена MasterSCADA, были отмечены следующие dll-файлы: “FB.dll”, “Insat.Opc.dll”, “InSAT.Library.dll”, “MasterSCADALib.dll”, “InternalModules.dll”, “MasterSCADA.Common.dll” (см. рисунок 5) [12].

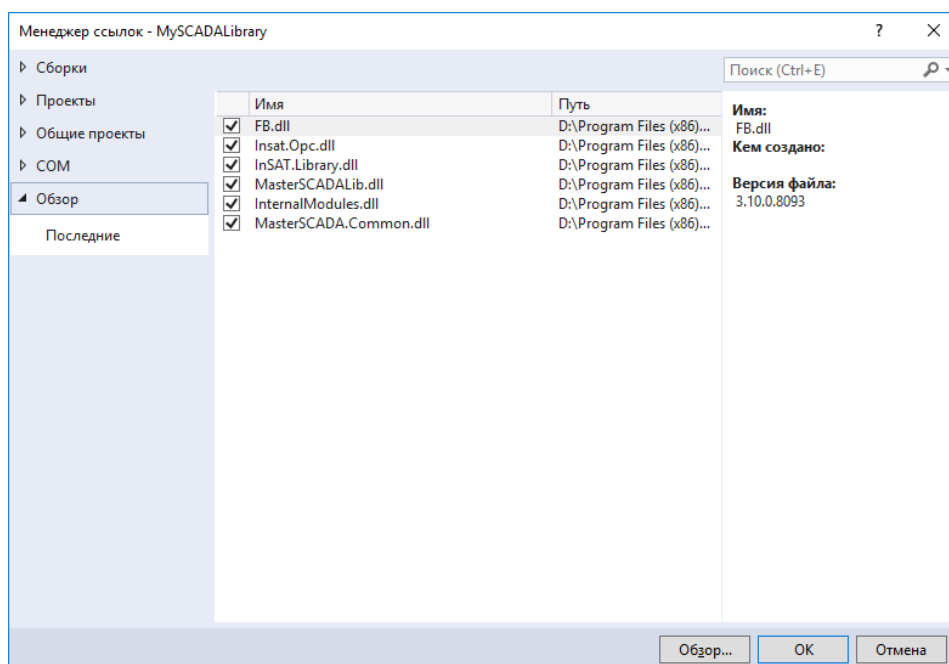


Рисунок 5 – Настройка ссылок проекта

3.2 Выбор набора компонентов и их свойств для реализации в библиотеке

На этом этапе разработки были определены компоненты для использования на мнемосхемах в среде MasterSCADA, которые будут разработаны. Для исследования были выбраны два наиболее часто используемых в проектах визуальных функциональных блока: кнопка и клапан.

Для каждого из этих компонентов были определены необходимые для отображения и использования свойства. Эти свойства указаны в таблице 1.

Таблица 1 – Свойства и параметры разрабатываемых функциональных блоков

	Кнопка MPButton	Клапан MPValve
Графические возможности	Вывод на тело кнопки различного редактируемого в MasterSCADA текста в зависимости от состояния (true или false); Заливка тела кнопки различными цветами, которые можно настроить в MasterSCADA, в зависимости от состояния (true или false).	Заливка тела клапана различными цветами, которые можно настроить в MasterSCADA, в зависимости от состояния (true или false).
Параметры (входы и выходы)	Вход “Состояние” позволяет определять нажата кнопка или нет из внешнего источника, которым может являться, например, переменная, получаемая из OPC-сервера;	Вход “Состояние” позволяет определять открыт клапан или нет из внешнего источника, которым может являться, например, переменная, получаемая из OPC-сервера; Вход “Процент” позволяет определять степень открытия клапана числом формата float от 0 до 1.

Продолжение таблицы 1 – Свойства и параметры разрабатываемых функциональных блоков

	Вход “Блок” позволяет заблокировать кнопку, т.е. при значении true этого поля нажатие кнопки на мнемосхеме будет проигнорировано; Выход “Выход” позволяет использовать состояние кнопки как переменную в других частях алгоритма.	Вход “Блок” позволяет заблокировать клапан в текущем состоянии. Выход “Выход” позволяет использовать состояние клапана как переменную в других частях алгоритма. Выход “Процент” позволяет использовать степень открытия клапана, представляющую из себя число формата float от 0 до 1.
--	---	--

3.3 Описание компонентов на языке C#

Для дальнейшей работы необходимо создать новый класс: у проекта в контекстном меню было выбрано “Добавить” ->” Создать элемент...” (см. рисунок 6).

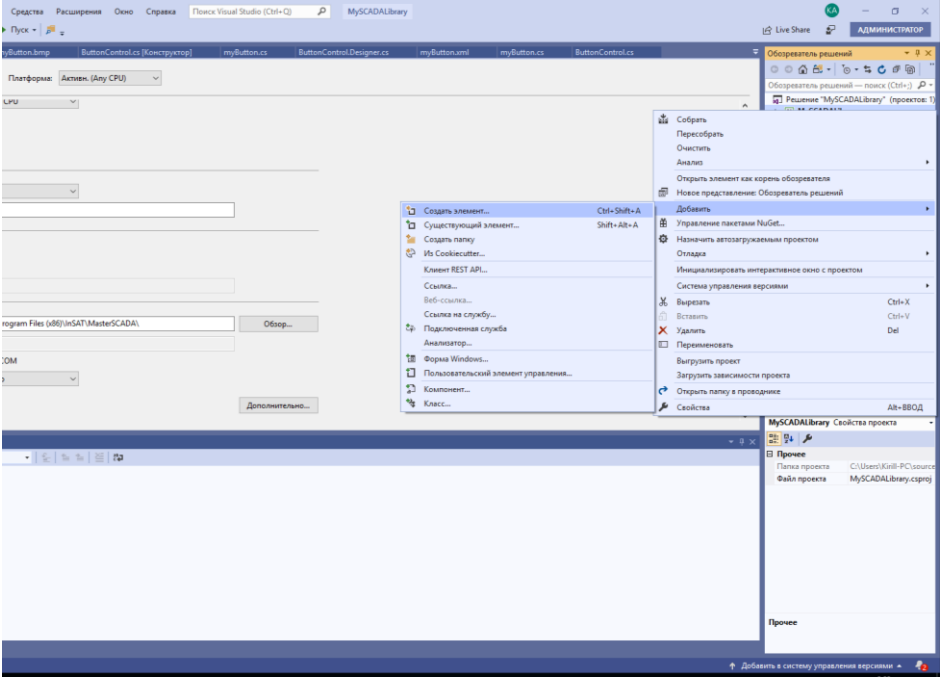


Рисунок 6 – Создание класса функционального блока

В открывшемся диалоге был выбран шаблон “Класс” (см. рисунок 7).

