

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

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Автоматизированный склад запасного имущества и принадлежностей

УДК 658.78.011.5

Студент

Группа	ФИО	Подпись	Дата
8Т7А	Кудинов Михаил Сергеевич		

Руководитель ВКР

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

Консультант

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старш. Преподаватель	Тутов И.А.	-		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Маланина В.А.	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Черемискина М.С.	-		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАР ИШИТР	Громаков Е.И.	К.Т.Н.		

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

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 осведомлённость 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 работу коллектива при решении инновационных инженерных задач в области автоматизации и управления техническими объектами, демонстрировать при этом готовность следовать профессиональной этике и нормам.
P10	Иметь широкую эрудицию, в том числе знание и понимание современных общественных и политических проблем, вопросов безопасности и охраны здоровья сотрудников, юридических аспектов, ответственности за инженерную деятельность, влияния инженерных решений на социальный контекст и окружающую среду.
P11	Понимать необходимость и уметь самостоятельно учиться, и повышать квалификацию в течение всего периода профессиональной деятельности.

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

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

Отделение школы (НОЦ) – Отделение автоматизации и робототехники

Уровень образования – бакалавриат

Период выполнения – осенний / весенний семестр 2020/2021 учебного года

Форма представления работы:

бакалаврская работа

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

Срок сдачи студентом выполненной работы:	03.06.2021
------------------------------------------	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
25.04.2021	Основная часть	60
04.05.2021	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	20
13.05.2021	Социальная ответственность	20

СОСТАВИЛ:

Руководитель ВКР

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

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 школы (НОЦ) – Отделение автоматизации и робототехники

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

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

бакалаврской работы

Студенту:

Группа	ФИО
8Т7А	Кудинов Михаил Сергеевич

Тема работы:

Автоматизированный склад запасного имущества и принадлежностей	
Утверждена приказом директора (дата, номер)	№50-15/с от 19.02.2021

Срок сдачи студентом выполненной работы:	03.06.2021
------------------------------------------	------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	Объектом исследования является виртуальная модель автоматизированной системы управления складом. <ul style="list-style-type: none"> • Программное обеспечение
---------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<p>Перечень подлежащих исследованию, проектированию и разработке вопросов</p>	<ul style="list-style-type: none"> • Разработка виртуальной модели склада в программе Factory I/O; • Разработка SCADA в программе Simple-Scada и связь с виртуальной моделью; • Подбор аппаратной части системы • Разработка структурной схемы автоматизации • Раздел «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»; • Раздел «Социальная ответственность».
<p>Перечень графического материала</p>	<p>Презентация в формате *.pptx</p>

<p>Консультанты по разделам выпускной квалификационной работы</p>	
<p>Раздел</p>	<p>Консультант</p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Маланина Вероника Анатольевна, доцент ОСГН ШБИП, к.э.н.</p>
<p>Социальная ответственность</p>	<p>Черемискина Мария Сергеевна, ассистент ООД ШБИП</p>
<p>Названия разделов, которые должны быть написаны на русском и иностранном языках:</p>	
<p> </p>	

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

Задание выдал руководитель / консультант (при наличии):

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

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
<p>8Т7А</p>	<p>Кудинов Михаил Сергеевич</p>	<p> </p>	<p> </p>

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

Группа	ФИО
8Т7А	Кудинову Михаилу Сергеевичу

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

Тема ВКР:

Автоматизированный склад запасного имущества и принадлежностей	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Разработка динамической визуализированной модели склада и его интеграция с виртуальным программно-техническим комплексом управления.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p>Правовые и организационные вопросы обеспечения безопасности:</p> <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<p>Трудовой кодекс Российской Федерации" от 30.12.2001 N 197-ФЗ (ред. от 31.12.2014) ГОСТ 12.2.032-78. Система стандартов безопасности труда. ГОСТ 12.2.032-79 «СССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования». ГОСТ 12.2049-80 «ССБТ. Оборудование производственное». ГОСТ 22269-76. Система «человек-машина». Рабочее место оператора. Взаимное расположение элементов рабочего места. Общие эргономические требования. ГОСТ Р 50923-96. Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения.</p>
<p>2. Производственная безопасность:</p> <p>Анализ выявленных вредных и опасных факторов</p> <p>Обоснование мероприятий по снижению воздействия</p>	<ol style="list-style-type: none"> 1. Недостаточная освещенность рабочей зоны. 2. Повышенный уровень шума. 3. Отклонение показателей микроклимата. 4. Электромагнитное излучение. 5. Электробезопасность.

3. Экологическая безопасность:	Анализ негативного воздействия на окружающую среду работы системы автоматизированного склада: <ul style="list-style-type: none"> • Атмосфера: потребление энергии; • Гидросфера: потребление электроэнергии; • Литосфера: образование отходов;
4. Безопасность в чрезвычайных ситуациях:	Возможные ЧС при эксплуатации автоматизированной системы управления складом: - пожар - короткое замыкание Типичная ЧС: - Получение физической травмы от случайно упавших объектов на складе

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

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Черемискина Мария Сергеевна	-		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Т7А	Кудинов Михаил Сергеевич		

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

Студенту:

Группа	ФИО
8Т7А	Кудинов Михаил Сергеевич

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. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Оклад консультанта - 15000 руб. Оклад инженера – 15000 руб.
2. <i>Нормы и нормативы расходования ресурсов</i>	Накладные расходы 10%; Районный коэффициент 30%
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Коэффициент отчислений на уплату внебюджетные фонды 30,2 %
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. <i>Оценка коммерческого и инновационного потенциала НТИ</i>	Выявление потенциальных потребителей, анализ конкурентных разработок, SWOT-анализ.
2. <i>Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок</i>	Планирование работ, построение диаграммы Ганта, планирование бюджета НТИ.
3. <i>Определение ресурсной, финансовой, экономической эффективности</i>	Оценка сравнительной эффективности исследования: Интегральный показатель коэффициент ресурсоэффективности – 4,2.
Перечень графического материала (с точным указанием обязательных чертежей):	
1. Карта сегментирования рынка 2. Оценка конкурентоспособности технических решений 3. Матрица SWOT 4. График проведения и бюджет НТИ 5. Диаграмма Ганта 6. Оценка ресурсной, финансовой и экономической эффективности НТИ	

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

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Маланина В.А.	Доцент, к.э.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8Г7А	Кудинов Михаил Сергеевич		

РЕФЕРАТ

Выпускная квалификационная работа 91 с., 13 рис., 18 табл., 2 листинга кода, 34 источника, 5 приложений.

Ключевые слова: автоматизированная система управления, склад, SCADA, человеко-машинный интерфейс, Raspberry Pi, C#, виртуальная динамическая модель.

Объектом исследования является автоматизированная система управления складом.

Цель работы – разработка наглядной модели автоматизированной системы управления складом, которую при наличии необходимых средств можно внедрить в различные логистические компании.

Проведён анализ существующих логистических АСУ ТП. Для управления объектом был использован программный комплекс Simple-Scada. Для реализации движения и взаимодействия элементов виртуальной модели были использованы возможности языка программирования C#.

Области применения – предприятия в сфере логистики, доставки грузов и компании с крупными складскими помещениями.

ОГЛАВЛЕНИЕ

Введение.....	13
1. Обзор рынка автоматизированных систем управления складом.....	15
1.1 Коробочные системы.....	16
1.2 Адаптируемые системы.....	16
1.3 Заказные системы.....	16
1.4 Сравнительная характеристика основных компаний WMS систем.....	17
2. Требования и цели разработки системы.....	20
2.1 Назначение и цели разработки системы.....	20
2.2 Требования к системе управления.....	20
3. Объект и методы исследования.....	21
3.1 Виртуальная модель складского помещения в Factory IO.....	21
3.2 Разработка человеко-машинного интерфейса в программе Simple-Scada.....	22
3.3 Программный комплект разработки C# .NET для взаимодействия с Factory I/O.....	25
4. Проектирование и разработка.....	27
4.1 Описание технологического процесса.....	27
4.2 Разработка структурной схемы автоматизации.....	28
4.3 Выбор аппаратно-технического обеспечения в сборки системы.....	29
4.3.1 Сравнительный анализ технологий считывания.....	29
4.3.2 Выбор RFID датчика.....	30
4.3.3 Выбор обрабатывающего устройства.....	32
4.4 Разработка алгоритма управления сортировкой грузов.....	33
4.5 Разработка программного обеспечения.....	34
4.5.1 Разработка модели в программе Factory I/O.....	34
4.5.2 Разработка человеко-машинного интерфейса в программе Simple-Scada.....	39
5. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение..	43
5.1 Предпроектный анализ.....	43
5.1.1 Потенциальные потребители результатов исследования.....	43
5.2 Анализ конкурентных технических решений.....	44
5.3 SWOT-анализ.....	46

5.4 Планирование научно-исследовательской работы.....	48
5.4.1 Структура работ в рамках научного исследования	48
5.4.2 Разработка графика проведения научно-технического исследования	49
5.5 Бюджет научно-технического исследования	53
5.5.1 Расчёт материальных затрат	53
5.5.2 Расчет амортизационных отчислений.....	53
5.5.3 Расчёт заработной платы и отчислений во внебюджетные фонды	54
5.5.4 Накладные расходы	56
5.5.5 Формирование бюджета научно-технического исследования	57
5.6 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования ..	58
Заключение по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение».....	59
6. Социальная ответственность.....	60
6.1 Правовые вопросы обеспечения безопасности.....	61
6.2 Эргономические требования.....	61
6.3 Производственная безопасность.....	63
6.3.1 Анализ вредных факторов.....	64
6.3.2 Анализ опасных факторов.....	70
Заключение по разделу «Социальная ответственность»	73
Заключение	75
Список литературы	76
Приложение А	78
Приложение Б	79
Приложение В.....	80
Приложение В.....	82
Приложение Г	84
Приложение Д.....	89

Введение

Находясь в условиях жесткой конкуренции, связанной с повсеместным внедрением автоматизированных и автоматических систем, производственные предприятия и научно-исследовательские институты заинтересованы в совершенствовании существующих и производстве новых АСУ ТП для предприятий промышленности. В то же время промышленность с каждым годом повышает эффективность производства из-за конкуренции в своих сферах. Таким образом рынок на данный момент находит в высокой точке развития из-за регулярного создания новых узлов и систем автоматизации.

Автоматизированные системы управления позволяют не только обезопасить людей, работающих непосредственно на промысле, от влияния вредных факторов, но и улучшить качество получаемого продукта и наладить бесперебойную работу комплексов установок.

Внедрение на производстве автоматизированных систем управления позволяет решить множество задач, приводящих к замещению человеческого труда аппаратным. Вследствие чего уменьшается численность рабочих, занятых непосредственно на производстве, увеличивается производительность труда и повышается его эффективность, уменьшается влияние человеческого фактора на технологический процесс.

Цель работы – разработка наглядной модели автоматизированной системы управления складом, которую при наличии необходимых средств можно внедрить в различные логистические компании. Для завершения работы использовались технические и программные средства платформы визуального обучения и языка программирования C#, математический аппарат, платформа построения алгоритмов Visio, ПО Factory I/O и программное обеспечение Simple-Scada для проектирования автоматизированных систем управления SCADA.

Помимо этого, целью является получение навыков по проектированию структурных схем и технической документации в программном комплексе

Компас 3D, который является одним из самых популярных приложений в данной сфере, и использованию справочной литературы и интернет-ресурсов для поиска необходимых решений. С помощью данной программы были разработаны документы функциональная и структурная схемы.

1. Обзор рынка автоматизированных систем управления складом

Система управления складом WMS — это программа, которая контролирует все операции, выполняемые при хранении. Аббревиатура WMS произошла от английского названия «Warehouse Management System».

Автоматизированная система управления складом включает в себя следующие компоненты:

1. Приложение – это человеко-машинный интерфейс оператора АСУ ТП, с которым происходит взаимодействие при внесении, изменения, удалении данных, реализации операций и формировании отчетов. Обычно это программное обеспечение устанавливается на персональный компьютер, планшет или смартфон.
2. Рабочий сервер — это комплекс программного и аппаратного обеспечения, отвечающий за хранение, обработку, расшифровку и любые другие операции с данными, полученными от клиентского приложения.
3. Бизнес-логика – обрабатывает запросы, генерирует отчеты, а также изменяет, добавляет и удаляет данные из общей базы данных. Информировать пользователя о выполнении операции посредством сообщения на экране клиентского приложения.

Когда товары перевозятся внутри компании, система создает карту с оптимальными маршрутами, которые позволяют максимально продуктивно использовать погрузочную технику и свести к минимуму холостые поездки. Если на складе имеется несколько типов погрузчиков, подбирается оптимальный вариант для каждой ситуации. После выполнения задания сканируется RFID метка, это исключает вероятность неправильной работы с грузом, новый код сразу добавляется в базу данных.

Существуют различные виды автоматизированных систем управления складом:

1. Коробочные системы
2. Адаптируемые

3. Заказные

1.1 Коробочные системы

Системы в штучной упаковке готовы к установке программных продуктов, которые предлагают несколько вариантов настройки и бизнес-требований. Тем не менее это должно означать, что эти системы работают хуже, чем другие модели АСУ ТП.

1.2 Адаптируемые системы

Адаптивные системы — это известная на практике группа WMS систем на международном рынке. Данные системы пользуются спросом в ситуации, когда ни одна коробочная система не способна выполнить необходимый набор функций, но список необходимых задач недостаточно обширный, чтобы разрабатывать заказную систему. Среди компаний, которые предоставляют подобные решения: HighJump Software, PSI Logistics (ПСИ) и многие другие.

Адаптивные системы обычно разрабатываются под Unix операционные системы, но также не мало примеров АСУ, которые работают под Windows, также данные системы используют систему управления базами данных. Стоимость разработки и установки такого рода систем варьируется от места установки и количества активных пользователей.

1.3 Заказные системы

Частные системы АУ — это множество складских зданий и комплекс различных видов деятельности. Основная функция индивидуального решения заключается во включении всех необходимых функций, требующихся заказчику.

Уникальной особенностью такого рода систем заключается в том, что такие решения не пишутся с нуля, они используют некоторый набор базовых функций, после их настройки происходит разработка и подключение необычных звеньев системы, которые требуются заказчику.

Эти системы обычно работают под семейством Unix систем. Популярные системы управления базами данных используются в качестве обработчика таблиц базы данных.

Сроки внедрения данной системы часто превышают 1-2 года, стоимость проекта оценивается в миллионы долларов, количество пользователей системы более 50 человек.

1.4 Сравнительная характеристика основных компаний WMS систем

Таблица 1.1 – Сравнительная характеристика АСУ складом

Название организации	Предоставляемые возможности
Blue Yonder (formerly JDA)	<ul style="list-style-type: none"> • Обеспечивает высокую прозрачность и точное ежедневное планирование в интегрированной сети поставок и продаж; • Поддерживает разные рабочие процессы и разные производственные среды; • Возможность составления отчетов по исключениям; • Обеспечивает моделирование «что, если»; • Предлагает отраслевые модели, чтобы помочь клиентам выбрать наиболее экономичное решение; • Поддерживает бизнес-правила для капиталоемких отраслей;
Infor	<ul style="list-style-type: none"> • Предлагает полную и надежную функциональность всех процессов промышленного предприятия различного типа производства; • Позволяет быстро и точно планировать загрузку оборудования, затраты на рабочую силу и потребности в сырье для производства продукции; • Позволяет отслеживать производство в соответствии с рабочими заданиями, производственными планами или этикетками KANBAN; • Обеспечивает быструю обработку заказов клиентов и гарантирует, что предложения клиентам реалистичны и осуществимы;
Körber	<ul style="list-style-type: none"> • Внесение изменений в решение в любое время; • Поддерживается конфигурация системы заказчика в полном объеме, в том числе самостоятельно выполненные разработки; • Последующие модификации системы и переход на новые версии позволяют заказчику сохранить всю проделанную работу; • Наличие сертифицированного встроенного модуля Advantage Link, который интегрирует все приложения Körber с основными мировыми бизнес-системами.