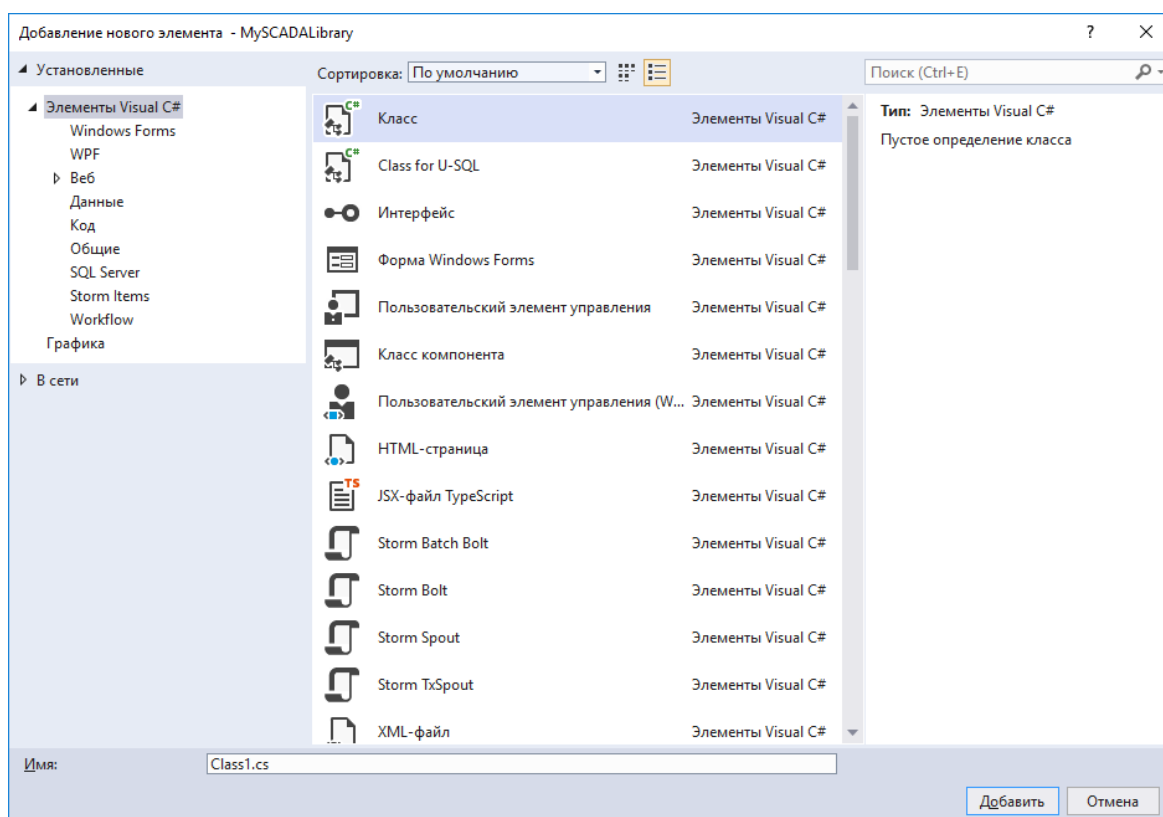


Рисунок 7 – Создание класса функционального блока

Для того, чтобы некоторыми пользоваться специальными командами необходимо унаследовать их от класса `FB.StaticFBBase`.

Также нужно добавить атрибут, определяющий метод сохранения ФБ в проекте MasterSCADA. Возможны два варианта: “`Serializable`” – для бинарного сохранения ФБ или “`SerializationType(SerializationType.Xml)`” – для сохранения настроек ФБ в текстовом виде в формате XML.

Кроме того, для корректной работы нужны атрибуты `ComVisuable` и `GuID` (Значение атрибута `Guid` должно быть уникальным, уникальное значение можно получить при помощи Visual Studio (“Средства”->” Создать GUID” в открывшемся окне требуется выбрать в качестве формата GUID “Формат реестра”)) для регистрации Функционального блока как COM объекта (см. рисунки 8 и 9).

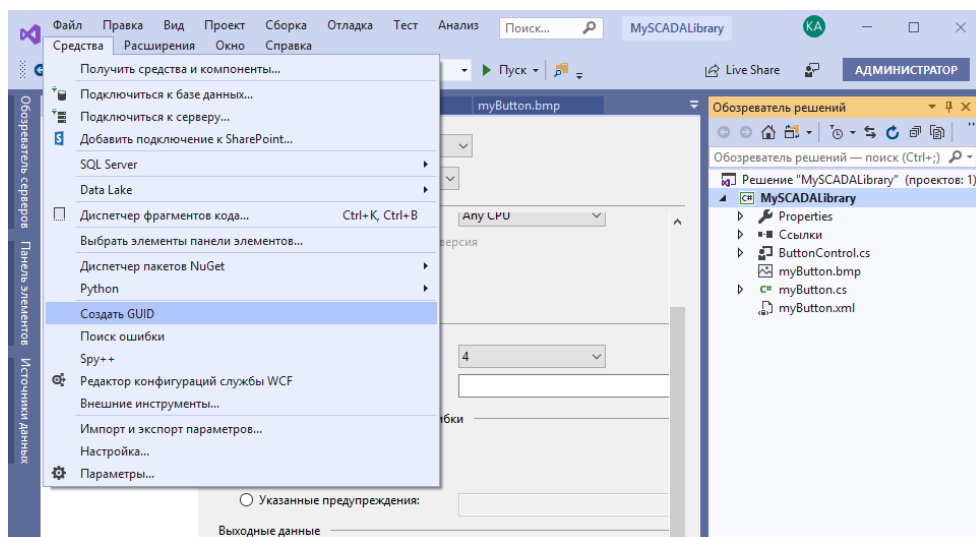


Рисунок 8 – Создание GUID

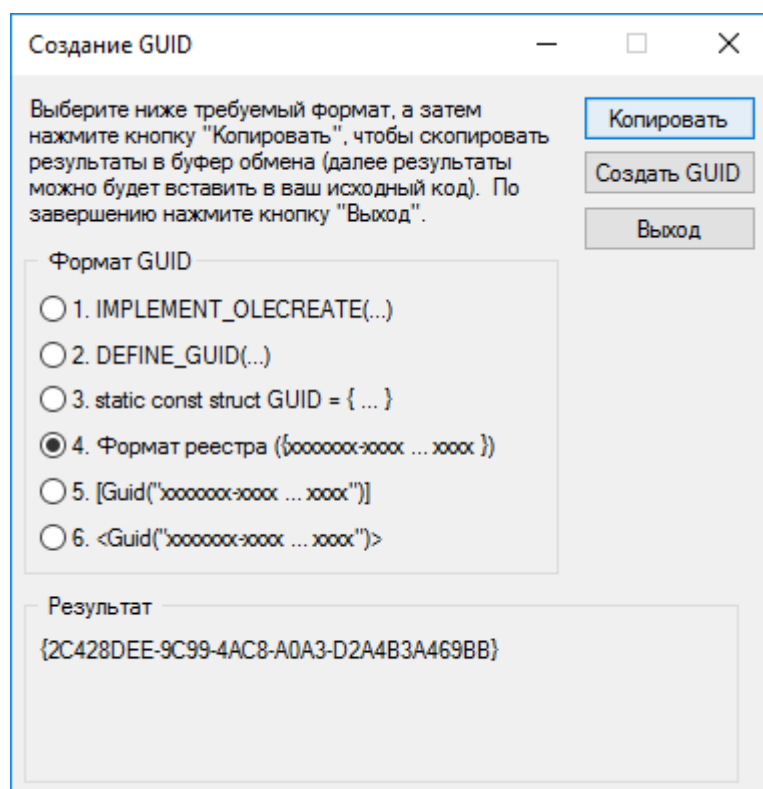


Рисунок 9 – Создание GUID

Были прописаны атрибуты CatID (определяет категорию палитры, в которой будет расположен функциональный блок) и Displayname (определяет имя блока) для регистрации класса в качестве функционального блока (см. рисунок 10).

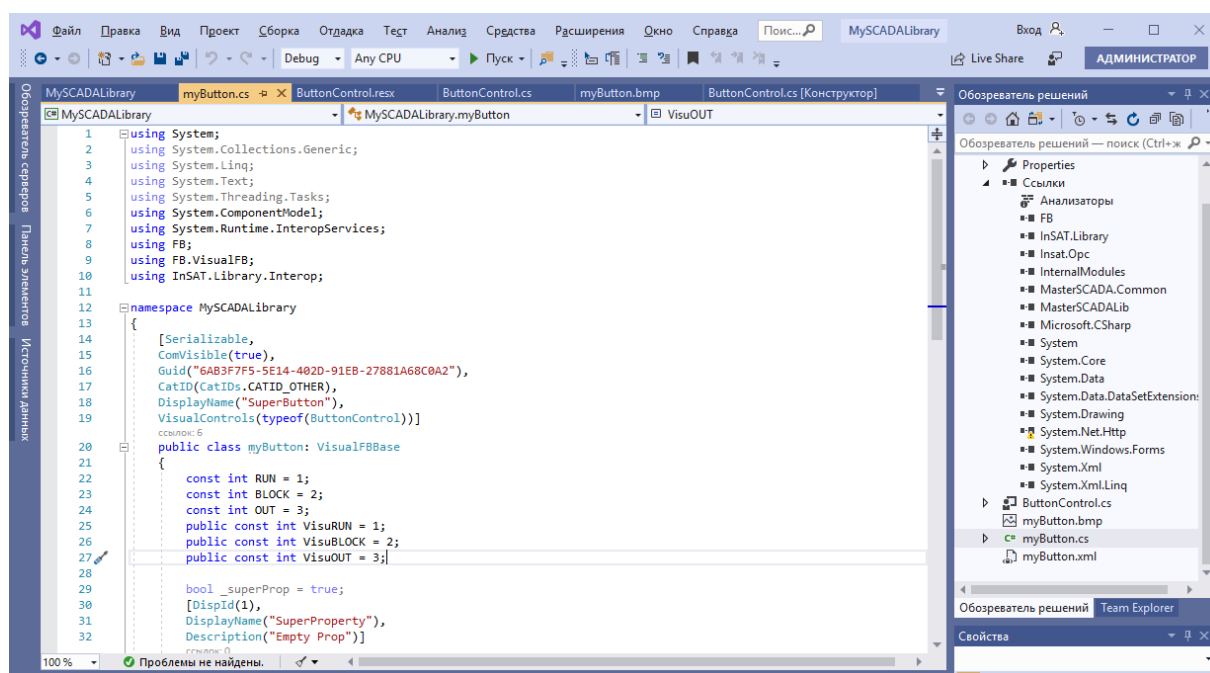


Рисунок 10– Описание класса функционального блока

Также был создан Xml-файл с настройками и Bmp-файл с изображением функционального блока. Эти файлы должны быть созданы в той же подпапке проекта что и файл класса ФБ. У этих файлов необходимо в свойстве “Действие при сборке” выставить вариант “Внедрённый ресурс”. Также важно учитывать, что Bmp-файл изображения должен быть шестнадцатичетным и быть размера 16 на 16 пикселей.

Далее необходимо описать выбранные свойства и параметры функционального блока на языке C# в файле с расширением cs. То, как это было реализовано представлено в листинге в приложении А.

4 РЕЗУЛЬТАТЫ ПРОВЕДЁННОЙ РАБОТЫ

4.1 Оценка результатов работы

Было рассмотрено понятие SCADA-системы, определены основные цели и задачи, которые ставит внедряющая систему компания перед разработчиком, были выделены основные параметры и характеристики для сравнения SCADA-систем, было проведено сравнение нескольких SCADA-систем. из этой части работы был сделан вывод, что модификация графических возможностей SCADA-системы может быть наиболее выгодным решением при её реализации.

В ходе проведения данной работы было рассмотрено и изучено множество понятий, понимание которых необходимо для выполнения задания. А именно: изучена программная среда MasterSCADA и понятие функционального блока в ней, рассмотрен технологический стандарт Component Object Model и программная платформа .NET Framework, освоена среда разработки программного обеспечения Visual Studio , в которой была реализована библиотека функциональных блоков и объектно-ориентированный язык программирования C#, на котором была написана большая часть программного кода библиотеки.

Была проделана работа по оценке необходимого набора компонентов и их свойств для реализации в библиотеке, реализован и зафиксирован алгоритм разработки функциональных блоков.

4.2 Преимущества разработанных компонентов над предустановленными

При разработке функциональных блоков одна из главных задач – понять какие свойства для динамизации их внешнего вида и параметры для использования в алгоритмах управления требуется реализовать. Именно это и определяет его полезность и его преимущества над уже реализованными функциональными блоками в предустановленных библиотеках. Поэтому главным преимуществом разработанных самостоятельно функциональных

блоков, на мой взгляд, является расширенный определённый самостоятельно под конкретные задачи набор свойств и параметров.

Например, в разработанном в ходе данной работы функциональном блоке MPButton присутствуют следующие графические возможности, не предусмотренные в стандартном предустановленном компоненте “Кнопка”:

1. Вывод на тело кнопки различного редактируемого в MasterSCADA текста в зависимости от состояния (true или false);
2. Заливка тела кнопки различными цветами, которые можно настроить в MasterSCADA, в зависимости от состояния (true или false).

Внешний вид кнопок и соответствующих их графическому представлению свойств можно видеть на рисунке 11.

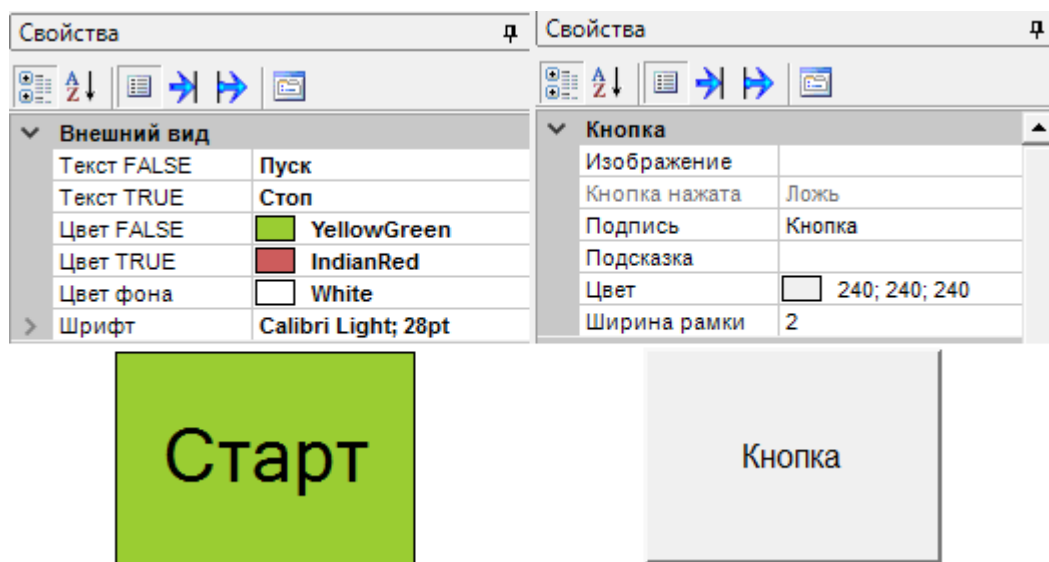


Рисунок 11 – Внешний вид кнопок и соответствующих их графическому представлению свойств

Другим важным преимуществом блока MPButton является то, что он отображается в дереве проекта как объект и у него есть ряд полей, которые можно использовать в алгоритме управления. Эти поля могут использоваться для изменения как состояния самой кнопки извне, так и для передачи её состояние в другие блоки или программы.

У разработанного в ходе данной работы блока MPButton имеются следующие параметры:

1. Вход “Состояние” позволяет определять нажата кнопка или нет из внешнего источника, которым может являться, например, переменная, получаемая из OPC-сервера;
2. Вход “Блок” позволяет заблокировать кнопку, т.е. при значении true этого поля нажатие кнопки на мнемосхеме будет проигнорировано;
3. Выход “ Выход” позволяет использовать состояние кнопки как переменную в других частях алгоритма.

На рисунке 12 представлено то, как функциональный блок MPButton и его входы выходы отображаются в дереве объектов программной среды MasterSCADA.

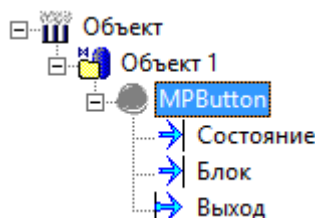


Рисунок 12 – Функциональный блок MPButton и его входы выходы

4.3 Возможность использования разработанной библиотеки для других SCADA-систем

Возможность использования разработанной библиотеки в других SCADA-систем ограничено прежде всего тем, что не все SCADA поддерживают программную платформу .NET Framework и технологический стандарт Component Object Model. Тем не менее, в большинстве популярных SCADA-систем таких как, например, Wonderware InTouch и Zenon такие компоненты могут быть использованы. Из этого следует, что разработанная библиотека может быть доработана и применяться в других программных средах. При этом при разработке функциональных блоков для других сред алгоритм разработки, в целом, остаётся прежним, но некоторые моменты будут отличаться.

Прежде всего, для разработки будут использоваться ссылки на другие сборки формата dll. Сборки, в свою очередь, будут содержать другие команды,

что может значительно повлиять на вид как программного кода, так и проекта в целом. Так же для некоторых SCADA-систем предпочтительнее будет использовать при разработке другие программные среды. Так, например, для системы InTouch компании Wonderware при разработке функциональных блоков, которые в англоязычной терминологии называются control, активно используется другая программа от той же компании – Wonderware ArchestrA Graphics [13].

Иногда для использования библиотек .NET потребуется значительно переработать проект. Так, например, в Zenon функциональные блоки на .NET, созданные в Visual Studio не поддерживаются по умолчанию, так что для их применения в этой среде лучшим решением по мнению разработчика будет создать двойной функциональный блок (dual control), который может использоваться в контейнерах управления ActiveX (ActiveX Control Containers) [14].