<p>Manhattan Associates</p>	<ul style="list-style-type: none"> • рассчитывать эффективность персонала; • использовать на складе новейшие технологии: системы голосового управления, RFID, интеграцию оборудования; • оптимизировать размещение товаров на складе; • управлять задачами сотрудников и контролировать загруженность складского персонала; • контролировать складские операции; • упростить документооборот, повысить эффективность взаимодействия с контрагентами; • выполнять все операции с помощью радиотерминалов.
-----------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Квадрант распределения компаний на рынке систем управления складом в мире представлен на рисунке 1.1. Квадрант правильно рассматривать необходимо исходя из следующих входных параметров:

1. «Лидеры» - поставщики с положительными рейтингами по обеим осям;
2. «Кандидаты» - поставщики с положительными оценками только за «возможность реализации»;
3. «Провидцы» - поставщики с положительной оценкой только за «полноту видения».
4. «Нишевые игроки» - поставщики с отрицательными рейтингами по обеим осям.



Рисунок 1.1 - Квадрант распределения компаний на рынке складских систем

2. Требования и цели разработки системы

2.1 Назначение и цели разработки системы

Разрабатываемая система управления имеет перспективу применения в средних и крупных логистических компаниях, занимающихся получением, отгрузкой товаров производственного назначения.

Объектом автоматизации являются процессы, происходящие в складских логистических помещениях: приемка, отгрузка, сортировка грузов. Работа со складом должна проводиться только посредством рабочего места оператора – персонального компьютера, планшета, смартфона.

2.2 Требования к системе управления

Функциональность АСУ ТП должна включать:

1. Автоматизацию процессов приемки грузов посредством считывания установленной RFID метки;
2. Автоматизацию бизнес-процессов размещения грузов на складе на основе различных алгоритмов сортировки;
3. Одновременная приемка товара от поставщика и отгрузка заказчику;
4. Группировка грузов по типу;
5. Возможность работы с грузом посредством таблицы в формате *.xlsx (*.csv) и человеко-машинного интерфейса;
6. Формирование расписания отгрузок в соответствии с их приоритетностью;
7. Автоматизация управления складскими запасами;
8. Автоматизация управления складскими мощностями: конвейерами, погрузчиками, подъемными механизмами;
9. Визуализация процессов происходящих на складе
10. Выдача оповещений о предупреждениях, авариях и ошибках в системе

3. Объект и методы исследования

3.1 Виртуальная модель складского помещения в Factory IO

Factory I/O – это 3D-симуляция фабрики для обучения технологиям автоматизации разработанная компанией Realgames. Платформа разработана таким образом, чтобы быть простой в использовании, она позволяет быстро построить виртуальную фабрику, используя набор обычных промышленных деталей. Factory I/O также включает в себя множество сцен, вдохновленных типичными промышленными приложениями, начиная от начального и заканчивая продвинутым уровнем сложности.

Наиболее распространенным сценарием является использование Factory I/O в качестве обучающей платформы ПЛК, поскольку ПЛК являются наиболее распространенными контроллерами, встречающимися в промышленных приложениях. Однако он также может использоваться с микроконтроллерами, SoftPLC, Modbus и многими другими технологиями.

Платформа Factory I/O в данном исследовательском проекте предоставляет обширные возможности, как сделать и протестировать автоматизированную систему управления без сборки аппаратной части. Работа с данной программой состоит из движения по сцене, которая изначально является пустым помещением, пример представлен на рисунке 3.1. Движения по сцене, как и сборка моделей, подключение контроллера является очень простой операцией, что оказало существенную роль при выборе данного программного обеспечения.

Так же гибкость данной платформы предоставляет возможность не создавать виртуальную модель самостоятельно, используя какие бы то ни было фреймворки или языки программирования, а собирать необходимую модель из представленных в программе блоков.

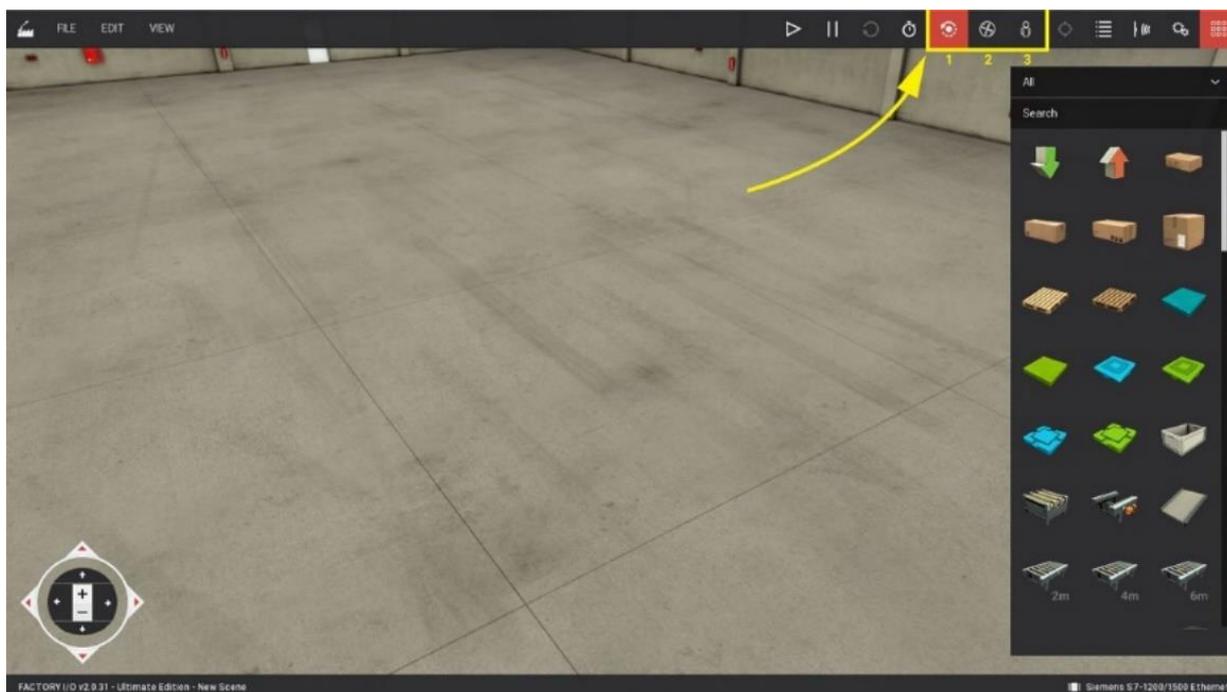


Рисунок 3.1. Начало работы в программе Factory I/O

3.2 Разработка человеко-машинного интерфейса в программе Simple-Scada

Simple-Scada – предназначена для разработки и обеспечения работы в реальном времени систем сбора, обработки, отображения и архивирования информации об объекте мониторинга или управления. Главная цель проекта – это простота и удобство для конечного пользователя.

В простейшей системе АСУ ТП каждый компьютер, по сути, является сервером, так как он читает и записывает значения тегов на устройства, работает с БД и т.д. Использование такой системы ведет к многократному увеличению нагрузки на сеть и контроллеры, удорожанию развертывания подобных систем вследствие высокой стоимости серверных версий SCADA-пакетов, а также к усложнению дальнейшей доработки проекта SCADA-системы. Пример схемы такой системы представлен на рисунке 3.2.

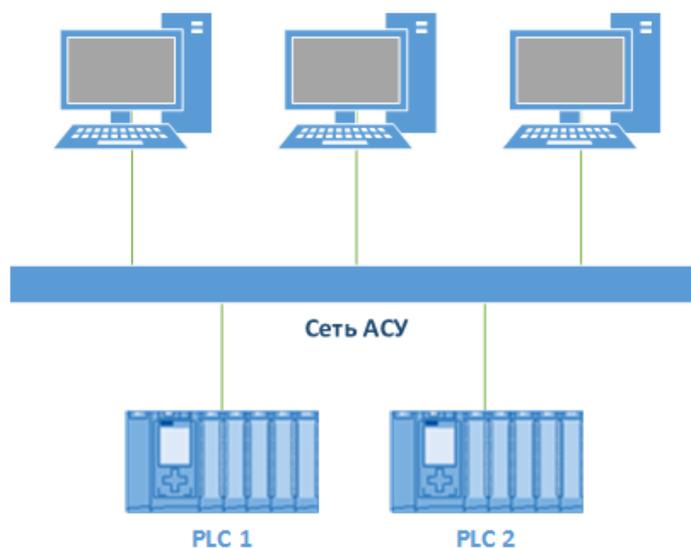


Рисунок 3.2 - Схема простейшей АСУ ТП с высокой потенциальной нагрузкой

Используемая в данной работе программа Simple-Scada позволяет повысить эффективность и скорость работы всей системы АСУТП, используя клиент-серверной архитектуру.

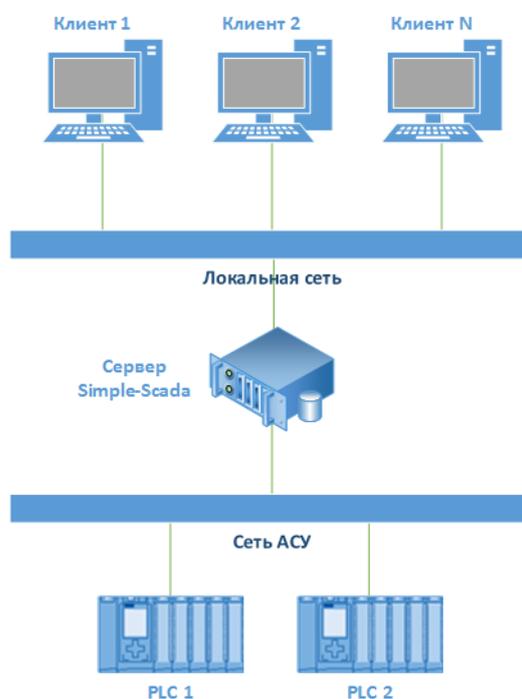


Рисунок 3.3 Клиент – серверная архитектура Simple-Scada

В такой системе сервер Simple-Scada берет на себя основные задачи:

1. подключается к OPC-серверам.
2. опрос контроллеров, читает и записывает значения тегов.
3. выполняет скрипты.

4. работает с базами данных.
5. работает с клиентами.

Клиенты Simple-Scada выполняют только задачи визуализации, все необходимые данные запрашиваются у сервера. Таким образом, при использовании клиент-серверной архитектуры достигаются следующие цели:

1. существенно снижается нагрузка на сеть и контроллеры.
2. снижается общая стоимость системы, так как достаточно приобрести любую версию Simple-Scada и необходимое количество клиентов.
3. упрощается процедура сопровождения проекта SCADA-системы. При внесении изменений в проект его не требуется переносить на клиенты вручную. Необходимо просто перезапустить проект на сервере - клиенты автоматически получают обновленный проект.

Если производственный процесс является критически важным, а стоимость остановки производства очень высока, возникает необходимость построения резервируемых систем.

Система с резервным сервером хранения данных позволяет в случае любой аварии сохранить все данные в изолированном месте, что потенциально сильно снижает стоимость ремонтных работ и устранение неполадок в системе, а также поломка не выводит и строя всю систему и позволяет ей работать в штатном режиме до ремонта основного сервера.

При разработке проектов в Simple-Scada используется технология OPC сервера. OPC (OLE for Process Control) – семейство программных технологий, предоставляющих единый интерфейс для управления объектами автоматизации и технологическими процессами.

Стандарт OPC разрабатывался с целью сократить затраты на создание и сопровождение приложений промышленной автоматизации. Суть OPC проста — предоставить разработчикам промышленных программ универсальный фиксированный интерфейс обмена данными с любыми устройствами.

Технология OPC уже стала мировым стандартом в области

автоматизации технологических процессов. С каждым годом все большее количество оборудования поставляется с OPC-серверами. Сама технология OPC находится в процессе постоянного совершенствования и оптимизации, новые версии стандарта расширяют сферу применения продуктов, поддерживающих спецификации OPC.

Технология OPC реализуется по принципу "клиент – сервер":

1. **OPC-сервер** – программа, получающая данные во внутреннем формате устройства или системы и преобразующая эти данные в формат OPC. OPC-сервер является источником данных для OPC-клиентов. По своей сути OPC-сервер – это некий универсальный драйвер физического оборудования, обеспечивающий взаимодействие с любым OPC-клиентом.
2. **OPC-клиент** – программа, принимающая от OPC-серверов данные в формате OPC.

Simple-Scada подпадает под категорию приложения "OPC-клиент" с обменом данными по спецификации OPC DA. В этом заключается большое преимущество - Simple-Scada может работать с любыми OPC-серверами.

3.3 Программный комплект разработки C# .NET для взаимодействия с Factory I/O

Программное обеспечение Factory I/O SDK предоставляет инструменты, позволяющие разрабатывать пользовательские приложения, получающие доступ к точкам симуляции ввода-вывода виртуальной модели. Эти приложения могут использоваться в качестве интерфейсов между заводским вводом-выводом и практически любыми технологиями, такими как ПЛК, микроконтроллеры, базы данных, электронные таблицы и так далее.

Все точки ввода-вывода симуляции могут быть доступны через SDK независимо от выбранного драйвера.

По своей сути SDK является программной библиотекой, написанной на языке программирования C#. В следствие этого фактора взаимодействие с ней желательно посредством этого языка программирования. Существование

такого удобного инструмента позволяет расширить перечень возможных программ, разработанных в среде Factory I/O, так как становится возможным взаимодействие программы не только с кодом для контроллеров, но и внешним исходным кодом, написанным под операционную систему Windows и это является очень важным в данном проекте.

4. Проектирование и разработка

4.1 Описание технологического процесса

Автоматизированная система управления складом предназначена для повышения эффективности работы логистической компании, уменьшения влияния человеческого фактора, ускорения работы и уменьшения материальных затрат.

Данная WMS система состоит из нескольких технологических узлов:

- RFID считыватель
- Роликовый конвейер
- Поворотный механизм
- Подъемный кран

Алгоритм работы установки представлен рисунке 4.1. Основными процессами в работе системы являются: определение конечной точки расположения груза и формирование правильного расположения груза на основе времени отгрузки. Таким образом груз, который должен быть отгружен раньше других, должен быть расположен территориально ближе к месту отгрузки.

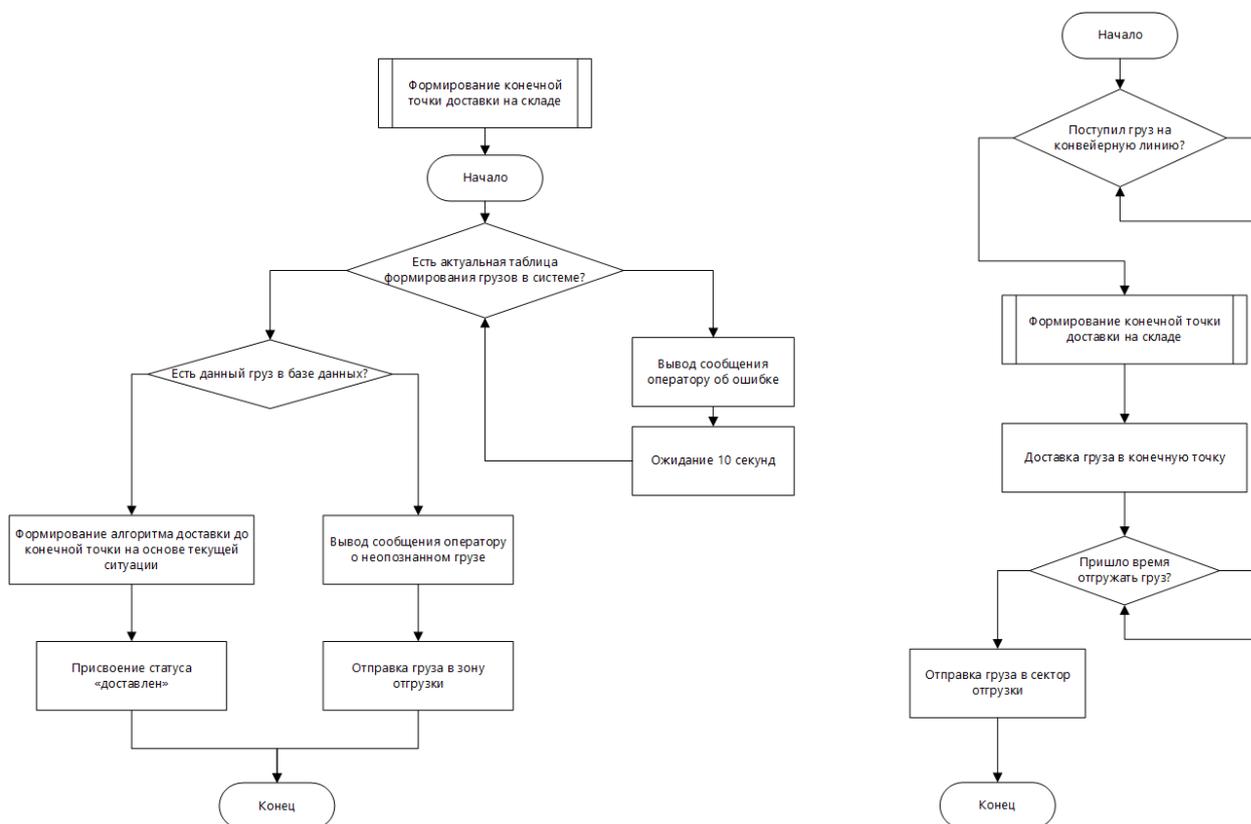


Рисунок 4.1 – Алгоритм работы АСУ ТП

Система производит автоматическое регулирование следующих параметров:

- Номер ячейки точки назначения;
- Траектория груза до точки назначения;
- Определение RFID метки груза;
- Количество заполненных ячеек склада;
- Количество товаров одной категории;

Система контролирует следующие параметры:

- Количество заполненных ячеек склада;
- Количество товаров одной категории;
- Количество свободных ячеек склада;
- Процент заполненных ячеек;

Основными задачами АСУ ТП склада являются:

- Контроль равномерной заполняемости стеллажей
- Формирование оптимального алгоритма погрузки/отгрузки грузов
- Преимущественное использование датчиков и установок отечественного производства

4.2 Разработка структурной схемы автоматизации

Проектирование автоматизированной системы выполнено по принципу трехуровневой иерархии. Структурная схема системы приведена в приложении А.