5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ

На сегодняшней день перспективностью научных разработок является не только сам процесс открытия, но и оценка его коммерческой ценности, что становится необходимым условием для поиска источников финансирования с целью проведения научного исследования.

Коммерческая привлекательность исследуемого проекта должна включать в себя не только повышение качества усовершенствованных технологических разработок наряду с предыдущими, но и его востребованность на соответствующем рынке, ценовая категория, бюджет самого научного исследования, а также срок, в течение которого продукт выйдет на потребительский рынок.

Следовательно, целью текущего раздела становится проектирование, создание конкурентоспособных разработок, которые отвечают основным требованиям в области ресурсосбережения и ресурсоэффективности. В данном разделе осуществлена оценка разработки базы компонентов для SCADA-системы.

5.1 Потенциальные потребители результатов исследования

Целевой аудиторией рынка, на которую направлена основная часть продаж текущей разработки, является отрасли моделирования технологических процессов, автоматизации и мехатроники, для которых данное исследование имеет практический интерес, поскольку позволяет снизить стоимость и увеличить функциональные возможности разрабатываемой SCADA-системы. Сегментирование рынка произведено по следующим критериям: размер компании и стоимость разработки в зависимости от программной среды, в которой разрабатывается SCADA-система, представлено в таблице 2.

Таблица 2 – Карта сегментирования

		Область применения			
		Электроэнергетика	Промышленное производство	Нефтегазовая отрасль	Водозабор, водоподготовка, распределение воды
Размер компании	Крупные				
	Средние				
	Мелкие				

Согласно карте сегментирования, SCADA-системы, разработанные в программной среде MasterSCADA, дополненной разработанными в ходе данной работы компонентами, подходят для мелких и средних компаний отраслей электроэнергетики, промышленного производства, нефтегазодобычи, водозабора, водоподготовка и распределение воды, а также для крупных компаний нефтегазовой и водообрабатывающей отраслей. При этом стоимость разработки систем остаётся относительно небольшой.

5.2 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 конкурирующих производителей SCADA-систем можно выделить следующих: InTouch (конкурент 1), TRACE MODE (конкурент 2).

SCADA-система InTouch – это достаточно мощная среда разработки визуализации и управления, разработанная британской компанией Aveva, для промышленной автоматизации технологических процессов и диспетчерского

контроля. SCADA система InTouch применяется для создания DCS (распределенных систем управления) и других АСУ ТП.

SCADA-система TRACE MODE - это первая интегрированная информационная система для управления промышленным производством, объединяющая в единые целые продукты класса SOFTLOGIC-SCADA / HMI-MES-EAM-HRM. Разработана российской компанией AdAstra Research Group,

Анализ приведен в таблице 3.

Таблица 3 – 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							
1. Предоставляемые возможности	0,1	5	5	5	0,5	0,5	0,5
2. Повышение производительности труда пользователя	0,1	5	5	4	0,5	0,5	0,4
3. Надежность	0,07	5	5	5	0,35	0,35	0,35
4. Удобство эксплуатации	0,15	5	4	2	0,75	0,6	0,3
5. Масштабируемость	0,1	5	5	4	0,5	0,5	0,4
6. Качество интерфейса	0,1	5	4	3	0,5	0,4	0,3
Экономические критерии оценки эффективности							
1. Конкурентоспособность	0,105	5	5	3	0,525	0,525	0,315

Продолжение таблицы 3 – Оценочная карта для сравнения конкурентных технических решений

2. Уровень проникновения на рынок	0,15	3	5	2	0,45	0,75	0,3
3. Цена	0,1	5	2	4	0,5	0,2	0,4
4. Предполагаемый срок эксплуатации	0,025	5	5	4	0,125	0,125	0,1
Итого	1	48	45	36	4,7	4,45	3,365

Основными преимуществами перед конкурирующими SCADA-системами являются масштабируемость, качество интерфейса, низкая цена и удобство эксплуатации, таким образом коэффициент конкурентоспособности MasterSCADA, дополненной разработанными компонентами:

$$K_{kc} = \frac{\frac{4,7}{4,45} + \frac{4,45}{3,365}}{2} = \frac{1,056 + 1,322}{2} = 1,189$$

Поскольку $K_{kc} > 1$, то данная SCADA является конкурентоспособной.

5.3 SWOT-анализ

Для исследования внутренней и внешней среды проекта был проведен комплексный анализ научно-исследовательского проекта – SWOT-анализ. Итоговая матрица SWOT-анализа, полученная в результате реализации всех этапов исследования, приведена в таблице 4.

Таблица 4 – Матрица SWOT-анализа

	Сильные стороны научно-исследовательского проекта: С1. Низкая стоимость; С2. Простота эксплуатации; С3. Модульность проекта; С4. Экологичность технологии; С5. Русскоязычная служба поддержки.	Слабые стороны научно-исследовательского проекта: Сл1. Недоработанность системы; Сл2. Малая база компонентов; Сл3. Невозможность пользователю дорабатывать и расширять компонентную базу без сторонних программ.
Возможности: В1. Осваивание новых отраслей; В2. Выход на иностранный рынок; В3. Появление дополнительного спроса на усовершенствованный продукт; В4. Повышение стоимости разработок конкурентов; В5. Хорошие связи с потребителями.	За счёт возможности расширения функционала проекта существует возможность осваивания новых отраслей и выхода на иностранный рынок. Простота эксплуатации позволяет уменьшить затраты на обучение персонала, а также дает возможность использования проекта иностранными клиентами. Санкции, наложенные на РФ, и высокий курс евро/доллара будут ограничивать появление новых иностранных технологий на российском рынке, и низкая цена нашей	Активная работа службы поддержки и сохранение решений на тематических форумах позволят минимизировать вред от ошибок и доработать систему. Освоение новых отраслей потребует внедрения новых компонентных баз. Вследствие этого, система начнёт удовлетворять запросам большего числа пользователей.

Продолжение таблицы 4 – Матрица SWOT-анализа

	системы позволит ей конкурировать с аналогами. То, что MasterSCADA – программа русских разработчиков, позволяет поддерживать хорошие связи с потребителями	
Угрозы: У1. Нацеленность на ограниченный круг компаний; У2. Появление более современных и дешевых SCADA-систем; У3. Отсутствие спроса на новые технологии производства.	Значительной угрозой может стать аналог, имеющий сравнительно меньшую стоимость и разработанный по средствам современных технологий.	Текущая разработка может быть не востребована покупателем в связи с тем, что лишь малая часть компаний заинтересована и способна финансово внедрять АСУ ТП. Малая компонентная база сужает круг компаний, которые смогут использовать систему. Пользователи могут отказаться от системы при появлении аналогов с более широкими возможностями редактирования и разработки по более низкой цене.

Таким образом, с использованием сильных сторон проекта имеется перспектива реализации возможностей. Низкая стоимость, простота эксплуатации и модульность проекта позволяют в полной мере использовать все возможности для развития проекта. Но слабые стороны и наличие угроз снижают конкурентоспособность продукта.

5.4 Планирование научно-исследовательских работ

Структура работ в рамках научного исследования

Планирование комплекса предполагаемых работ осуществляется в следующем порядке:

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

Для реализации проекта необходимы исполнители в лице руководителя и инженера. Перечень этапов, работ и исполнителей приведен в таблице 5.

Таблица 5 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ раб.	Содержание работ	Должность исполнителя
Постановка целей и задач, получение исходных данных	1	Составление и утверждение задания на ВКР	Руководитель
Теоретические исследования	2	Выбор направления исследований	Инженер
	3	Подбор и изучение материалов по теме	Инженер
	4	Разработка календарного плана	Руководитель, инженер
Разработка и проектирование системы	5	Проведение теоретических расчетов и обоснований	Инженер
	6	Создание программной части	Инженер
	7	Тестирование и отладка работы системы	Руководитель, инженер
Обобщение и оценка результатов	8	Оценка полученных результатов	Руководитель
Оформление отчета по НИР (комплекта документации по ОКР)	9	Проведение технико-экономических расчетов и оценка безопасности и экологичности проекта	Инженер
	10	Составление пояснительной записки	Инженер

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 $t_{ожи}$ используется следующая формула (1):

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

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

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

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

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

$$T_{pi} = \frac{t_{ожи}}{Ч_i}, \quad (2)$$

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

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

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

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

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

где T_{ki} – продолжительность выполнения i -ой работы в календарных днях;
 T_{pi} – продолжительность выполнения i -ой работы в рабочих днях;
 $k_{\text{кал}}$ – коэффициент календарности.