На полевом уровне данной схемы располагаются местные показывающие приборы и первичные СИ, исполнительные механизмы и устройства местного управления. Для контроля технологических параметров предусматривается использование средств КИПиА.

Средний уровень предусматривает сбор и первичную обработку информации с устройств полевого уровня, контроль обозначенных параметров, передачу (прием) данных на средний уровень (со среднего уровня). На основе информации, поступившей на средний уровень,

формируются команды управления (автоматически или оператором). В данной проектируемой системе средний уровень представлен распределённой системой управления в составе двух шкафов (ввода-вывода и управления). Оба шкафа построены на базе ПЛК.

На верхнем уровне происходит сосредоточение, обработка и упорядочивание (формирование базы данных) информации с нижних уровней. Также предусматривается индикация необходимых параметров, регистрация и хранение информации. Здесь происходит формирование отчетной документации и осуществление управления технологическими режимами системы.

4.3 Выбор аппаратно-технического обеспечения в сборки системы

4.3.1 Сравнительный анализ технологий считывания

NFC использует ту же концепцию, что и технология RFID. Это ветвь высокочастотной RFID, работающей на частоте около 13,56 МГц.

Однако есть несколько небольших отличий, как показано ниже:

Различия в использовании

Основное применение RFID — это отслеживание активов, управление запасами, контроль доступа, отслеживание посетителей и отслеживание инструментов.

Напротив, технология NFC ориентирована на безопасную передачу данных. Этот метод используется в бесконтактных платежах, обмене данными и смарт-плакатах.

Инфраструктурные различия

При использовании RFID вам необходимо иметь RFID-метку и считыватель. И наоборот, устройство NFC может действовать как считыватель и как метка. Эта уникальная функция позволяет осуществлять P2P-обмен данными между двумя устройствами с поддержкой NFC.

Вариации сложности приложения

RFID использует простую концепцию, при которой считыватель извлекает информацию из тега. Считыватель должен обнаруживать

радиоволны меток для захвата данных.

NFC - более сложная технология, которая позволяет выполнять операции чтения / записи. Эта функция превращает ваш телефон в цифровой, обеспечивая беспрепятственное общение между пользователями.

На основании вышеизложенного RFID – это идеальный выбор, если вы хотите отслеживать свои активы или управлять своими запасами. RFID-метки отправляют сигналы считывающему устройству, так что вы можете получить информацию о своем целевом объекте за считанные минуты.

NFC - отличная технология, когда вам нужна сложная система, которая обеспечивает одноранговый обмен, чтение карт и интеллектуальное чтение плакатов. Однако эта технология работает на небольших расстояниях, и устройства должны быть поблизости, чтобы она работала эффективно.

4.3.2 Выбор RFID датчика

RFID датчик – это система связи, которая расшифровывается как метод радиочастотной идентификации. Это метод, задача которого распознать живые или неодушевленные предметы с помощью использования радиоволн. В качестве Auto-ID используются отпечатки пальцев или сетчатка глаза, голос, одежда.

Данная технология позволяет автоматизировать распознавание и учет любых объектов. Процесс простой – данные из RFID-метки отправляются к считывающему устройству, в результате, в программе отражаются изменения. Чтобы все успешно функционировало, требуется антенна для улавливания сигналов от передатчика и правильная установка всего оборудования. Если сравнивать технологию с NFC, то у RFID есть преимущество. Она ловит сигнал на сравнительно большем расстоянии.

При помощи частотной радио идентификации можно автоматизировать и упростить самые разные процессы. Для этого создаются программы:

- по контролю наличию сотрудников на рабочих местах и времени их прихода/ухода;

- определение автомобиля;
- склады;
- производственные операции;
- определение личности человека для входа в программы или комнаты.

Существует несколько категорий меток:

- Активные – питаются от личного источника. Распространяются даже на больших расстояниях, часто включены дополнительные функции. Относительно значительные размеры. Незаменимы везде, где необходимо функционировать даже на большом расстоянии.
- Полупассивные – есть батарея и половина возможностей предыдущей разновидности.
- Пассивные – не имеют собственного источника питания. Работают только в пределах 8 метров от ридера, но компактные и стоят дешевле.

В качестве потенциально подходящих датчиков были подобраны несколько наименований: Zebra FX9600, Nordic ID Sampo S1, Chainway UR4 UHF. Они имеют схожие характеристики, но существенно отличаются по цене. Рассмотрим их характеристики, чтобы понять какой именно нам подходит.

Таблица 4.1 – Сравнительная характеристика RFID сканеров

	Zebra FX9600	Nordic ID Sampo S1	Chainway UR4 UHF
Протокол передачи	ISO 18000-63	ISO 18000-6C	ISO 18000-6C
Безопасность	UL 60950-01	Предназначен для помещений	UL 60950-01

Соединения	10/100 BaseT Ethernet (RJ45); USB со стороны контроллера и клиента (тип А и В);	USB или Ethernet	RS232
Рабочая температура	От -20° до +55°С	от -20 °С до 55 °С	От -25 до 65 °С
Влажность	5-95%	5-70%	10-95%
Герметичность	IP53	нет	IP53
Индикаторы состояния	Многоцветные светодиоды: питание, работа, состояние и приложения	Светодиодные индикаторы: Питание, RFID вкл., RFID выкл., Передача данных	Многоцветные светодиоды: питание, работа, состояние и приложения
Цена	109000	75000	35000

На основании приведенных технических характеристик можно сделать вывод, что все три сканера являются подходящими для разрабатываемой системы. Самый дорогой из них - FX9600 имеет ключевое преимущество в защищенности корпуса датчика. Но для WMS это не является необходимым критерием при разработке. Исходя из приведенной таблицы и приоритета на наименьшую цену установки это был выбран UR4 UHF, который имеет наименьшую цену, но тем не менее имеет более высокую частоту приема, алюминиевый корпус, производственный стандарт передачи данных RS232 и самую низкую цену среди аналогов.

4.3.3 Выбор обрабатывающего устройства

В качестве управляющего устройства для механических установок и обработки информации с датчиков могут подходить как промышленные микроконтроллеры, так и одноплатные компьютеры, например, Raspberry pi.

Для того, какой тип управляющего устройства подходит в решении необходимо разобрать какие функции он будет выполнять. На основании это было принято решение о выборе.

Список функций устройства:

1. Обработка таблицы *.csv и передача данных в базу данных
2. Обработка команд оператора и отправка сигнала на исполнительные механизмы: конвейер, подъемники
3. Обеспечение бесперебойной работы человеко-машинного интерфейса
4. Передача данных по двум каналам передачи: WiFi, Ethernet
5. Обработка входящего потока RFID меток
6. Сортировка приходящих грузов
7. Отправка команд на отгрузку грузов

Данный список не отражает менее важные функции, которые стоят перед управляющим устройством в WMS системе. Также для использования микроконтроллера необходимо разработать электрическую схему, включающую ethernet, модуль WiFi, RS232, блок питания. Также в схеме необходимо учесть требования питания, пожаробезопасность и изготовить корпус устройства. На основании это было принято решение, что использование одноплатного компьютера Raspberry pi является оптимальным решением, так как в нем есть все вышеперечисленное.

4.4 Разработка алгоритма управления сортировкой грузов

Ключевой характеристикой автоматизированной системой управления складом является скорость доставки грузов на отгрузку. Данный параметр зависит не только от скорости перемещения груза по конвейеру, но и от выбранного алгоритма сортировки. Среди них алгоритм распределения по типу груза, последовательного заполнения, наименьшего время доставки до отгрузки. В разрабатываемом проекте выбран последний тип сортировки. Так как он является оптимальным для склада, в котором регулярно товар перемещается между пунктами получения и выдачи, кроме того, было принято решение использовать наиболее простой вариант, он представлен на рисунке

4.2. Алгоритм последовательного заполнения является неоптимальным так, как одна часть склада будет меньше заполняться, что приведет к простоям оборудования.

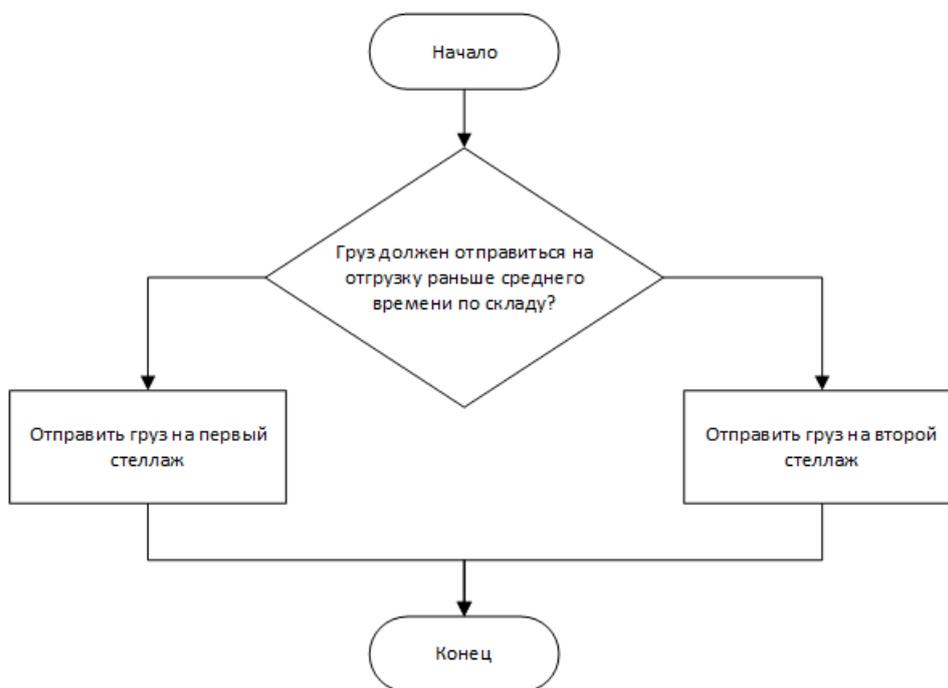


Рисунок 4.2 – Выбранный алгоритм сортировки грузов

4.5 Разработка программного обеспечения

4.5.1 Разработка модели в программе Factory I/O

Пользователя, начинающего работу с Factory I/O, встречает начальный экран приложения, показанный на рисунке 4.3. В верхней части окна программы находятся область управления моделированием (запуск симуляции, остановка, повтор, вид), панель задач. В правой области находятся все доступные для добавления в симуляцию объекты.

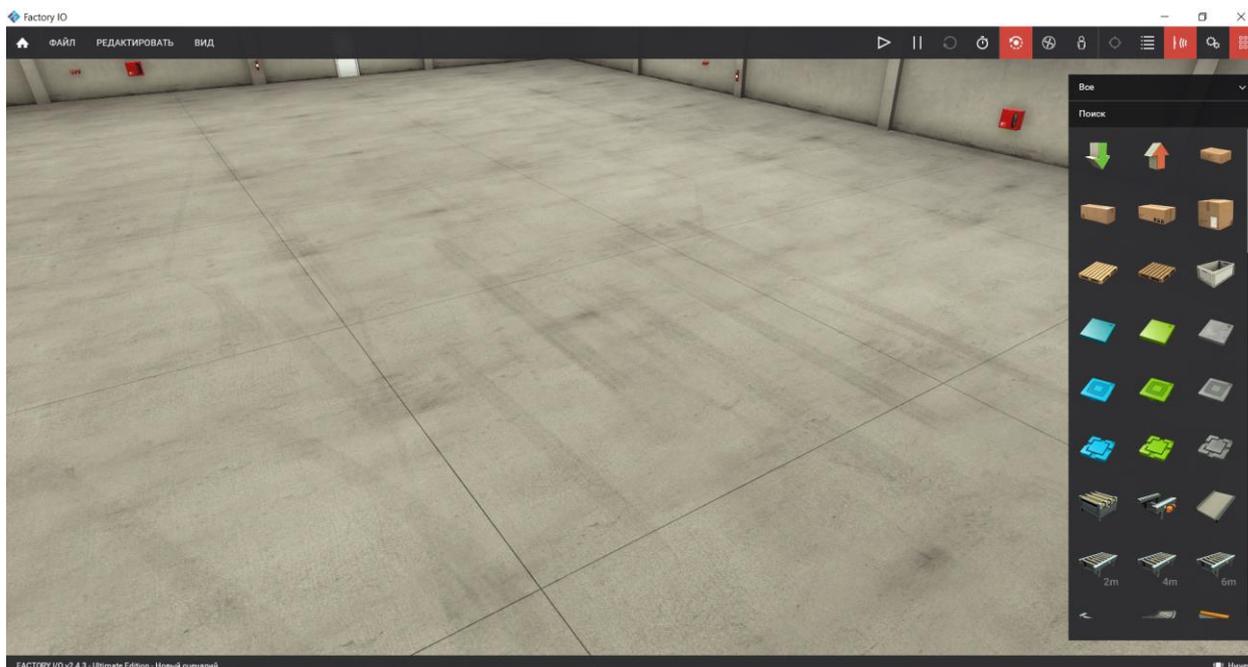


Рисунок 4.3 – Начало работы в Factory I/O

Сборка модели склада выполняется путем перемещения объектов в рабочее поле. В данном проекте необходимы следующие объекты:

1. Конвейер
2. Конвейер поворотный
3. Кран-штабелер
4. Стеллажи
5. Сканер RFID
6. Световозвращающий датчик

Результатом сборки вышеперечисленных компонентов получается модель, представленная на рисунке 4.4.