Коэффициент календарности определяется по формуле:

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

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

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

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

Полученные данные сведены в таблицу 6.

Таблица 6 – Временные показатели проведения научного исследования

Название работы	Трудоемкость работы						Длительность работы в рабочих днях, T _{pi}		Длительность работы в календарных днях, T _{ki}	
	t _{min} , чел- дни		t _{max} , чел- дни		t _{ож} , чел- дни					
	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер
Составление и утверждение задания на ВКР	2		6		3.6		3.6		5	
Выбор направления исследований		5		10		7		7		9
Подбор и изучение материалов по теме		3		5		3.8		3.8		5

Продолжение таблицы 6 – Временные показатели проведения научного исследования

Разработка календарного плана	2	2	3	3	3	3	1.5	1.5	2	2
Проведение теоретических расчетов и обоснований		2		6		3.6		3.6		5
Создание программной части		18		24		20.4		20.4		25
Тестирование и отладка работы системы		10		14		12.5		12.5		16
Оценка полученных результатов	3	3	5	5	3.8	3.8	1.9	1.9	3	3
Проведение технико-экономических расчетов и оценка безопасности и экологичности проекта		6		15		9.6		9.6		12
Составление пояснительной записки		8		13		10		10		12
Итого	7	57	14	95	10.4	73.7	7	70,3	10	89

На основе полученной таблицы 5 строится календарный план-график. График строится для максимального по длительности исполнения работ с разбивкой по месяцам и декадам. График работ приведен в таблице 7.

Таблица 7 – Календарный план-график выполнения проекта

№ работ	Вид работ	Исполнители	T _{кi} , кал. дн.	Продолжительность выполнения работ																	
				Янв.			февр.			март			апр.			май			июнь		
				1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
1	Составление и утверждение задания на ВКР	Руководитель	5				■														
2	Выбор направления исследований	Инженер	9				■	■													
3	Подбор и изучение материалов по теме	Инженер	5					■													
4	Разработка календарного плана	Руководитель, инженер	2					■	■												
5	Проведение теоретических расчетов и обоснований	Инженер	5						■												
6	Создание программной части	Инженер	25							■	■	■									
7	Тестирование и отладка работы системы	Инженер	16										■	■	■						
8	Оценка полученных результатов	Руководитель, инженер	3											■	■						
9	Проведение технико-экономических расчетов и оценка безопасности и экологичности проекта	Инженер	12											■	■	■					
10	Составление пояснительной записки	Инженер	12													■	■	■			

■ – руководитель

■ – инженер

5.5 Бюджет научно-технического исследования (НТИ)

Расчет материальных затрат НТИ

Расчет материальных затрат осуществляется по следующей формуле:

$$З_{\text{м}} = (1 + k_{\text{т}}) \cdot \sum_{i=1}^m \Pi_i \cdot N_{\text{расхи}}, \quad (5)$$

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

$N_{\text{расхи}}$ – количество материальных ресурсов i -го вида, планируемых к использованию при выполнении научного исследования;

Π_i – цена приобретения единицы i -го вида потребляемых материальных ресурсов;

$k_{\text{т}}$ – коэффициент, учитывающий транспортно-заготовительные расходы.

В таблице 8 сведены данные о материальных затратах на научное исследование.

Таблица 8 – Материальные затраты

Наименование	Единица измерения	Количество	Цена за ед., руб.	Затраты на материалы, (Зм), руб.
Пакет Master SCADA	Шт.	1	71 493	71 493
Системный блок TC-Office	Шт.	1	17 780	17 780
Монитор C24F390F	Шт.	1	8 114	8 114
Клавиатура CBR KB 111	Шт.	1	450	450
Мышь Oklick 155M	Шт.	1	350	350
Контроллер WAGO_750-881	Шт.	1	36 580	36 580
Соединительные провода	Шт.	3	64	192
Всего за материалы				134 959
Транспортно-заготовительные расходы (3 %)				4048,8
Итого				139 008

Основная заработная плата исполнителей темы

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

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

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

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

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

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

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

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

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

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

Среднедневная зарплата рассчитывается по формуле:

$$З_{дн} = \frac{З_m \cdot M}{F_d}, \quad (8)$$

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

M – количество месяцев работы без отпуска в течение года:

при отпуске в 48 раб. дней $M = 10,4$ месяца, 6-дневная неделя;

F_d – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн.

Баланс рабочего времени приведен в таблице 9.

Таблица 9 – Баланс рабочего времени

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

Месячный должностной оклад работника:

$$З_M = З_{TC} \cdot (1 + k_{np} + k_d) \cdot k_p, \quad (9)$$

где $З_{TC}$ – заработная плата по тарифной ставке, руб.;

k_{np} – премиальный коэффициент, равный 0,3;

k_d – коэффициент доплат и надбавок составляет примерно 0,2-0,5;

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

Расчет основной платы представлен в таблице 10.

Таблица 10 – Расчёт основной заработной платы

Исполнители	Оклад	k_{np}	k_d	k_p	$З_M$, руб	$З_{дн}$, руб.	T_p , раб. дн.	$З_{осн}$, руб.
Руководитель	33 664	-	-	1,3	43 763	1813,29	7	12 693,03
Инженер	12 664	-	-	1,3	16 463	682,14	70,3	47 953,82
Итого:								60 646,85

Дополнительная заработная плата исполнителей темы

Расчет дополнительной заработной платы ведется по формуле:

$$З_{дон} = k_{дон} \cdot З_{осн}, \quad (10)$$

где $k_{\text{доп}}$ – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,12 – 0,15).

Получим:

$$З_{\text{доп НР}} = 0.12 * 12\,693,03 = 1523,17$$

$$З_{\text{доп И}} = 0.12 * 47\,953,82 = 5754,46$$

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:

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

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

Расчет отчислений приведен в таблице 11.

Таблица 11 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.	Дополнительная заработная плата, руб.	Коэффициент отчислений	Отчисления
Руководитель	12 693,03	1523,17	0,271	3 852,59
Инженер	47 953,82	5754,46		14 554,94
Итого				18 407,53

Прочие прямые затраты

К данному виду затрат относятся затраты на электроэнергию. Для юридических лиц стоимость 1 кВт·ч составляет 5,8 рублей. При умеренном использовании стационарный компьютер средней мощности потребляет 180 Вт в час в среднем, а также использование монитора приводит к потреблению 40 Вт, контроллер WAGO_750-881 – 24 Вт в час, таким образом используемое оборудование потребляет в сумме 244 Вт в час. В день на работу затрачивается

5 часов, всего на работу с компьютером и оборудованием затрачивается 46 дней. Тогда затраты на электроэнергию составят:

$$З_{эн} = 0.244 \cdot 46 \cdot 5 \cdot 5,8 = 325,5 \text{ руб.}$$

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 (12):

$$C_{\text{накл}} = k_{\text{накл}} \cdot (З_{\text{осн}} + З_{\text{доп}}) \quad (12)$$

$$C_{\text{накл}} = 0,16 \cdot (60\,646,85 + 7277,63) = 10\,867,92$$

где $k_{\text{накл}} = 0,16$ – коэффициент накладных расходов.

Таким образом, величина накладных расходов составляет 10 867,92 руб.

Формирование бюджета затрат научно-исследовательского проекта

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

Определение бюджета затрат приведено в таблице 11.

Таблица 11 – Расчет бюджета затрат НТИ

Наименование статьи	Сумма, руб
1. Материальные затраты НТИ	139 008
2. Затраты по основной заработной плате исполнителей темы	60 646,85
3. Затраты по дополнительной заработной плате исполнителей темы	7 277,63
4. Отчисление во внебюджетные фонды	18 407,53

Продолжение таблицы 11 – Расчет бюджета затрат НТИ

5. Прочие расходы	325,5
6. Накладные расходы	10 867,92
7. Бюджет затрат НТИ	236 533,43

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

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

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

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

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

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

Φ_{max} зависит от сложности проекта для которого разрабатывается АСУ.

Результаты расчета интегрального финансового показателя представлены в таблице 12.

Таблица 12 – Расчет интегрального финансового показателя

	Исп 1	Исп 2	Исп 3
Стоимость варианта	236 533,43	570369,41	377355,68
Интегральный финансовый показатель	0,415	1	0,662

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

$$I_{\text{pi}} = \sum a_i \cdot b_i \quad (14)$$

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

a_i – весовой коэффициент i-го варианта исполнения разработки;

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

n – число параметров сравнения.

Сравнительный анализ приведен в таблице 13.

Таблица 13 - Сравнительная оценка вариантов исполнения

Критерии	Весовой коэффициент	Исп. 1	Исп. 2	Исп. 3
1. Предоставляемые возможности	0,2	5	5	5
2. Повышение производительности труда пользователя	0,2	5	5	4
3. Надежность	0,1	5	5	5
4. Удобство эксплуатации	0,15	5	4	2
5. Масштабируемость	0,2	5	5	4
6. Качество интерфейса	0,15	5	4	3
Итого	1	30	28	23

$$I_{p-исп1} = 0,2 \cdot 5 + 0,2 \cdot 5 + 0,1 \cdot 5 + 0,15 \cdot 5 + 0,2 \cdot 5 + 0,15 \cdot 5 = 5$$

$$I_{p-исп2} = 0,2 \cdot 5 + 0,2 \cdot 5 + 0,1 \cdot 5 + 0,15 \cdot 4 + 0,2 \cdot 5 + 0,15 \cdot 4 = 4,7$$

$$I_{p-исп3} = 0,2 \cdot 5 + 0,2 \cdot 4 + 0,1 \cdot 5 + 0,15 \cdot 2 + 0,2 \cdot 4 + 0,15 \cdot 3 = 3,85$$

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

$$I_{исп.1} = \frac{I_{p-исп1}}{I_{финр.1}} \quad (15)$$

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

$$\mathcal{E}_{cp} = \frac{I_{исп.1}}{I_{исп.2}} \quad (16)$$

Результаты расчета показателей сведены в таблицу 14.

Таблица 14 – Сравнительная эффективность разработок

Показатель	Исп. 1	Исп. 2	Исп. 3
Интегральный финансовый показатель	0,415	1	0,662
Интегральный показатель ресурсоэффективности	5	4,7	3,85
Интегральный показатель эффективности	12,048	4,7	5,816
Сравнительная эффективность вариантов исполнения	$\mathcal{E}_{12}=2,563.$ $\mathcal{E}_{13}=2,072.$		

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

6 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

Объектом исследования данной выпускной квалификационной работы является разработка SCADA-системы. Основной задачей исследования было усовершенствование интерфейса, с которым работает оператор SCADA-системы путём разработки и интеграции собственного набора функциональных блоков.

Рабочая зона оператора SCADA-системы располагается в специально оборудованном помещении, где работник занимается непосредственно своими обязанностями.

Область применения объекта - любое производство использующее автоматизированную систему управления технологическим процессом.

Правовые и организационные вопросы обеспечения безопасности

Специальные правовые нормы трудового законодательства

Существующая на сегодняшний день редакция Трудового кодекса не приравнивает работу за компьютером к вредным условиям труда. Поэтому нормативными документами, регламентирующими деятельность работника с использованием ПЭВМ, являются СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к ПЭВМ и организации труда и Инструкция по охране труда при работе с ПК. Нормативным документом СанПин 2.2.2/2.4.1340-03 установлено, что при работе с компьютером, пользователь обязан делать перерыв в размере 15 минут каждый час непрерывной работы [15]. «Инструкция по охране труда при работе на ПК» устанавливает следующие требования для работников: к работе с ПК допускаются работники, не имеющие медицинских противопоказаний; женщины в период беременности, кормления грудью к выполнению всех видов работ, с использованием компьютера, не допускаются [16].

Организационные мероприятия при компоновке рабочей зоны

Рабочее место сотрудника обуславливается состоянием помещения, качеством мебели, техники, соблюдением всех санитарно-технических и гигиенических требований и правил, что регулируется 34 главой Трудового

кодекса. Трудовым кодексом также регламентируется организация службы по охране труда, каждое рабочее место подлежит аттестации с целью выявления вредных и опасных факторов. Согласно статье 216.1 Трудового кодекса существуют органы исполнительной власти, которые осуществляют государственную экспертизу труда.

На основе СанПиН 2.2.542-96 установлены следующие требования к рабочему месту: в помещении, где происходит эксплуатация персонального компьютера, должны иметься источники естественного и искусственного освещения; площадь рабочего места для взрослых пользователей не должна быть меньше 6,0 м².

Этим же стандартом регулируются требования к оборудованию, находящемуся на рабочем месте: высота рабочей поверхности стола должна быть в пределах (680 – 800) мм; расположение монитора от глаз пользователя должно составлять (600 – 700) мм; расположение клавиатуры на поверхности стола от края – (100 – 300) мм; высота стула над полом для человека со средним ростом (161 – 170) см должна составлять 420 мм [17].

Безопасность локальных вычислительных сетей

Проводные ЛВС обычно состоят из узлов, соединенных в сеть посредством сетевого коммутатора и сетевых кабелей.

Беспроводные ЛВС используют радиоволны высокой частоты для передачи сетевых пакетов. При использовании ЛВС в пределах физически защищённых областей требуются базовые технические меры и средства контроля и управления. Однако в случае использования беспроводных технологий применении ЛВС в более широкой среде одна только физическая защита не гарантирует достаточный уровень безопасности.

Для обеспечения информационной безопасности были разработаны такие нормативные документы, как: ГОСТ ИСО/МЭК 27001, ГОСТ ИСО/МЭК 13335, ГОСТ ИСО/МЭК 27002, ГОСТ ИСО/МЭК 27033-1, ГОСТ ИСО/МЭК 27006, ГОСТ ИСО/МЭК 27005, ГОСТ ИСО/МЭК 27004, ГОСТ ИСО/МЭК 27003.

Риски проводной ЛВС связаны с:

- Несанкционированным доступом и изменениями, вносимыми в ПК, серверы и другие соединенные с ЛВС устройства;
- Паролями низкого качества;
- Хищением аппаратных средств;
- Импортом вредоносной программы через электронную почту и доступ к веб-страницам;
- Отказом аппаратных средств;
- Несанкционированными соединениями с инфраструктурой ЛВС (коммутаторами, коммутационными стойками);
- Несанкционированными соединениями с выходными устройствами;
- Паролями по умолчанию на портах управления сетевых устройств;
- Сбоем устройств;
- Недостаточной физической защитой и др.

Беспроводные локальные сети подвержены всем уязвимостям проводных ЛВС, а также некоторым уязвимостям, связанным с характеристиками беспроводной сетью.

Основные риски безопасности при использовании беспроводных сетей включают в себя риски с:

- Несанкционированным доступом;
- Взаимными и преднамеренными помехами;
- Неправильной конфигурацией;
- Отключением по умолчанию безопасного режима доступа;
- Небезопасными протоколами шифрования;
- Не всегда существующей возможностью идентификации пользователей беспроводной сети;
- Неисправными устройствами др.

Меры и средства для защиты средств ЛВС могут включать в себя:

- Физические меры и средства контроля и управления безопасностью;
- Строгая политика паролей;
- Требование регистрации на каждом ПК/рабочей станции с использованием, по крайней мере, пары идентификатор пользователя/пароль;
- Отображение времени последней успешной регистрации;
- Запрет повторного отображения имени пользователя в последней успешной регистрации или любого списка ранее использованных имен пользователей;
- Установка программных средств защиты от вредоносной программы (включая антивирусные программные средства) и регулярное автоматическое их обновление и др. [18].

В работе веб-сервера ПЛК в ходе проверок были обнаружены следующие уязвимости:

1. Злоумышленник может удаленно управлять встроенным веб-сервером ПЛК при условии, что пользователь-жертва имеет активный сеанс и в процессе работы осуществил переход по вредоносной ссылке.
2. Злоумышленник может перевести устройство в режим «Ошибка» при отправке на порт 443/tcp(HTTPS) специально созданного пакета. Для восстановления системы необходимо осуществить ее перезапуск.
3. Существует возможность изменения HTTP-заголовков при условии, что пользователь осуществил переход по вредоносной ссылке.

Перечисленные уязвимости были устранены в более новых версиях ПО ПЛК [19].

В качестве дополнительного средства защиты ПЛК можно использовать его только в безопасных сетях, не подключенных к сети Интернет. Другим аппаратным решением может являться использование шлюзов безопасности, которые отделяют сервер от сети Интернет.