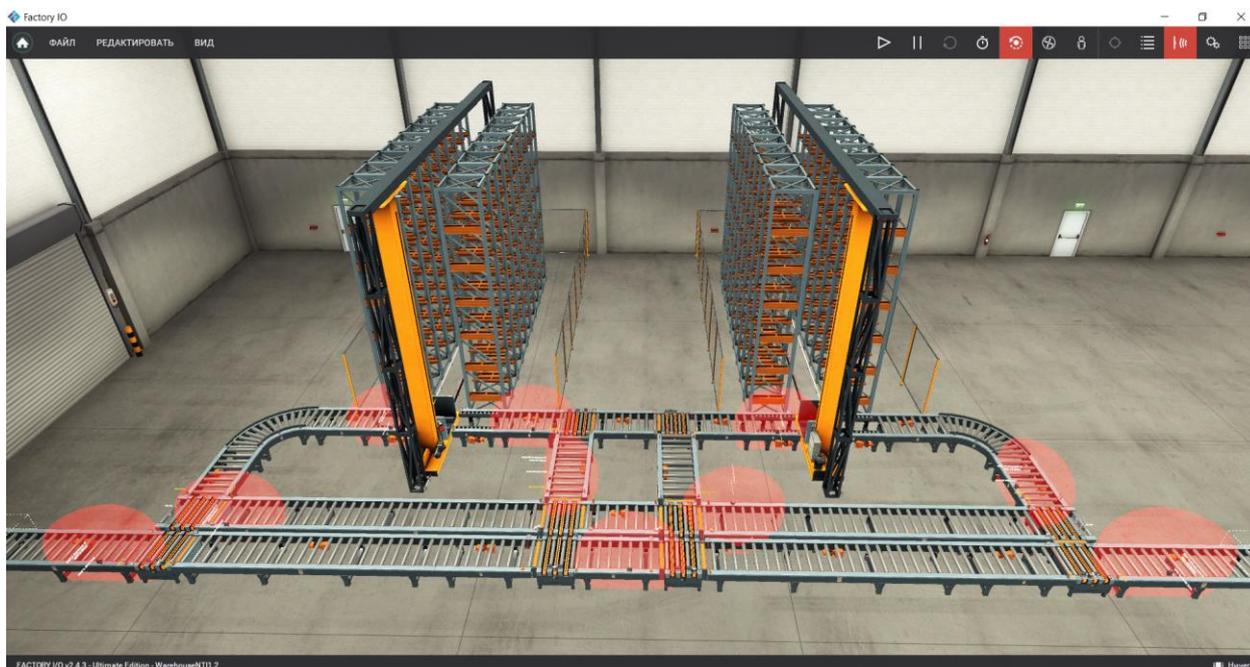


Рисунок 4.4 – Собранная модель Factory I/O

После сборки модели необходимо было написать исходный код, так как для работы требуется подключить SDK (комплект для разработки программного обеспечения) написанный специально для Factory I/O. SDK – это библиотека, написанная на языке программирования C# на платформе .NET, для упрощения работы с симуляцией, а также исключение необходимости писать исходный код по ПЛК.

Первым этапом в написании кода является запись тегов всех установок и датчиков, присутствующих в виртуальной модели. В процессе работы было написано порядка 200 тегов, отвечающих за идентификацию и создание сценария работы установки. Пример листинга кода тегов приведен в листинге кода 4.1, с полной версией можно ознакомиться в приложении Г.

```

MemoryBit loadrcA4 = MemoryMap.Instance.GetBit("Load RC A4",
MemoryType.Output);
MemoryBit unloadrcA5 = MemoryMap.Instance.GetBit("Unload RC A5",
MemoryType.Output);
MemoryInt targetPositionA = MemoryMap.Instance.GetInt("Target Position A",
MemoryType.Output);
MemoryBit forksLeftA = MemoryMap.Instance.GetBit("Forks Left A",
MemoryType.Output);
MemoryBit liftA = MemoryMap.Instance.GetBit("Lift A ", MemoryType.Output);
MemoryBit forksRightA = MemoryMap.Instance.GetBit("Forks Right A ",
MemoryType.Output);
    
```

При написании исходного кода необходимо следовать не только алгоритму, разработанному в разделе 4.5, но и логической последовательности движения грузов на складе. Так, например, логика движения должна препятствовать созданию пробок на конвейерной ленте. Чтобы избежать данных недоработок в коде была разработана схема движения грузов с указанием тегов. Часть схемы приведена на рисунке 4.5, полный чертеж представлен в приложении Б.

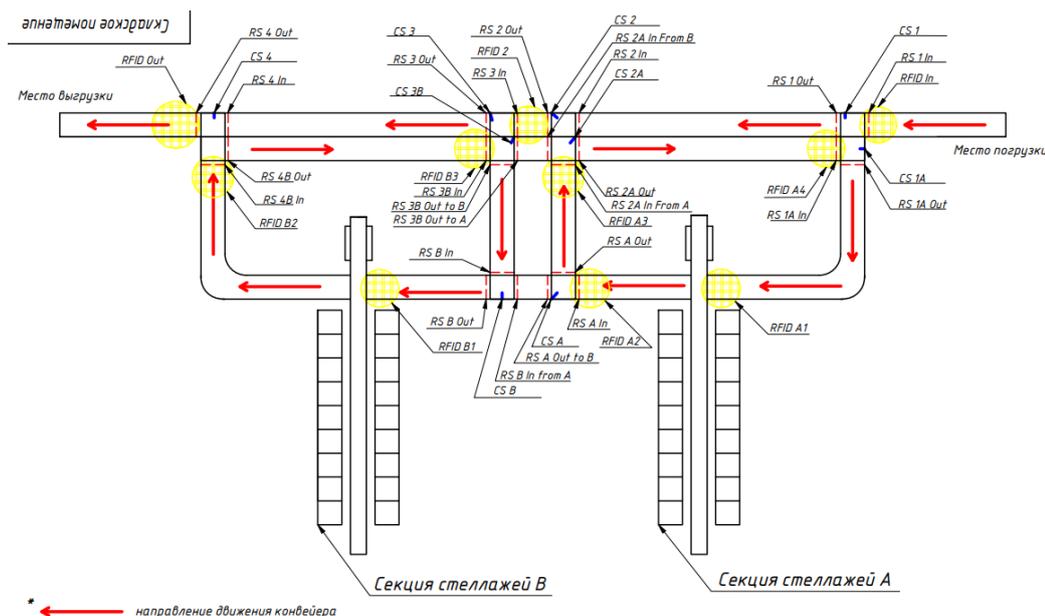


Рисунок 4.5 – схема движения грузов с указанием тегов

Так же на рисунке 4.6 приведена схема с указанием названий тегов приводов конвейеров. Полная схема представлена в приложении В.

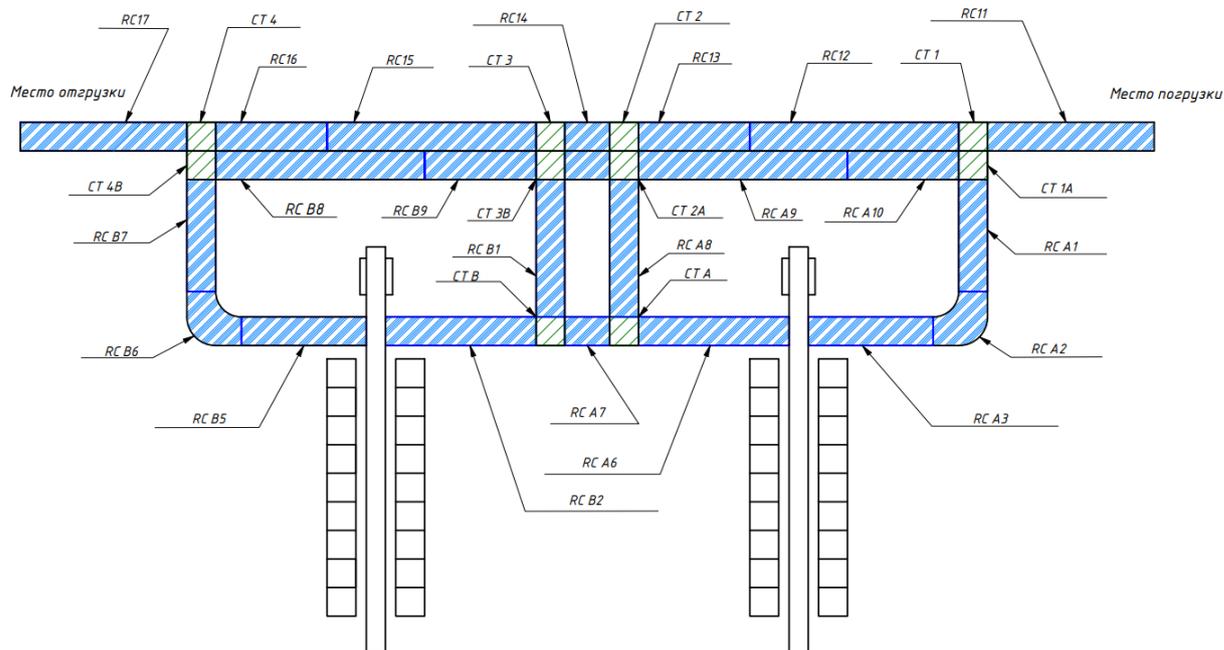


Рисунок 4.6 – Схема приводов автоматизированного склада

Создав подробный алгоритм работы и начертив схемы WMS, написание исходного кода становится по-настоящему простой и механической задачей. В листинге кода 4.2 показан отрезок кода, в котором описан процесс движения ящиков на складе. Полный листинг кода программы представлен в приложении Д.

После прохождения данного этапа разработки проекта модель склада полностью готова к работе и стабильно функционирует.

Листинг кода 4.2 – Исходный код с процессом движения ящиков по складу

```
if (state == State.State0)
{
    targetPosition.Value = 55;

    if (ftMovXZ.Q)
        state = State.State1;
}
else if (state == State.State1)
{
    //ожидание загрузки

    if (!atLoad.Value)
        state = State.State2;
}
else if (state == State.State2)
{
    forksLeft.Value = true;

    if (rtAtLeft.Q)
        state = State.State3;
}
else if (state == State.State3)
{
    lift.Value = true;

    if (ftMovZ.Q)
        state = State.State4;
}
else if (state == State.State4)
{
    forksLeft.Value = false;

    if (rtAtMiddle.Q)
        state = State.State5;
}
else if (state == State.State5)
{
    targetPosition.Value = ++currentPos;

    state = State.State6;
}
else if (state == State.State6)
{
    //смена местоположения ящика.

    if (ftMovXZ.Q)
        state = State.State7;
}
```

4.5.2 Разработка человеко-машинного интерфейса в программе Simple-Scada

Создание проекта на программной платформе Simple-Scada начинается с создания пустого проекта. На рисунке 4.7 показано приветственное окно проекта.

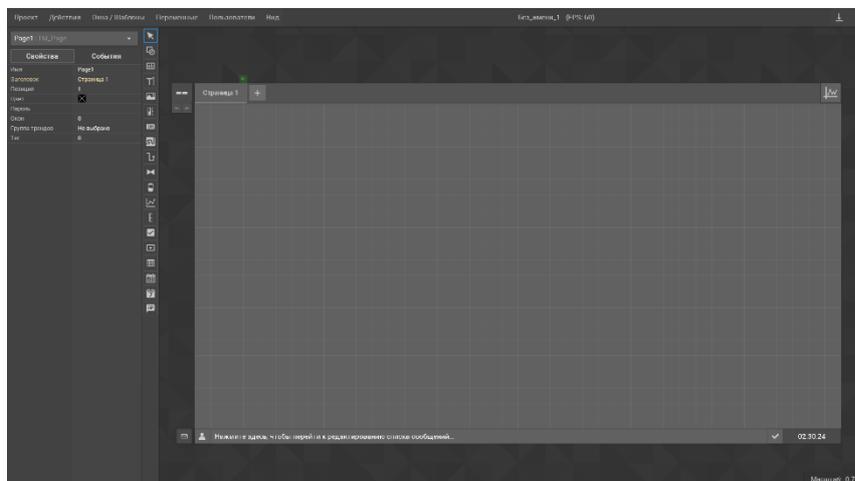


Рисунок 4.7 - Начальный экран проекта

Слева рабочего поля программы расположены главные элементы создания Scada системы, а также поля изменения свойств и событий системы. Создание рабочей панели не представляет особых сложностей. На рисунке 4.8 представлена готовый визуальный проект системы.

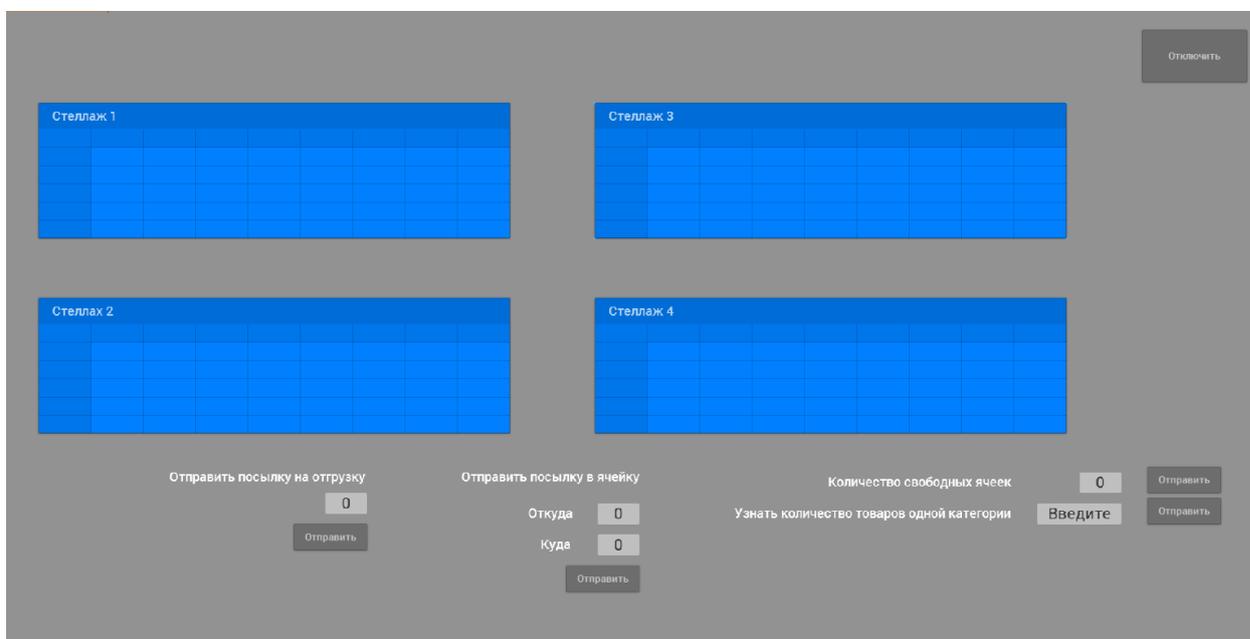


Рисунок 4.8 – Проект склада в Simple-Scada

Для взаимодействия с виртуальной моделью склада необходимо настроить связь SCADA с Factory I/O путем настройки OPC сервера.

Стандарт OPC разрабатывался с целью сократить затраты на создание и сопровождение приложений промышленной автоматизации. Суть OPC проста — предоставить разработчикам промышленных программ универсальный фиксированный интерфейс обмена данными с любыми устройствами.

Технология OPC уже стала мировым стандартом в области автоматизации технологических процессов. С каждым годом все большее количество оборудования поставляется с OPC-серверами. Сама технология OPC находится в процессе постоянного совершенствования и оптимизации, новые версии стандарта расширяют сферу применения продуктов, поддерживающих спецификации OPC.

Simple-Scada подпадает под категорию приложения "OPC-клиент" с обменом данными по спецификации OPC DA/UA. В этом заключается большое преимущество - Simple-Scada может работать с любыми OPC-серверами.

Настройка OPC сервера проводится через отдельную подпрограмму «Simple-Scada Server». Окно программы показано на рисунке 4.9. Последовательно выбирая вкладки: состояние сервера, проекты, OPC-серверы, клиенты, базы данных, журнал. Для такого простого, как разрабатываемый проект не требуется производить какие-либо специфичные настройки, кроме запуска проекта на локальном хосте по адресу 127.168.0.1. Путем несложных манипуляций, производим связь созданных тегов Factory I/O и событий в Simple-Scada, после чего производится запуск OPC сервера, симуляции Simple-Scada, исходного кода C# в visual studio и виртуальной модели Factory I/O.