Безопасность элементов системы управления

Стандарт ГОСТ Р МЭК 61508 устанавливает общий подход к вопросам обеспечения безопасности для всех стадий жизненного цикла систем, состоящих из электрических и/или электронных, и/или программируемых электронных элементов, используемых для выполнений функций обеспечения безопасности.

Программируемые логические контроллеры относятся программируемым электронным устройствам, т.е. к средствам, основанным на использовании компьютерных технологий, включающим в себя аппаратное и программное обеспечение, а также устройства ввода и/или вывода.

Уровень полноты безопасности – это дискретный уровень, соответствующий диапазону значений полноты безопасности (принимаящий одно из четырех возможных значений). Уровень полноты безопасности, равный 4, является наивысшим уровнем полноты безопасности, а уровень полноты безопасности, равный 1, соответствует наименьшей полноте безопасности. Чем выше уровень полноты безопасности, тем ниже вероятность того, что система, связанная с безопасностью, не сможет выполнить указанные функции безопасности или не будет в состоянии, когда потребуется, принять указанное состояние.

Для определения уровня полноты безопасности для режима работы с низкой частотой запросов рассматриваются значения средней вероятности опасных отказов по запросу функции безопасности, для режима с высокой частотой запросов рассматриваются значения средней вероятности опасных отказов функции безопасности в час. Значения уровней безопасности для низкочастотного режима приведены в таблице 15, а для высокочастотного в таблице 16.

Таблица 15 – Уровни полноты безопасности для низкочастотного режима работы

Уровень полноты безопасности	Средняя вероятность опасного отказа функции безопасности по запросу
4	$> 10^{-5} - < 10^{-4}$
3	$> 10^{-4} - < 10^{-3}$
2	$> 10^{-3} - < 10^{-2}$
1	$> 10^{-2} - < 10^{-1}$

Таблица 16 – Уровни полноты безопасности для высокочастотного режима

Уровень полноты безопасности	Средняя частота опасных отказов функции безопасности
4	$> 10^{-9} - < 10^{-8}$
3	$> 10^{-8} - < 10^{-7}$
2	$> 10^{-7} - < 10^{-6}$
1	$> 10^{-6} - < 10^{-5}$

Отказом считается прекращение способности функционального блока выполнять необходимую функцию, либо функционирование этого блока любым способом, отличным от требуемого. Средняя вероятность отказа на запрос в зависимости от схемы подключения.

Электробезопасность

На любом производстве, где используются системы автоматизации технологических процессов, рабочие ежедневно сталкиваются с различными электроприборами, которые могут нести в себе опасность поражения электрическим током. Действие электрического тока на человека носит многообразный характер. Проходя через организм человека, электрический ток вызывает термическое (ожоги), электролитическое (разложение крови), биологическое (раздражение тканей, нарушение кровообращения и дыхания) и механическое действие (судороги, разрывы кожи, сосудов, переломы костей). Все это многообразие действий может привести к двум видам поражения:

электрическим травмам и электрическим ударам (см. таблицу 17) [20].

Таблица 17 – Воздействие на организм различных значений силы тока

Сила тока, мА	Воздействие
(20 – 25)	Паралич рук, затруднение дыхания
(50 – 80)	Паралич дыхания
(90 – 100)	Фибриляция сердца
>300	Паралич сердца

Решение по обеспечению экологической безопасности

Преимущества автоматизированных систем управления технологическими процессами, в том числе и SCADA-систем, заключаются в том, что они позволяют оперативно устранять или минимизировать последствия любых образовавшихся утечек и других аварийных ситуаций путем дистанционного управления технологическими параметрами, информация о которых поступает моментально в контрольно-диспетчерские пункты. С одной стороны, быстрое реагирование на утечки способствует улучшению экологии, сокращению вредных для человека выбросов в результате различных неисправностей транспортной системы, с другой стороны, система СКАДА позволяет сэкономить значительные объемы энергоресурсов.

Таким образом, SCADA-систем способны обеспечивать работу, соответствующую всем современным нормам экологической безопасности и охраны труда. Поэтому получила широкое распространение на производствах множества отраслей и имеет огромный потенциал дальнейшего внедрения и использования.

Безопасность в чрезвычайных ситуациях

Анализ вероятных ЧС, которые могут возникнуть при исследовании объекта

В данной работе в качестве рабочего места рассматривается место оператора ЭВМ установки системы автоматического управления. Наиболее

типичной ситуацией возникновения чрезвычайной ситуации на рабочем месте является пожар. Причиной пожара могут служить такие факторы как: неисправное электрооборудование, неисправная проводка, не соблюдение техники пожарной безопасности.

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

Возникновение пожара может быть обусловлено такими факторами как: возникновение короткого замыкания в электропроводке, возгорание устройств вычислительной аппаратуры, нарушение правил пожарной безопасности, возгоранием электроприборов и электроустановок при неправильном их использовании.

Основными мерами для предотвращения пожаров являются соблюдения всех стандартов по проектированию и строительству зданий и сооружений, своевременный ремонт и проверка электрооборудования, а также соблюдение правил пожарной безопасности [21].

При обнаружении пожара необходимо выполнить следующие действия:

- Сообщить о пожаре по телефону 01 или 010 с мобильного телефона, где необходимо указать адрес объекта и место возникновения пожара
- В случае задымления помещения необходимо лечь на пол и как можно скорее покинуть помещение.
- Принять все возможные средства по эвакуации людей в соответствии с планом эвакуации здания.
- По возможности отключить электроэнергию и приступить к тушению пожара, не подвергая свою жизнь опасности.

На рабочем месте используются только порошковые огнетушители, предназначенные для ликвидации пожаров в жилых, промышленных, складских и административных помещениях [22].

ЗАКЛЮЧЕНИЕ

Результатом выполнения выпускной квалификационной работы является библиотека функциональных блоков для программной среды MasterSCADA.

В ходе выполнения работы было рассмотрено понятие SCADA-системы, определены основные цели и задачи, которые ставит внедряющая систему компания перед разработчиком, были выделены основные параметры и характеристики для сравнения SCADA-систем, было проведено сравнение нескольких SCADA-систем. из этой части работы был сделан вывод, что модификация графических возможностей SCADA-системы может быть наиболее выгодным решением при её реализации.

Также было рассмотрено и изучено множество понятий, понимание которых необходимо для выполнения задания. А именно: изучена программная среда MasterSCADA и понятие функционального блока в ней, рассмотрен технологический стандарт Component Object Model и программная платформа .NET Framework, освоена среда разработки программного обеспечения Visual Studio, в которой была реализована библиотека функциональных блоков и объектно-ориентированный язык программирования C#, на котором была написана большая часть программного кода библиотеки.

Была проделана работа по оценке необходимого набора компонентов и их свойств для реализации в библиотеке, реализован и зафиксирован алгоритм разработки функциональных блоков. Проведен сравнительный анализ SCADA-систем. Был исследован вопрос возможности использования разработанной библиотеки в других SCADA-системах.

В результате проведения технико-экономического анализа выявлено, что исследуемый метод модификации SCADA-системы позволит значительно снизить затраты на её разработку.

Таким образом, разработанная библиотека функциональных блоков для программной среды MasterSCADA удовлетворяет всем поставленным задачам, имеет высокую гибкость и возможность дальнейшего развития и использования в других SCADA-системах.

CONCLUSION

The result of the work is a library of controls for the MasterSCADA.

In the course of the work, the concept of a SCADA system was considered, the main goals and tasks of implementing SCADA were identified, the main parameters and characteristics for comparing SCADA systems were identified, and several SCADA systems were compared. It was concluded that the modification of the graphical capabilities of the SCADA system might be the most profitable solution for its implementation.

Many concepts necessary for the assignment were reviewed and studied: the MasterSCADA software environment and the idea of a control, the Component Object Model technology standard and the .NET Framework software platform, the Visual Studio software development environment and the object-oriented programming language C#.

The required set of library components and their properties were evaluated, the algorithm for developing functional blocks was implemented and recorded. Some SCADA systems were compared. The possibility of using the developed library in other SCADA-systems was investigated.

As a result, it was revealed that the method of modification of the SCADA system under study would significantly reduce the cost of its development.