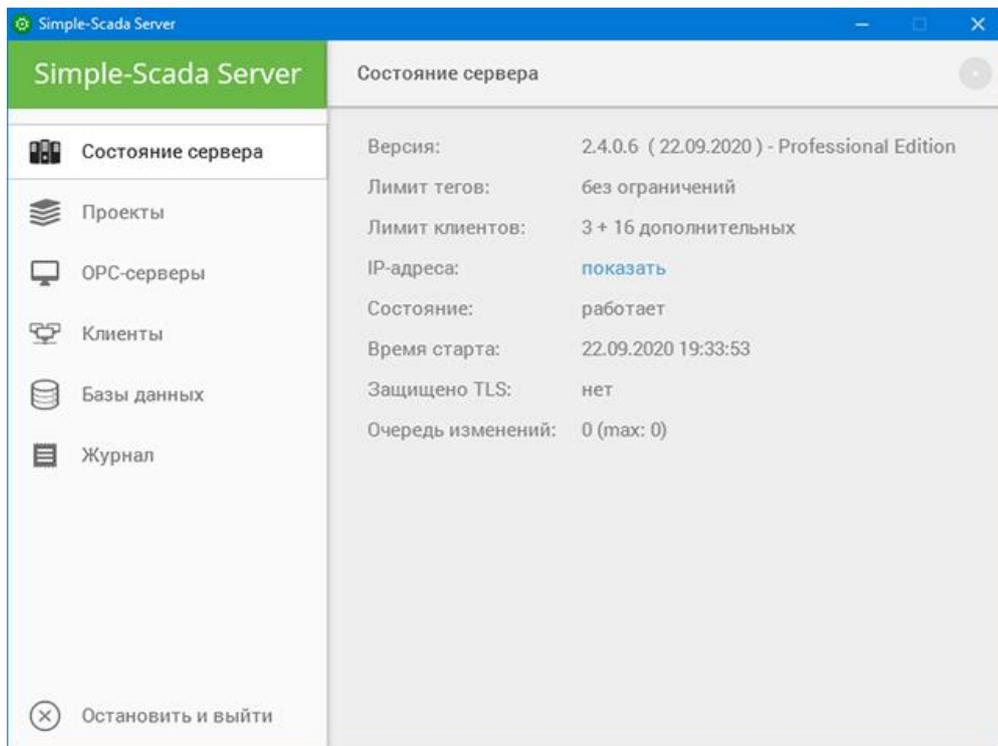


Рисунок 4.9 – Окно программы Simple-Scada server

5. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

На сегодняшний день перспективность научных разработок и исследований определяется не столько масштабом открытия, сколько их коммерческой ценностью. При поиске источников финансирования для проведения научного исследования необходимым условием является оценка коммерческой ценности разработки с целью получения прибыли в ходе ее коммерциализации. Коммерческая привлекательность открытий, помимо превышения технических параметров над уже существующими разработками, определяется еще и способностью разработчика быстро найти ответы на вопросы о востребованности его изобретения на рынке, конечной стоимости продукта, бюджете проекта и срок, за который продукт сможет выйти на рынок.

Целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является анализ и расчет основных параметров для реализации конкурентоспособной разработки, которая в перспективе принесет выгоду и позволит сократить издержки производства, а также будет отвечать требованиям ресурсоэффективности и ресурсосбережения. Суть разработки заключается в автоматизации процессов, происходящих в складских помещениях, а именно автоматизация приемки, отгрузки груза, сортировка его на складе и другие манипуляции с ними.

5.1 Предпроектный анализ

5.1.1 Потенциальные потребители результатов исследования

В ходе выполнения работы было разработано программное обеспечение для автоматизированной системы управления складом, разработана виртуальная модель и подобрано оборудование, соответствующее требованиям процесса. Потенциальными потребителями результатов данной разработки являются коммерческие организации, занимающиеся оптовой торговлей, доставкой посылок, розничной торговлей. Для проведения качественного анализа потребителей результатов разработки необходимо

рассмотреть целевой рынок и сегментировать его. Карта сегментирования рынка приведена в таблице 5.1.

Таблица 5.1 – Карта сегментирования рынка по актуальности научного исследования

		Размер складских объемов			
		Малый	Средний	Повышенный	Высокий
Размер компании	Мелкие				
	Средние				
	Крупные				

По приведенной карте сегментирования можно сделать вывод о том, что основными потребителями являются крупные компании, для которых важна автоматизация из-за больших проходящих объемов грузов в течение всей работы.

5.2 Анализ конкурентных технических решений

Для проведения данного анализа необходимо составить оценочную карту, в которой будут отображены сравнительные характеристики проектируемой АСУ ТП складских помещений и существующая система управления складом. В качестве конкурентной разработки будет выступать автоматизированная система хранения мелкоштучных товаров, разработанный компанией СКМАТИК. Для проведения наиболее эффективного анализа должна быть использована вся имеющаяся информация о конкурентных разработках, такая, как технические характеристики разработки, ее конкурентоспособность, уровень завершенности научного исследования, финансовое положение конкурентов и перспектива развития созданной установки.

Анализ конкурентных технических решений определяется по формуле

$$2: \quad K = \sum(V_i * B_i), \quad (2)$$

где K – конкурентоспособность разработки;

V_i – вес показателя;

B_i – балл i -го показателя.

Оценочная карта представлена в таблице 5.2.

Таблица 5.2 – Оценочная карта

Критерии оценки	Вес критерия	Баллы		Конкурентоспособность	
		Проектируемая АСУ ТП	Конкурентная разработка	Проектируемая АСУ ТП	Конкурентная разработка
Технические характеристики оценки ресурсоэффективности					
Удобство в эксплуатации	0,1	4	4	0,4	0,4
Безопасность	0,2	4	5	0,8	1
Быстродействие	0,07	5	3	0,35	0,21
Долговечность	0,08	4	3	0,32	0,24
Надежность	0,1	4	3	0,4	0,3
Точность измерений	0,1	5	4	0,5	0,4
Простота управления	0,05	4	3	0,2	0,15
Экономические критерии ресурсоэффективности					
Конкурентоспособность	0,07	3	1	0,21	0,07
Цена	0,1	3	3	0,3	0,3
Срок выхода на рынок	0,06	2	4	0,12	0,24

Продолжение таблицы 5.2 – Оценочная карта

Предполагаемый срок эксплуатации	0,07	4	3	0,28	0,21
Итого	1	42	38	3,88	3,52

Из приведенной таблицы можно сделать вывод о том, что недостатком конкурентной разработки является использование старого оборудования, что отрицательно сказывается на общей производительности установки.

По сравнению с уже существующей АСУ расходом нефти в УПН разрабатываемая система имеет следующие конкурентные преимущества: быстрое действие, надежность, долговечность, предполагаемый срок эксплуатации.

5.3 SWOT-анализ

SWOT-анализ представляет собой комплексный анализ разрабатываемого проекта, необходимый для исследования внешней и внутренней среды проекта. По результатам его проведения у организации появляется представление о перспективах развития и возможных угрозах.

Итоговая матрица SWOT-анализа представлена в таблице 5.3.

Таблица 5.3 – SWOT-анализ

	<p>Сильные стороны:</p> <p>С1: Надежность;</p> <p>С2: Простота эксплуатации;</p> <p>С3: Экономичность за счет использования оборудования от отечественных производителей;</p> <p>С4: Экологичность технологии;</p>	<p>Слабые стороны:</p> <p>Сл1: Длительный период доставки груза до ячейки хранения;</p> <p>Сл2: Длительный срок поставки оборудования;</p> <p>Сл3: Отсутствие возможности в производстве разрабатываемых установок «под ключ»;</p> <p>Сл4: Отсутствие квалифицированного персонала для поддержания корректной работы установки у потребителя.</p>
<p>Возможности:</p> <p>В1: Повышение стоимости конкурентных разработок;</p> <p>В2: Сотрудничество с изготовителями необходимого оборудования;</p> <p>В3: Появление спроса на новую разработку;</p> <p>В4: Использование оборудования отечественных производителей.</p>	<p>Создание конкурентоспособного проекта. Использование доступного современного оборудования, способного повысить качество и производительность работы установки.</p>	<p>Замена необходимого оборудования на схожее по функциональности.</p> <p>Проведение вводного инструктажа по эксплуатации установки на предприятии заказчика.</p>
<p>Угрозы:</p> <p>У1: Отсутствие спроса на продукты;</p> <p>У2: Высокая конкуренция в выбранной отрасли;</p> <p>У3: Увеличение цен необходимого оборудования;</p> <p>У4: Появление более качественных аналогов.</p>	<p>Улучшение качества работы системы.</p> <p>Выявление преимуществ новой системы над существующей.</p> <p>Применение разработки на внутреннем рынке.</p>	<p>Осуществление поиска и закупки оборудования заблаговременно.</p> <p>Совершенствования установки с учетом существующего оборудования.</p>

В ходе проведения SWOT-анализа выявлены возможности и угрозы для развития автоматизированной системы управления в будущем. Также по результатам проведенного SWOT-анализа можно сделать вывод о преимуществах разрабатываемой системы: надежность установки, простота эксплуатации, экономичность, экологичность производства. Помимо преимуществ выявлены и недостатки: длительный срок поставки

оборудования, отсутствие возможности собирать установки «под ключ». Для устранения слабых сторон и минимизации последствий в случае возникновения угроз необходимо регулярно совершенствовать установку с использованием современных технологий и оборудования, а также заранее осуществлять закупку необходимого оборудования.

5.4 Планирование научно-исследовательской работы

5.4.1 Структура работ в рамках научного исследования

Для наиболее эффективного достижения результата необходимо разбить процесс выполнения ВКР на этапы. При реализации проекта рассматриваются 2 исполнителя: консультант (К) и инженер (И). примерный порядок составления этапов и распределение исполнителей по ним приведен в таблице 5.4.

Таблица 5.4 – Перечень этапов и распределение исполнителей

Этапы работы	Загрузка исполнителей
1. Постановка целей и задач исследования	К – 80% И – 20%
2. Обзор и изучение научной литературы	И – 100%
3. Сбор данных по существующим разработкам	И – 100%
4. Разработка технического задания	К – 30% И – 70%
5. Календарное планирование работ	К – 20% И – 80%
6. Описание технологического процесса	И – 100%
7. Разработка структурной схемы	И – 100%
8. Разработка функциональной схемы автоматизации	И – 100%
9. Разработка схемы информационных потоков	И – 100%
10. Разработка алгоритмов управления	К – 20% И – 80%
11. Проектирование SCADA-системы	И – 100%
12. Оформление расчётно-пояснительной записки	И – 100%
13. Подведение итогов	К – 100%

5.4.2 Разработка графика проведения научно-технического исследования

Трудоёмкость выполнения исследования оценивается экспертным путём в силу вероятностного характера величины. За единицу измерения трудоёмкости принимаются человеко-дни. Ожидаемая трудоёмкость рассчитывается по формуле 3:

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

где $t_{ож}$ – ожидаемая трудоёмкость выполнения работы чел.дн;

t_{\min} – минимально возможная трудоёмкость выполнения заданной работы, чел.дн;

t_{\max} – максимально возможная трудоёмкость выполнения заданной работы, чел.дн.

Для построения графика проведения научных работ в форме диаграммы Ганта длительность каждого из этапов работ в рабочих днях переводится в календарные дни по формуле 4:

$$T_{КД} = T_{РД} \cdot K_{КД}, \quad (4)$$

где $T_{КД}$ – продолжительность выполнения работы в календарных днях;

$T_{РД}$ – продолжительность выполнения работы в рабочих днях;

$K_{КД}$ – коэффициент календарности.

Коэффициент календарности рассчитывается по формуле 5:

$$K_{КД} = \frac{T_{КД}}{T_{КД} - T_{ВД} - T_{ПД}}, \quad (5)$$

где $T_{КД}$ – количество календарных дней в году;

$T_{ВД}$ – количество выходных дней в году;

$T_{ПД}$ – количество праздничных дней в году.

Значение коэффициента календарности для 2021 года 6:

$$K_{КД} = \frac{365}{365-118} = \frac{365}{247} = 1,45$$

(6)

С учетом данных таблицы 5.3 и приведенных выше формул составляется расчетная таблица 5.4. Диаграмма Ганта, представляющая собой календарный график работ, приведена в таблице 5.5.

Таблица 5.5 – Временные метрики реализации НТИ

Название работы	Трудоемкость работы						Длит. работ ы в рабоч их днях T P i		Длит. работы в календарн ых днях Tki	
	t_{min} , чел. - дн.		t_{max} , чел. - дн.		$t_{ож}$, чел. - дн.					
	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер	Руководитель	Инженер
Составление и утверждение ТЗ	2	2	4	4	2.8	2.8	1.4	1.4	2.058	2.06
Изучение необходимой литературы		8		10	0	8.8	0	8.8	0	12.9
Изучение аналогов устройства		5		8	0	6.2	0	6.2	0	9.11
Календарное планирование	1	1	2	2	1.4	1.4	0.7	0.7	1.029	1.03
Подбор компонентов		3		5	0	3.8	0	3.8	0	5.59

для реализации устройства										
Покупка компонентов		1		2	0	1.4	0	1.4	0	2.06
Сборка и настройка аппаратной части устройства		7		10	0	8.2	0	8.2	0	12.1
Написание программной части		3		5	0	3.8	0	3.8	0	5.59
Тестирование и отладка человеко-машинного интерфейса	3	3	5	5	3.8	3.8	1.9	1.9	2.793	2.79
Доработка НМИ-панели		2		3	0	2.4	0	2.4	0	3.53
Анализ полученных результатов		1		3	0	1.8	0	1.8	0	2.65
Написание пояснительной записки		10		15	0	12	0	12	0	17.6
Итого:							4	52.4	5.88	77

Базируясь на данных, представленных в таблице 5.4 построим календарный план–график, представленный в таблице 5.6. Для начала разобьем работы, а точнее ход их выполнения, на месяцы и декады, а затем построим график для максимальной длительности реализации работ в рамках разработки прототипа человеко-машинного интерфейса.

Таблица 5.6 – Диаграмма Ганта, план график работ

№ работ	Виды работ	Исполнители		Продолжительность выполнения работ												
				февр.			март			апр.			май			
				1	2	3	1	2	3	1	2	3	1	2	3	
1	Составление и утверждение ТЗ	Руководитель Инженер	2.058 2.06	█												
2	Изучение необходимой литературы	Инженер	12.9	█	█	█										
3	Изучение аналогов устройства	Инженер	9.11			█	█	█								
4	Календарное планирование	Руководитель Инженер	1.029 1.03				█									
5	Описание технологического процесса	Инженер	5.59				█	█								
6	Разработка структурной схемы		2.06					█								
7	Разработка функциональной части	Инженер	12.1					█	█	█						
8	Написание программной части	Инженер	5.59							□						
9	Тестирование и отладка человеко-машинного интерфейса	Руководитель Инженер	2.793 2.79								█	█				
10	Доработка человеко-машинной панели	Инженер	3.53								□					
11	Анализ полученных результатов	Инженер	2.65									█	█			
12	Написание пояснительной записки	Инженер	17.6										█	█	█	█

5.5 Бюджет научно-технического исследования

Данный пункт включает в себя все затраты, необходимые на проведение исследования до начала осуществления работ, что позволит судить об экономической эффективности работы. В этом разделе учитываются следующие статьи расходов:

1. материальные затраты;
2. амортизационные отчисления;
3. заработная плата исполнителей;
4. отчисления во внебюджетные фонды;
5. накладные расходы.

5.5.1 Расчёт материальных затрат

Материальные затраты представляют собой совокупность материальных ценностей, расходуемых в процессе выполнения работ.

Сбор необходимой информации по существующим разработкам и выполнение практической части работы требует ряд определенных программ, таких, как Microsoft Office, Factory IO, Компас-3D и др. К основной части программ студенты Томского Политехнического университета имеют бесплатный доступ, остальные можно найти в свободном доступе в сети Интернет. Основные затраты представляют собой расходы на канцелярские принадлежности, используемые в ходе работы. Основная работа производилась с использованием ноутбука с бесплатным доступом к программе Factory IO.

5.5.2 Расчет амортизационных отчислений

В качестве основного оборудования выступает ноутбук, так как на нем выполняются все основные этапы работы: сбор информации, создание схем автоматизации, моделирование системы в программе Factory IO, оформление расчетно-пояснительной записки. По календарному плану время написания выпускной квалификационной работы занимает 5 месяцев. Стоимость ноутбука 25000 рублей. Срок полезного использования для офисной техники составляет около 3 лет.

Норма амортизации N_A рассчитывается как (7):

$$N_A = \frac{1}{T} * 100\%, \quad (7)$$

где T – срок полезного использования, лет (в нашем случае 3 года).

Тогда норма амортизации:

$$N_A = \frac{1}{3} * 100\% = 33,3\%. \quad (8)$$

Годовые амортизационные отчисления:

$$A_{\text{год}} = 25000 * 0,33 = 8250 \text{ руб.} \quad (9)$$

Ежемесячные амортизационные отчисления:

$$A_{\text{мес}} = \frac{8250}{12} = 687,5 \text{ руб.} \quad (10)$$

Итоговая сумма амортизации основных средств:

$$A = 687,5 * 3437,5 \text{ руб.} \quad (11)$$

5.5.3 Расчёт заработной платы и отчислений во внебюджетные фонды

Основная заработная плата студента и консультанта за выполнение НТИ, рассчитывается по формуле 12:

$$Z_{\text{осн}} = Z_{\text{дн}} * T_p, \quad (12)$$

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

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

$Z_{\text{дн}}$ – среднедневная заработная плата.