Thus, the developed library of functional blocks for the MasterSCADA satisfies all the goals and has high flexibility and the possibility of further development and use in other SCADA systems.

СПИСОК ЛИТЕРАТУРЫ

1. Сайт Industrial Automation Software Solutions by Inductive Automation [Электронный ресурс]. – Режим доступа: <https://inductiveautomation.com> (содержит сведения о SCADA-системах).
2. Сайт Achieve|DE – Empowering Organizations to Lead with Insight [Электронный ресурс]. – Режим доступа: <https://achievede.com/> (содержит сведения о принципах внедрения САУ).
3. Вожаев Д.В. Обзор основных критериев оценки современных SCADA-систем // НЕФТЯНОЕ ХОЗЯЙСТВО. – 2006 г. – №8. – С. 86-87
4. Сайт ИнСАТ - Интеллектуальные Системы Автоматизации Технологии - промышленная автоматизация во всех отраслях [Электронный ресурс]. – Режим доступа: <https://www.insat.su/> (содержит сведения о SCADA-системе MasterSCADA).
5. Сайт Wonderware Russia [Электронный ресурс]. – Режим доступа: <https://www.wonderware.ru/> (содержит сведения о SCADA-системе InTouch).
6. Сайт SCADA TRACE MODE. Российская SCADA система для АСУ ТП [Электронный ресурс]. – Режим доступа: <http://www.adastra.ru/> (содержит сведения о SCADA-системе TRACE MODE).
7. Основы проектирования в MasterSCADA. — М.: Методическое пособие, 2016. —277 с..
8. Component Object Model // Википедия. [2018—2018]. Дата обновления: 06.09.2018. URL: <https://ru.wikipedia.org/?oldid=94918206> (дата обращения: 06.09.2018).
9. .NET Framework // Википедия. [2019—2019]. Дата обновления: 02.05.2019. URL: <https://ru.wikipedia.org/?oldid=99546341> (дата обращения: 02.05.2019).
10. Microsoft Visual Studio // Википедия. [2019—2019]. Дата обновления: 06.05.2019. URL: <https://ru.wikipedia.org/?oldid=99633235> (дата обращения: 06.05.2019).

11. С Sharp // Википедия. [2019—2019]. Дата обновления: 06.05.2019. URL: <https://ru.wikipedia.org/?oldid=99632909> (дата обращения: 06.05.2019).
12. Создание ФБ и ВФБ для MasterSCADA на платформе Microsoft .Net Framework — М.: Методическое пособие, 2014. —26 с..
13. From VBA to .NET The variety of zenon programming interfaces — Амстердам: Ing. Punzenberger COPA-DATA GmbH, 2010 —11 с.
14. Information Unlimited Magazine for Automation Industry— Зальцбург: Ing. Punzenberger COPA-DATA GmbH, 2009 —25 с.
15. СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к ПЭВМ и организации труда.
16. Инструкция по охране труда при работе на персональном компьютере. [Электронный ресурс]. URL: <http://businessforecast.by/partners/646/1379> (дата обращения 15.04.2019).
17. СанПиН 2.2.542-96. Гигиенические требования к видеодисплейным терминалам, персональным электронно-вычислительным машинам и организации работы.
18. ГОСТ Р ИСО/МЭК 27033-1-2011. Методы и средства обеспечения безопасности. Безопасность сетей. Часть 1. Обзор и концепции.
19. ГОСТ Р МЭК 61508-2012. Функциональная безопасность систем электрических, электронных, программируемых электронных, связанных с безопасностью.
20. ГОСТ Р 12.1.019-2009 Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты.
21. ГОСТ 12.1.004-91 «Пожарная безопасность. Общие требования»
22. ГОСТ Р 51057-2001 – «Техника пожарная. Огнетушители переносные. Общие технические требования. Методы испытаний»

ПРИЛОЖЕНИЕ А

(обязательное)

ЛИСТИНГ КОДА ФУНКЦИОНАЛЬНОГО БЛОКА

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Drawing;
using System.Data;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Runtime.InteropServices;
using FB.VisualBasic;
namespace MySCADALibrary
{
    [ComVisible(true),
     Guid("FC93644F-A4ED-43F5-B9D6-2A4D73BB07CD")]
    public partial class ButtonControl : VisualControlBase
    {
        public ButtonControl()
        {
            InitializeComponent();
        }
        bool mouseButtonPressed;
        bool buttonRun;
        // Свойства отображаемые для контролла
        bool _mode = false;
        [DispId(1),
         DisplayName("Обработка нажатия"),
```

```

Description("FALSE - реверс, TRUE - моментальное"),
Category("Поведение")]
public bool Mode
{
    get { return _mode; }
    set { _mode = value; }
}
bool _externalRun = false;
[DispId(2),
DisplayName("Внешний источник состояния"),
Description("Используется внешний источник состояния кнопки?"),
Category("Внешний вид")]
public bool ExternalRun
{
    get { return _externalRun; }
    set { _externalRun = value; }
}
Color _colorTrue = Color.IndianRed;
[DispId(3),
DisplayName("Цвет TRUE"),
Description("Цвет кнопки с состоянием TRUE"),
Category("Внешний вид")]
public Color ColorTrue
{
    get { return _colorTrue; }
    set { _colorTrue = value; }
}
Color _colorFalse = Color.YellowGreen;
[DispId(4),
DisplayName("Цвет FALSE"),

```

```

Description("Цвет кнопки с состоянием FALSE"),
Category("Внешний вид")]
public Color ColorFalse
{
    get { return _colorFalse; }
    set { _colorFalse = value; }
}
String _textTrue = "Stop";
[DispId(5),
DisplayName("Текст TRUE"),
Description("Текст кнопки с состоянием TRUE"),
Category("Внешний вид")]
public string TextTrue
{
    get { return _textTrue; }
    set { _textTrue = value; }
}
String _textFalse = "Go";
[DispId(6),
DisplayName("Текст FALSE"),
Description("Текст кнопки с состоянием False"),
Category("Внешний вид")]
public string TextFalse
{
    get { return _textFalse; }
    set { _textFalse = value; }
}
Font _textFont = new Font("Calibri Light", 28);
[DispId(7),
DisplayName("Шрифт"),

```

```

        Description("Параметры шрифта"),
        Category("Внешний вид")]
public Font TextFont
{
    get { return _textFont; }
    set { _textFont = value; }
}

private void ButtonControl_Load(object sender, EventArgs e)
{
}

private void Button1_Click(object sender, EventArgs e)
{
    if (!_mode)
    {
        buttonRun = !buttonRun;
        FBConnector.SetPinValue(myButton.VisuOUT,
!FBConnector.GetPinBool(myButton.VisuOUT));
    }
}

private void Button1_MouseDown(object sender, MouseEventArgs e)
{
    if (e.Button == System.Windows.Forms.MouseButtons.Left)
mouseButtonPressed = true;
}

private void Button1_MouseUp(object sender, MouseEventArgs e)
{
    if (e.Button == System.Windows.Forms.MouseButtons.Left)
mouseButtonPressed = false;
}

private void Timer1_Tick(object sender, EventArgs e)

```

```

    {
        if (!FBConnector.DesignMode) return;
        if((_externalRun && FBConnector.GetPinBool(myButton.VisuRUN))
|| (!_externalRun && buttonRun))
        {
            button1.BackColor = _colorTrue;
            button1.FlatAppearance.MouseOverBackColor =
Color.FromArgb(_colorTrue.A - 22, _colorTrue);
            button1.FlatAppearance.MouseDownBackColor =
Color.FromArgb(_colorTrue.A - 44, _colorTrue);
            button1.Text = TextTrue;
        }
        else
        {
            button1.BackColor = _colorFalse;
            button1.FlatAppearance.MouseOverBackColor =
Color.FromArgb(_colorTrue.A - 22, _colorFalse);
            button1.FlatAppearance.MouseDownBackColor =
Color.FromArgb(_colorTrue.A - 44, _colorFalse);
            button1.Text = TextFalse;
        }
        If (FBConnector.GetPinBool(myButton.VisuBLOCK))
button1.BackColor = Color.Silver;
        button1.Enabled=!FBConnector.GetPinBool(myButton.VisuBLOCK);
        if (_mode)
        {
            FBConnector.SetValue<bool>(myButton.VisuOUT,mouseButtonPressed);
            buttonRun = mouseButtonPressed;
        }
    }
}
}
}
}

```