Среднедневная заработная плата $Z_{\text{дн}}$ рассчитывается по формуле 13:

$$Z_{\text{дн}} = \frac{Z_M * M}{F_d} \quad (13)$$

где Z_M – месячный должностной оклад работника, руб.;

M – количество месяцев работы без отпуска в течение года:

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

F_d – действительный годовой фонд рабочего времени научно-технического

персонала (в рабочих днях).

В таблице 16 представлен баланс рабочего времени.

Таблица 5.7 – Баланс рабочего времени

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

Месячный должностной оклад работника рассчитывается по формуле 14:

$$Z_M = Z_{TC} * (1 + k_{пр} + k_d) * k_p, \quad (14)$$

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

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

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

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

Оклад консультанта принимается равным 15000 рубля, оклад инженера принимается равным размеру не ниже МРОТ – 15000 рублей.

Расчет основной заработной платы приведен в таблице 5.8.

Таблица 5.8 – Основная заработная плата

Исполнитель и	Оклад, руб.	$k_{пр}$	$k_{д}$	$k_{р}$	$Z_{м}$, руб.	$Z_{дн}$, руб.	$T_{р}$, раб. дн.	$Z_{осн}$, руб.
Консультант	15000	0,3	0,2	1,3	29250	1212	7,8	9453,6
Инженер (студент)	15000	0,3	0,2	1,3	29250	1212	84,56	102486,72

Дополнительная заработная плата рассчитывается по формуле 15:

$$Z_{доп} = Z_{осн} * k_{доп}, \quad (15)$$

где $k_{доп}$ – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,1).

Отчисления во внебюджетные фонды в соответствии с Налоговым кодексом РФ рассчитываются по формуле 16:

$$Z_{внеб} = (Z_{осн} + Z_{доп}) * 0,302, \quad (16)$$

где $Z_{осн}$ – основная заработная плата, руб.;

$Z_{доп}$ – дополнительная заработная плата, руб.

Результаты расчетов дополнительной заработной платы и отчислений во внебюджетные фонды представлены в таблице 5.9.

Таблица 5.9 – Расчет дополнительной заработной платы и отчислений во внебюджетные фонды

Исполнители	$Z_{осн}$	$Z_{доп}$	$Z_{внеб}$
Консультант	9453,6	945,36	3140,5
Инженер	102486,72	10248,67	34046,1
Итого	111940,32	11194	37186,6

5.5.4 Накладные расходы

Накладные расходы учитывают прочие затраты компании, не

попавшие в предыдущие статьи расходов: ксерокопирование и печать материалов исследования, размножение материалов, оплата услуг связи и электроэнергии и т.д. Величина накладных расходов определяется по формуле 17:

$$Z_{\text{накл}} = \text{сумма всех статей} * k_{\text{нр}}, \quad (17)$$

где $k_{\text{нр}}$ – коэффициент, учитывающий накладные расходы (составляет 10%).

$$Z_{\text{накл}} = (3437,5 + 111940,32 + 11194 + 37186,6) * 0,1 = 16375,8.$$

Сумма накладных расходов равна 16375,8 рублей.

5.5.5 Формирование бюджета научно-технического исследования

Расчет бюджета затрат на научно-техническое исследование приведен в таблице 5.10.

Таблица 5.10 – Расчет бюджета затрат научно-техническое исследование

Наименование статьи	Сумма, руб.	Удельный вес, %
Затраты на амортизацию	3437,5	1,9
Затраты на основную заработную плату	111940,32	62,14
Затраты на дополнительную заработную плату	11194	6,24
Отчисления во внебюджетные фонды	37186,6	20,6
Накладные расходы	16375,8	9,12
Итого:	180134,22	100

По результатам проведенных расчетов можно сделать вывод о том, что основная часть затрат на проведение научно-технического исследования расходуется на заработную плату исполнителей.

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

Интегральный финансовый показатель разработки рассчитывается как:

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}}, \quad (18)$$

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

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

Φ_{max} – максимальная стоимость исполнения.

Интегральный показатель ресурсоэффективности рассчитывается как:

$$I_{pi} = \sum a_i * b_i, \quad (19)$$

где I_{pi} – интегральный показатель ресурсоэффективности;

a_i – весовой коэффициент i -го варианта исполнения разработки;

b_i – балльная оценка i -го варианта исполнения разработки.

Сравнительная оценка характеристик представлена в таблице 5.11.

Таблица 5.11 – Сравнительная оценка характеристик вариантов исполнения проекта

Критерии	Весовой коэффициент параметра	Баллы		Ресурсоэффективность	
		Проектируемая разработка	Конкурентная разработка	Проектируемая разработка	Конкурентная разработка
Удобство в эксплуатации	0,2	4	4	0,8	0,8
Безопасность	0,2	4	5	0,8	1
Быстродействие	0,1	5	3	0,5	0,3
Надежность	0,2	4	3	0,8	0,6
Долговечность	0,1	4	3	0,4	0,3
Точность измерений	0,1	5	4	0,5	0,4

Простота управления	0,1	4	3	0,4	0,3
Итого	1	30	25	4,2	3,7

Из расчетов, приведенных в таблице 18 можно сделать вывод о том, что проектируемая разработка по интегральному показателю ресурсоэффективности является более выгодным и превосходит уже существующую разработку. В силу того, что проект является научной разработкой, рассчитать интегральный финансовый показатель разработки не представляется возможным.

Заключение по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

В данном разделе выявлены основные потребители, разработка может применяться крупными логистическими компаниями в полном объеме и частично мелкими и средними. Проведен анализ конкурентных разработок, в качестве конкурентной разработки будет выступать автоматизированная логистическая линия компании СКАМАТИК. По сравнению с уже существующей АСУ разрабатываемая система имеет следующие конкурентные преимущества: быстрдействие, надежность, долговечность, предполагаемый срок эксплуатации.

В ходе проведения SWOT-анализа выявлены возможности и угрозы для развития автоматизированной системы управления в будущем. Способы минимизации влияния выявленных угроз представлены в подразделе 3. Расчет бюджета научно-технического исследования показал, что основная часть затраченных средств уйдет на заработную плату сотрудникам, меньше всего средств уйдет на материальные затраты. Общий бюджет разработки составляет 180134,22 рублей, продолжительность работ – 116 дней.

Заключительным этапом была оценка ресурсоэффективности разрабатываемого проекта. Оценка показала, что разрабатываемый проект превосходит уже существующий по интегральному показателю ресурсоэффективности, а рассчитать интегральный финансовый показатель

разработки не представляется возможным ввиду того, что разрабатываемый проект является лишь научно-исследовательской разработкой.

6. Социальная ответственность

Эффективность деятельности предприятия напрямую зависит от комфортной организации и регулярного улучшения условий труда сотрудников предприятия, так как условия труда подразумевают под собой совокупность факторов, оказывающих непосредственное влияние на здоровье и трудоспособность сотрудников, что прямым образом влияет на результат их деятельности и деятельность предприятия в целом.

Любая производственная деятельность связана с воздействием вредных и опасных факторов производства на работника. Основными задачами обеспечения безопасности жизнедеятельности являются: идентификация опасности и оценка негативных воздействий рабочего места, предупреждение вредных воздействий на человека, защита от опасных факторов, создание комфортных условий труда для сотрудника.

В данной работе рассматривается динамическая визуализированная модель склада и его интеграция с виртуальным программно-техническим комплексом управления, система позволяет осуществлять контроль за процессом без непосредственного присутствия работника вблизи установки.

При такой автоматизации роль обслуживающего персонала ограничивается наблюдением за оптимальной работой установки и исправностью оборудования, ремонтом и наладкой аппаратуры в случае ее поломки. Контроль за установкой в основном связан с работой за персональным компьютером, где оператор может наблюдать за показателями необходимых величин и управлять работой установки.

При постоянной работе за компьютером человек подвергается воздействию следующих опасных и вредных производственных факторов: отклонения показателей микроклимата, повышенный уровень шума, недостаток света на рабочем месте. Помимо этого, работа за компьютером характеризуется умственным напряжением, нагрузкой на глаза, спину,

шейный отдел позвоночника, монотонностью трудового процесса.

6.1 Правовые вопросы обеспечения безопасности

Независимо от организационно-правовых форм и форм собственности государственный контроль в организациях осуществляется уполномоченными государственными органами и инспекциями. К числу таких органов можно отнести Федеральную инспекцию труда, Федеральную службу по надзору в сфере защиты прав потребителей и благополучия человека, а также государственную экспертизу условий труда. Права и обязанности работодателя и сотрудника регламентируются Трудовым кодексом Российской Федерации.

Опираясь на трудовой кодекс Российской Федерации, можно сделать вывод о том, что в условиях непрерывной работы производственного предприятия сотрудники не имеют возможности работать по пяти- или шестидневной рабочей неделе. В связи с этим персонал имеет сменные графики, которые позволяют осуществлять непрерывное обслуживание производственных процессов, регулярные выходные дни для отдыха каждой бригады, постоянный состав бригады, находящейся на смене, работу сотрудников сменами, имеющими постоянную продолжительность. Обычно на предприятиях применяется график, состоящий из четырех бригад сменности, при этом непосредственную трудовую деятельность осуществляют 3 бригады, тогда как четвертая отдыхает. Составление графиков работы бригад осуществляется на основании ст. 110 трудового кодекса Российской Федерации о предоставлении работникам еженедельного непрерывного отдыха продолжительностью не менее 42 часов.

6.2 Эргономические требования

Под рабочим местом оператора подразумевается зона трудовой деятельности оператора, оснащенная техническими средствами и вспомогательным оборудованием, необходимым для осуществления функций контроля и управления системой и объектом. Рабочее место при выполнении работ в положении сидя должно соответствовать требованиям ГОСТ 12.2.032-

78. Целью организации рабочего места является оптимизация условий труда, обеспечивающих максимальную эффективность и надежность работы оператора. При организации рабочего места необходимо опираться на требования эргономики, то есть учитывать все факторы, влияющие на эффективность действий человека при обеспечении безопасных приемов его работы.

Размер рабочего места зависит от характера труда и может ограничиваться площадью, оснащенной основным и вспомогательным технологическим оборудованием, технологической оснасткой, инструментами и приспособлениями, а также пультом и щитом управления. Рабочие места проектируются с учетом антропометрических данных человека усредненных размеров человеческого организма, так как несоответствие размещения технологического оборудования физиологическим возможностям оператора может существенно повлиять на качество работы и скорость утомляемости сотрудника. Комфортной рабочей средой рабочего места называется такое состояние внешней среды на рабочем месте, которое обеспечивает оптимальную динамику работоспособности оператора, хорошее самочувствие и сохранение его здоровья.

Площадь на одно рабочее место пользователей ПЭВМ с ВДТ на базе электроннолучевой трубки (ЭЛТ) должна составлять не менее 6 м², а объем не менее 20 м³. Схемы размещения рабочих мест с ВДТ и ПЭВМ должны учитывать расстояния между рабочими столами с видеомониторами (в направлении тыла поверхности одного видеомонитора и экрана другого видеомонитора), которое должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов - не менее 1,2 м. По отношению к проемам с естественным светом рабочие места операторов должны располагаться так, чтобы он падал сбоку, по возможности слева.

В случае выполнения пользователями ПЭВМ какой-либо творческой работы, требующей значительного умственного напряжения, рекомендуется изолировать их рабочие места друг от друга перегородками.

Конструкция рабочего стула (кресла) должна обеспечивать поддержание рациональной рабочей позы при выполнении работы, позволять изменять позу с целью снижения статического напряжения мышц шейно-плечевой области и спины для предупреждения развития утомления. Тип рабочего стула (кресла) должен выбираться в зависимости от характера и продолжительности работы с ВДТ и ПЭВМ с учетом роста пользователя. Рабочий стул (кресло) должен быть подъемно-поворотным и регулируемым по высоте и углам наклона сиденья и спинки, а также расстоянию спинки от переднего края сиденья, при этом регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию. Поверхность сиденья, спинки и других элементов стула (кресла) должна быть полумягкой, с нескользящим, слабо электризующимся и воздухопроницаемым покрытием, обеспечивающим легкую очистку от загрязнений.

Высота рабочей поверхности стола для взрослых пользователей должна регулироваться в пределах 680 - 800 мм; при отсутствии такой возможности высота рабочей поверхности стола должна составлять 725 мм. Модульными размерами рабочей поверхности стола для ПЭВМ, на основании которых должны рассчитываться конструктивные размеры, следует считать: ширину 800, 1000, 1200 и 1400 мм, глубину 800 и 1000 мм при нерегулируемой его высоте, равной 725 мм. Рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной - не менее 500 мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног - не менее 650 мм.

6.3 Производственная безопасность

Для выбора факторов необходимо использовать ГОСТ 12.0.003-2015 «Опасные и вредные факторы. Классификация». Перечень опасных и вредных факторов, характерных для проектируемой автоматизированной системы управления складом представлен в таблице 6.1.

Таблица 6.1 – Возможные опасные и вредные факторы

Факторы (12.0.003 - ГОСТ 2015 ССБТ)	Этапы работ			Нормативные документы
	Установка всего необходимого программного обеспечения	Разработка программного обеспечения	Эксплуатация	
Отклонение показателей микроклимата	+	+	+	1. СанПиН 2.2.4.548–96. Гигиенические требования к микроклимату производственных помещений. 2. ГОСТ 12.1.003-2014 ССБТ. Шум. Общие требования безопасности. 3. СП 52.13330.2016 Естественное и искусственное Освещение. 4. ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности. 5. СП 60.13330.2016 Отопление, вентиляция и кондиционирование воздуха. 6. ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление, зануление.
Повышенный уровень шума			+	
Недостаток естественного освещения и недостаточная освещённость рабочей зоны	+	+	+	
Электромагнитные излучения	+	+	+	
Электрический ток	+		+	

6.3.1 Анализ вредных факторов

6.3.1.1 Недостаточная освещённость рабочей зоны; отсутствие или недостаток естественного света

Рабочая зона или рабочее место оператора автоматизированной системы управления складом освещается таким образом, чтобы можно было отчетливо видеть процесс работы, не напрягая зрения, а также исключается прямое попадание лучей источника света в глаза.

Недостаточная освещенность рабочего пространства неблагоприятно влияет на органы зрения. Для обеспечения рационального освещения (отвечающего техническим и санитарно-гигиеническим нормам) необходимо правильно подобрать светильники в сочетании с естественным светом.

Поддерживать чистоту оконных стекол и поверхностей светильников. Правильно организованное освещение создает благоприятные условия труда, повышает работоспособность, действует на человека тонизирующие, создаёт хорошее настроение, улучшает протекание основных процессов нервной высшей деятельности и увеличивает производительность труда. Постоянная работа перед монитором способствует зрительному перенапряжению.

Кроме того, уровень необходимого освещения определяется степенью точности зрительных работ. Наименьший размер объекта различения составляет 0.5 - 1 мм. В помещении присутствует естественное освещение. По нормам освещенности и отраслевым нормам, работа за ПК относится к зрительным работам высокой точности для любого типа помещений. Нормирование освещенности для работы за ПК приведено в таблице 6.2.

Таблица 6.2 – Нормирование освещенности для работы с ПК

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта различения, мм	Разряд зрительной работы	Подразряд зрительной работы	Относительная производительность	Искусственное освещение				Естественное освещение	
					Освещенность на рабочей поверхности от системы общего освещения	Цилиндрическая освещенность ПК	Объединенный показатель URG, не более	Коэффициент пульсации освещенности К _п , не более	КЕО еп, %, при	
									Верхнем иликом бинированном	Боковым
Высокой точности	От 0,3 до 0,5	Б	1	Не менее 70	300	100	21	15	22,3	11
			2	Не менее 70	200	75	24	15	22,5	00,7

Требования к освещению на рабочих местах, оборудованных ПК, представлены в таблице 6.3.

Таблица 6.3 – Требования к освещению на рабочих местах с ПК

Освещенность на рабочем столе	300-500 лк
Освещенность на экране ПК	Не выше 300 лк
Блики на экране	Не выше 40 кд/м ²
Показатель ослеплённости	Не более 20
Показатель дискомфорта	Не более 15
Отношение яркости: - между рабочими поверхностями	3:1-5:1
- между поверхностями стен и оборудования	10:1
Коэффициент пульсации	Не более 5%

В случае отключения рабочего освещения предусмотрено аварийное освещение $E = 10$ лк.

Эвакуационное освещение предусмотрено в проходах, на лестницах, которое обеспечивает освещенность в помещениях 0,5 лк, на открытых территориях 0,2 лк.

Светильники аварийного освещения присоединяются к независимому источнику питания, а светильники для эвакуации людей к сети независимого от рабочего освещения. Для аварийного освещения применяют светильники с лампами накаливания.

6.3.1.2 Повышенный уровень шума

Во время работы оператор автоматизированной системы управления складом может быть подвергнут одному из важных факторов, оказывающих влияние на качество выполняемой работы, - шуму. Источниками шума являются: осветительные приборы, кондиционер, вентиляция и звуки, доносящиеся с улицы.

Шум оказывает негативное влияние на органы слуха, а также на всю нервную систему, тем самым ослабляя внимание и снижая концентрацию сотрудника.

Шум является одним из важных факторов, оказывающих влияние на качество выполняемой работы. При выполнении работ с повышенными

требованиями к процессам наблюдения и дистанционного управления производственными циклами, рабочие места за пультами в кабинах наблюдения и дистанционного управления без речевой связи по телефону предельно допустимое звуковое давление равно 75 дБА.

Нормирование уровней шума в производственных условиях осуществляется в соответствии с ГОСТ 12.1.003-83 [25]. Согласно данному документу при выполнении основной работы на персональной электронно-вычислительной машине (ПЭВМ) уровень шума на рабочем месте не должен превышать 50 дБА.

В качестве средств коллективной защиты можно применять: звукопоглощающие материалы, звукоизоляцию, акустические экраны.

В качестве средств индивидуальной защиты работник может использовать специальные противошумные наушники.

Одним из самых простых и действенных способов облегчения работы, является отдых, поэтому целесообразно устраивать кратковременные перерывы в течении рабочего дня при отсутствующих источниках шума.

6.3.1.3 Отклонение показателей микроклимата

Основными характеристиками микроклимата являются влажность, температура и скорость движения воздуха. Продолжительное влияние неблагоприятных условий на человека могут привести к ухудшению самочувствия сотрудника, что снизит производительность его труда.

По степени физической тяжести работа оператора АСУ от носится к категории легких работ, а основная нагрузка на организм приходится на зрительные органы и позвоночник человека. Ввиду того, что основным видом работы оператора автоматизированной системы является работа с прикладным программным обеспечением и технической документацией, потенциальными источниками опасных и вредных факторов являются персональные компьютеры и мониторы. Для минимизации воздействия этих факторов на человека в помещении должны быть предусмотрены оптимальные параметры микроклимата, которые установлены по критериям

оптимального теплового и функционального состояния человека. В соответствии с временем года и категорией тяжести работ определены оптимальные величины показателей микроклимата согласно требованиям и приведены в таблице 6.4, а допустимые величины показателей микроклимата на рабочих местах производственных помещений приведены в таблице 6.5.

Таблица 6.4 – Оптимальные величины показателей микроклимата на рабочих местах

Период года	Категория работ	Температура воздуха, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	Категория 1а	23-25	40-60	0,1
Теплый	Категория 1а	20-22	40-60	0,1

Таблица 6.5 – Допустимые величины показателей микроклимата на рабочих местах

Период года	Категория работ	Температура воздуха, °С		Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
		Ниже оптимальных не более	Выше оптимальных не более		Ниже оптимальных не более	Выше оптимальных не более
Холодный	Категория 1а	20-21,9	24,1-25	15-75	0,1	0,1
Теплый	Категория 1а	21-22,9	25,1-28	15-75	0,1	0,2

В зимнее время в помещении предусмотрена система отопления. Она обеспечивает достаточное, постоянное и равномерное нагревание воздуха. В соответствии с характеристикой помещения определен расход свежего воздуха приведен в таблице 6.6.

Таблица 6.6 – Расход свежего воздуха

Характеристика помещения	Объемный расход подаваемого в помещение свежего воздуха, м ³ /на одного человека в час
Объем до 20 м ³ на человека	Не менее 30
20...40 м ³ на человека	Не менее 20
Более 40 м ³ на человека	Естественная вентиляция

Для поддержания параметров микроклимата в диапазоне оптимальных на рабочем месте применяются следующие мероприятия: устройство систем вентиляции, кондиционирование воздуха и отопление помещения.

6.3.1.4 Электромагнитное излучение

Персональный компьютер и периферийные устройства являются потенциальными источниками электромагнитного излучения. Электромагнитные излучения оказывают негативное влияние на сердечно-сосудистую, нервную и эндокринную систему, а также могут привести к онкологическим заболеваниям.

Рабочие места с персональными электронными вычислительными машинами должны быть оборудованы защитным занулением; подача электрического тока в помещение должна осуществляться от отдельного независимого источника питания; необходима изоляция токопроводящих частей и ее непрерывный контроль; должны быть предусмотрены защитное отключение, предупредительная сигнализация и блокировка. Во избежание негативного воздействия от электромагнитного излучения необходимо следовать основным нормам. Рекомендуемые требования к уровням электромагнитных полей на рабочих местах, оборудованных ПВМ представлены в таблице 6.7.

Таблица 6.7 – Временные допустимые уровни ЭМП, создаваемых ПЭВМ на рабочих местах

Наименование параметров		ВДУ ЭМП
Напряженность электрического поля	В диапазоне частот 5 Гц – 2 кГц	25 В/м
	В диапазоне частот 2 кГц – 400 кГц	2,5 В/м
Плотность магнитного поля	В диапазоне частот 5 Гц – 2 кГц	250 нТл
	В диапазоне частот 2 кГц – 400 кГц	25 нТл
Напряженность электростатического поля		15 Кв/М

Для снижения воздействия электромагнитного излучения применяют следующие меры:

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

6.3.2 Анализ опасных факторов

6.3.2.1 Электробезопасность

Электробезопасность – это система организационных и технических мероприятий и средств, обеспечивающих защиту от вредного и опасного воздействия электрического тока, электрической дуги, электромагнитного поля и статического электричества.

Электрический ток представляет собой скрытый тип опасности, т.к. его трудно определить в токо- и нетоковедущих частях оборудования, которые являются хорошими проводниками электричества. Смертельно опасным для жизни человека считают ток, величина которого превышает 0,05 А, ток менее 0,05 А – безопасен (до 1000 В). Персональные электронные вычислительные машины и периферийные устройства являются потенциальными источниками опасности поражения человека электрическим током. В процессе работы оператора с персональным компьютером возможен удар током при соприкосновении с токоведущими частями оборудования.

Рабочие места с ПЭВМ должны быть оборудованы защитным занулением; подача электрического тока в помещение должна осуществляться от отдельного независимого источника питания; необходима изоляция токопроводящих частей и ее непрерывный контроль; должны быть предусмотрены защитное отключение, предупредительная сигнализация и блокировка. Помещение, в котором расположено рабочее место, относится к категории без повышенной опасности, и соответствует установленным условиям согласно с:

1. напряжение питающей сети 220 В, 50 Гц;
2. относительная влажность воздуха 50%;
3. средняя температура около 24°C;
4. наличие непроводящего полового покрытия.

Организационные мероприятия для защиты от поражения электрическим током:

1. перед началом работы оператор должен убедиться в отсутствии висящих проводов, в целостности провода питания, а также в отсутствии видимых токоведущих частей;
2. оператору запрещается прикасаться к задней панели системного блока при включенном питании;
3. оператору запрещается самостоятельное вскрытие и ремонт оборудования;
4. все работы по устранению неисправностей должен проводит квалифицированный персонал с исправным инструментом;
5. оператор должен не допускать попадание влаги на поверхность устройств.

6.3.2.2 Экологическая безопасность

В результате осуществления научно-технического исследования была разработана модель автоматизированной системы управления складскими помещениями. Рассмотрим влияние данного проекта на окружающую среду.

Масштабирование и увеличение мощностей производства находится в

прямой зависимости от состояния энергетики, развитие которой оказывает значительное влияние на природную среду, неся ответственность за различные виды загрязнений воздуха, воды, литосферы, а также являясь мажоритарным потребителем энергетических ресурсов, который определяют уровень их добычи.

Снижение уровня загрязнения можно осуществить путем усовершенствования оборудования, которое производит электроэнергию, применяя более экономичные и результативные технологии, а также использования новейших методов получения электроэнергии. Помимо прочего, необходимо осуществить внедрения современных методов обезвреживания и способов очистки отходов производства. Одноплатные микро-ЭВМ «Raspberry Pi», которые могут быть использованы в установке могут быть утилизированы, так как не содержат токсичных материалов. Для безопасной утилизации с точки зрения окружающей среды, а также удаления устаревших устройств необходимо обратиться к компании фирмы Raspberry, которая имеет необходимые сертификаты для утилизации и удаления лома электронного оборудования.

6.3.2.3 Безопасность в чрезвычайных ситуациях, пожарная безопасность

Под пожарной безопасностью понимается состояние объекта, при котором исключается возможность пожара, а в случае его возникновения предотвращается воздействие на людей опасных факторов пожара и обеспечивается защита материальных ценностей.

Источниками зажигания на рабочем месте оператора автоматизированной системы управления складом могут быть электронные схемы от электронных вычислительных машин, приборы, применяемые для технического обслуживания, устройства электропитания, где в результате различных нарушений образуются перегретые элементы, электрические искры и дуги, способные вызвать загорания горючих материалов.

Для операторской установлена категория пожарной опасности Д.

Основными задачами пожарной профилактики являются: предотвращение пожаров и возгорания, локализация возникших пожаров, защита людей и материальных ценностей, тушение пожара.

Мероприятия по пожарной профилактике разделяются на организационные, технические, эксплуатационные и режимные.

Организационные мероприятия предусматривают правильную эксплуатацию оборудования, правильное содержание зданий и территорий, противопожарный инструктаж рабочих и служащих, обучение производственного персонала правилам противопожарной безопасности, издание инструкций, плакатов, наличие плана эвакуации.

К техническим мероприятиям относятся: соблюдение противопожарных правил, норм при проектировании зданий, при устройстве электропроводов и оборудования, отопления, вентиляции, освещения, правильное размещение оборудования. Необходимо предусмотреть ряд мер, направленных на обеспечение тушения пожара: обеспечить подъезды к зданию; обесточивание электрических кабелей; наличие пожарных щитов и ящиков с песком в коридорах; наличие гидрантов с пожарными рукавами; телефонная связь с пожарной охраной; огнетушители: химический пенный ОХП-10 и углекислотный ОУ-2.

Заключение по разделу «Социальная ответственность»

В данной главе были рассмотрены вопросы организации рабочего места оператора АСУ ТП, проведен анализ основных вредных и опасных факторов, характерных для работы с проектируемой установкой, а также рассмотрены вопросы защиты окружающей среды.

Ввиду высокой надежности и безопасности проектируемой системы управления, которые обеспечиваются дистанционной передачей показаний основных физических величин на экран оператора, оператор, осуществляющий контроль за корректной работой системы, может осуществлять свою деятельность в специально оборудованной операторской, а не непосредственно на объекте. Также благодаря непрерывному контролю

показаний основных физических величин возможно аварийное отключение установки, что позволит избежать поломок и своевременно провести ремонтные работы.

Заключение

В ходе выполнения выпускной квалификационной работы была реализованная система управления складом, разработана виртуальная модель склада, написано программное обеспечение для их взаимодействия, так же была разработана структурная схема автоматизации и чертежи ключевых звеньев склада.

Структурная схема была выполнена по ГОСТ 21.208.2013 о автоматизации. Данная схема дает необходимое представление о системе, ее работе, составе оборудования, схеме взаимодействия звеньев автоматизации.

Для автоматизации склада, операторного контроля за системой и управления был выбран датчик UR4 UHF и управляющее устройство Raspberry Pi.

В ходе выполнения данной работы был разработан алгоритм взаимодействия всех узлов установки, а также листинг кода на языке программирования C# на платформе .NET. Для хранения информации используется таблица базы данных usql lite, все данные в которой хранятся в зашифрованном виде.

Результатом данной работы является работающий проект в программе Factory I/O осуществляющий получение, сортировку, отгрузку грузов согласно разработанному алгоритму. Взаимодействие SCADA системы со складом осуществляется посредством OPC-DA сервера.

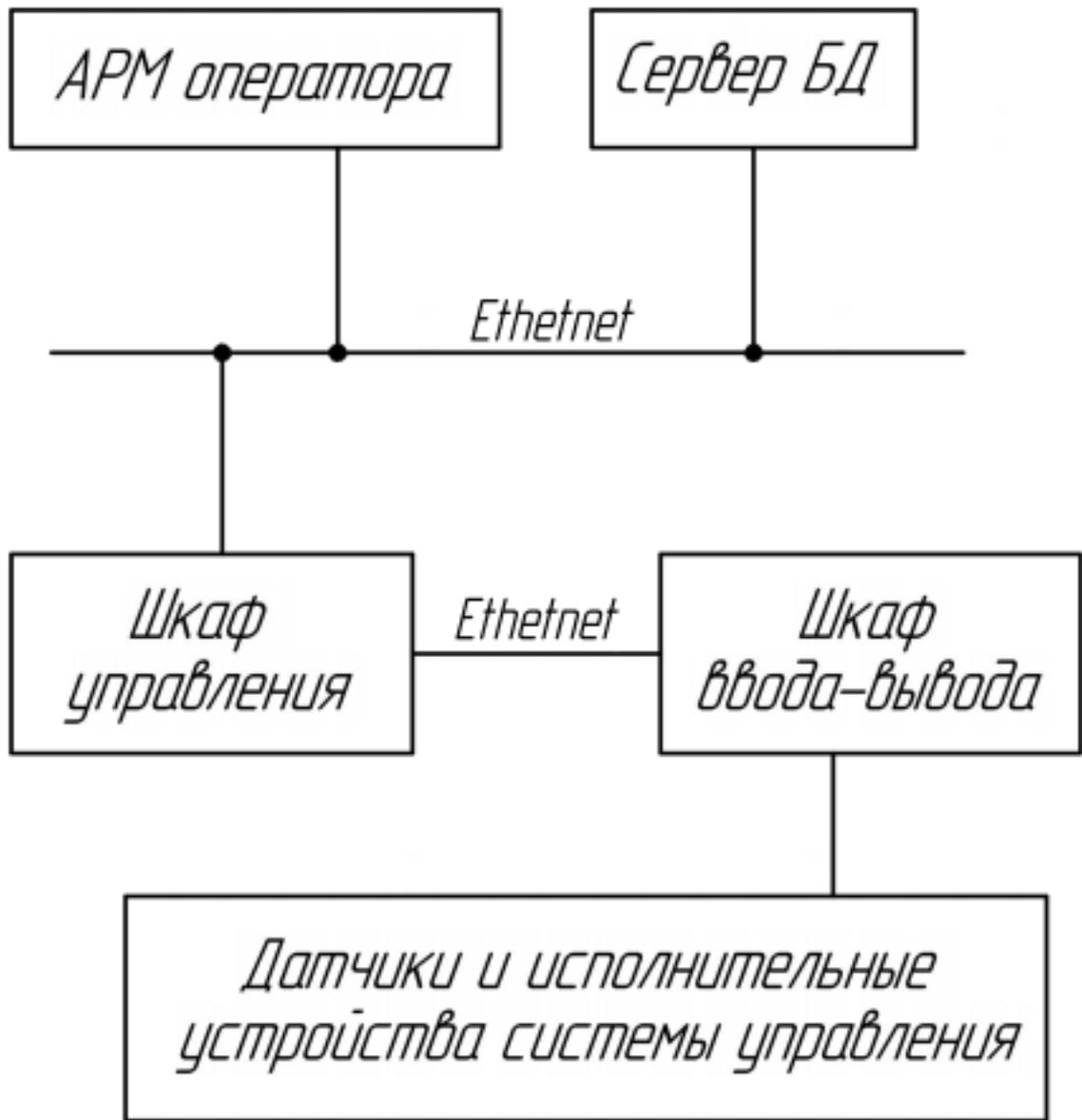
Список литературы

1. Руководство. Simple-Scada [электронный ресурс]. – Режим доступа: <https://simple-scada.com/help/manual/index.html>
2. Классификация wms систем [электронный ресурс]. – Режим доступа: <http://studfiles.ru/>
3. Технологии считывания информации RFID и NFC [электронный ресурс]. – Режим доступа: <https://cleverence.ru/>
4. OPC сервера [электронный ресурс]. – Режим доступа: <http://elibrary.ru/>
5. Информационное общество: современное состояние и перспективы развития [электронный ресурс]. – Режим доступа: <http://kubsau.ru/>
6. Анализ WMS-систем для автоматизации бизнес-процессов [электронный ресурс]. – Режим доступа: <https://habr.com/>
7. Габитов, Руслан Фаритович диссертация на тему информационные технологии автоматизации [электронный ресурс]. – Режим доступа: <http://dlib.rsl.ru/>
8. Сбор и обработка информации [электронный ресурс]. – Режим доступа: <http://sp-in.ru/>
9. WMS как инструмент повышения эффективности складской логистики организации [электронный ресурс]. – Режим доступа: <http://elibrary.ru/>
10. Комплексные (интегрированные) системы обеспечения безопасности [электронный ресурс]. – Режим доступа: <http://studentlibrary.ru/>
11. Автоматизация технологических процессов с применением программируемых логических контроллеров [электронный ресурс]. – Режим доступа: <http://elibrary.ru/>
12. Программируемые логические контроллеры, принцип действия [электронный ресурс]. – Режим доступа: <https://helpiks.org/>
13. Программно-технические комплексы автоматизированных систем управления [электронный ресурс]. – Режим доступа: <http://elibrary.ru/>
14. Классификация WMS систем [электронный ресурс]. – Режим доступа: <https://tadviser.ru/>
15. ГОСТ Р 51904-2002. Программное обеспечение встроенных систем. Общие требования к разработке и документированию.
16. ГОСТ 34602-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы. Москва: Изд-во стандартов, 1989. – 12 с.
17. МЭК 61131-3-2016. Контроллеры программируемые. Языки программирования. Москва: Изд-во стандартов, 2016.
18. ГОСТ 24.104-85. Автоматизированные системы управления. Общие

- требования. Москва: Изд-во стандартов, 2013.
19. ГОСТ 21.408-2013. Система проектной документации для строительства (СПДС). Правила выполнения рабочей документации автоматизации технологических процессов (с Поправками). Москва: Изд-во стандартов, 2013.
 20. ГОСТ 19.002-80. Схемы алгоритмов и программ. Правила выполнения. Москва: Изд-во стандартов, 2080.
 21. ГОСТ 21.208-2013. Система проектной документации для строительства (СПДС). Автоматизация технологических процессов. Обозначения условные приборов и средств автоматизации в схемах.
 22. Трудовой кодекс Российской Федерации" от 30.12.2001 N 197-ФЗ (ред. от 31.12.2014).
 23. ГОСТ 12.2.032-78. Система стандартов безопасности труда.
 24. СанПиН 2.2.1/2.1.1.1278 – 03. Гигиенические требования к естественному, искусственному и совмещённому освещению жилых и общественных зданий. М.: Минздрав России, 2003.
 25. СП 52.13330.2011 Свод правил. Естественное и искусственное освещение.
 26. СН 2.2.4/2.1.8.562 – 96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории застройки.
 27. СанПиН 2.2.4.548 – 96. Гигиенические требования к микроклимату производственных помещений. М.: Минздрав России, 1997.
 28. ГОСТ 12.1.005-88. Общие санитарно-гигиенические требования к воздуху рабочей зоны.
 29. СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы.
 30. ГОСТ 12.1.038-82. Система стандартов безопасности труда. Электробезопасность. Предельно допустимые значения напряжений прикосновения и токов.
 31. ГОСТ 12.1.004–91 ССБТ. Пожарная безопасность. Общие требования.85
 32. ГОСТ 12.1.003-83. Система стандартов безопасности труда. Шум. Общие требование безопасности.
 33. Белов С.В. Безопасность жизнедеятельности и защита окружающей среды: учебник для вузов. – М.: Изд-во Юрайт, 2013. – 671с.
 34. Федеральный закон "Об отходах производства и потребления" от 24.06.1998 N 89-ФЗ, "Об отходах производства и потребления».

Приложение А
(обязательное)
Структурная схема

ФЮРА.420602.01



ФЮРА.420602.01

Изм.	Лист	№ докум.	Подп.	Дата
Разраб.		Кудинов М.С.		
Проб.		Тугов И.А.		
Т.контр.				
Н.контр.				
Утв.		Курганов В.В.		

Лит.	Масса	Масштаб
у		
Лист 1		Листов

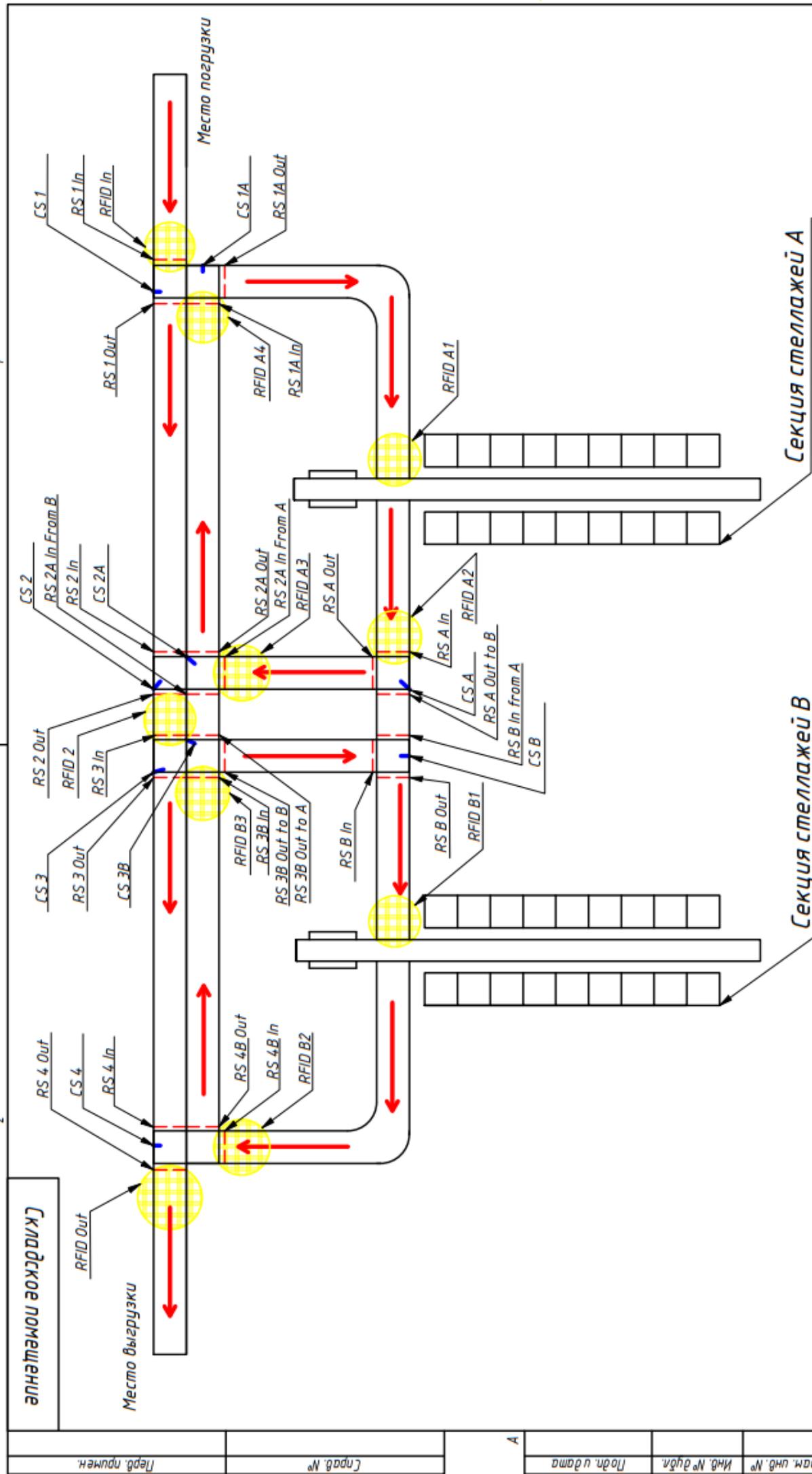
Структурная схема

ТПУ ИШИТР ОАР гр. 8Т7А

Приложение Б

(обязательное)

Чертеж вид сверху – датчики



- * направление движения конвейера
- RFID датчик
- Датчик движения (RS)
- Поворотный датчик (CS)

Все названия в выносках полностью соответствуют названиям датчиков на модели Factory IO

Инд. № подл.	Подп. и дата	Взам. инд. №	Инд. инд. №	Инд. № дубл.	Подп. и дата
Справ. №					
Перв. примен.					

Складское помещение		Лит.	Масса	Масштаб
Склад СП-1, вид сверху.				1:1
Датчики		Лист	1	Листов
				1

Приложение В
(Обязательное)
Вид сверху – приводы

Приложение Г

(обязательное)

Листинг кода тегов датчиков склада

1. MemoryBit loadrcA4 = MemoryMap.Instance.GetBit("Load RC A4", MemoryType.Output);
2. MemoryBit unloadrcA5 = MemoryMap.Instance.GetBit("Unload RC A5", MemoryType.Output);
3. MemoryInt targetPositionA = MemoryMap.Instance.GetInt("Target Position A", MemoryType.Output);
4. MemoryBit forksLeftA = MemoryMap.Instance.GetBit("Forks Left A", MemoryType.Output);
5. MemoryBit liftA = MemoryMap.Instance.GetBit("Lift A", MemoryType.Output);
6. MemoryBit forksRightA = MemoryMap.Instance.GetBit("Forks Right A", MemoryType.Output);
7. MemoryBit loadrcB3 = MemoryMap.Instance.GetBit("Load RC B3", MemoryType.Output);
8. MemoryBit unloadrcB4 = MemoryMap.Instance.GetBit("Unload RC B4", MemoryType.Output);
9. MemoryInt targetPositionB = MemoryMap.Instance.GetInt("Target Position B", MemoryType.Output);
10. MemoryBit forksLeftB = MemoryMap.Instance.GetBit("Forks Left B", MemoryType.Output);
11. MemoryBit liftB = MemoryMap.Instance.GetBit("Lift B", MemoryType.Output);
12. MemoryBit forksRightB = MemoryMap.Instance.GetBit("Forks Right B", MemoryType.Output);
13. MemoryBit rc11 = MemoryMap.Instance.GetBit("RC (4m) 1.1", MemoryType.Output);
14. MemoryBit rc12 = MemoryMap.Instance.GetBit("RC (6m) 1.2", MemoryType.Output);
15. MemoryBit rc13 = MemoryMap.Instance.GetBit("RC (2m) 1.3", MemoryType.Output);
16. MemoryBit rc14 = MemoryMap.Instance.GetBit("RC (2m) 1.4", MemoryType.Output);
17. MemoryBit rc15 = MemoryMap.Instance.GetBit("RC (6m) 1.5", MemoryType.Output);
18. MemoryBit rc16 = MemoryMap.Instance.GetBit("RC (2m) 1.6", MemoryType.Output);
19. MemoryBit rc17 = MemoryMap.Instance.GetBit("RC (4m) 1.7", MemoryType.Output);
20. MemoryBit rcA1 = MemoryMap.Instance.GetBit("RC A1", MemoryType.Output);
21. MemoryBit rcA2 = MemoryMap.Instance.GetBit("Curved RC A2", MemoryType.Output);
22. MemoryBit rcA3 = MemoryMap.Instance.GetBit("RC A3", MemoryType.Output);
23. MemoryBit rcA6 = MemoryMap.Instance.GetBit("RC A6", MemoryType.Output);
24. MemoryBit rcA7 = MemoryMap.Instance.GetBit("RC A7", MemoryType.Output);
25. MemoryBit rcA8 = MemoryMap.Instance.GetBit("RC A8", MemoryType.Output);
26. MemoryBit rcA9 = MemoryMap.Instance.GetBit("RC A9", MemoryType.Output);
27. MemoryBit rcA10 = MemoryMap.Instance.GetBit("RC A10", MemoryType.Output);
28. MemoryBit rcB1 = MemoryMap.Instance.GetBit("RC B1", MemoryType.Output);
29. MemoryBit rcB2 = MemoryMap.Instance.GetBit("RC B2", MemoryType.Output);
30. MemoryBit rcB5 = MemoryMap.Instance.GetBit("RC B5", MemoryType.Output);
31. MemoryBit rcB6 = MemoryMap.Instance.GetBit("Curved RC B6", MemoryType.Output);
32. MemoryBit rcB7 = MemoryMap.Instance.GetBit("RC B7", MemoryType.Output);
33. MemoryBit rcB8 = MemoryMap.Instance.GetBit("RC B8", MemoryType.Output);
34. MemoryBit rcB9 = MemoryMap.Instance.GetBit("RC B9", MemoryType.Output);
35. MemoryBit rcB10 = MemoryMap.Instance.GetBit("RC B10", MemoryType.Output);
36. MemoryBit ct1p = MemoryMap.Instance.GetBit("CT 1 (+)", MemoryType.Output);
37. MemoryBit ct1n = MemoryMap.Instance.GetBit("CT 1 (-)", MemoryType.Output);
38. MemoryBit ct2p = MemoryMap.Instance.GetBit("CT 2 (+)", MemoryType.Output);
39. MemoryBit ct2n = MemoryMap.Instance.GetBit("CT 2 (-)", MemoryType.Output);
40. MemoryBit ct3p = MemoryMap.Instance.GetBit("CT 3 (+)", MemoryType.Output);
41. MemoryBit ct3n = MemoryMap.Instance.GetBit("CT 3 (-)", MemoryType.Output);
42. MemoryBit ct4p = MemoryMap.Instance.GetBit("CT 4 (+)", MemoryType.Output);
43. MemoryBit ct4n = MemoryMap.Instance.GetBit("CT 4 (-)", MemoryType.Output);
44. MemoryBit ct1Ap = MemoryMap.Instance.GetBit("CT 1A (+)", MemoryType.Output);
45. MemoryBit ct1An = MemoryMap.Instance.GetBit("CT 1A (-)", MemoryType.Output);
46. MemoryBit ct2Ap = MemoryMap.Instance.GetBit("CT 2A (+)", MemoryType.Output);
47. MemoryBit ct2An = MemoryMap.Instance.GetBit("CT 2A (-)", MemoryType.Output);
48. MemoryBit ct3Bp = MemoryMap.Instance.GetBit("CT 3B (+)", MemoryType.Output);
49. MemoryBit ct3Bn = MemoryMap.Instance.GetBit("CT 3B (-)", MemoryType.Output);
50. MemoryBit ct4Bp = MemoryMap.Instance.GetBit("CT 4B (+)", MemoryType.Output);
51. MemoryBit ct4Bn = MemoryMap.Instance.GetBit("CT 4B (-)", MemoryType.Output);
52. MemoryBit ctAp = MemoryMap.Instance.GetBit("CT A (+)", MemoryType.Output);
53. MemoryBit ctAn = MemoryMap.Instance.GetBit("CT A (-)", MemoryType.Output);
54. MemoryBit ctBp = MemoryMap.Instance.GetBit("CT B (+)", MemoryType.Output);
55. MemoryBit ctBn = MemoryMap.Instance.GetBit("CT B (-)", MemoryType.Output);
56. MemoryBit ct1L = MemoryMap.Instance.GetBit("CT 1 Left", MemoryType.Output);
57. MemoryBit ct1R = MemoryMap.Instance.GetBit("CT 1 Right", MemoryType.Output);
58. MemoryBit ct2L = MemoryMap.Instance.GetBit("CT 2 Left", MemoryType.Output);
59. MemoryBit ct2R = MemoryMap.Instance.GetBit("CT 2 Right", MemoryType.Output);

```

60. MemoryBit ct3L = MemoryMap.Instance.GetBit("CT 3 Left", MemoryType.Output);
61. MemoryBit ct3R = MemoryMap.Instance.GetBit("CT 3 Right", MemoryType.Output);
62. MemoryBit ct4L = MemoryMap.Instance.GetBit("CT 4 Left", MemoryType.Output);
63. MemoryBit ct4R = MemoryMap.Instance.GetBit("CT 4 Right", MemoryType.Output);
64. MemoryBit ct1AL = MemoryMap.Instance.GetBit("CT 1A Left", MemoryType.Output);
65. MemoryBit ct1AR = MemoryMap.Instance.GetBit("CT 1A Right", MemoryType.Output);
66. MemoryBit ct2AL = MemoryMap.Instance.GetBit("CT 2A Left", MemoryType.Output);
67. MemoryBit ct2AR = MemoryMap.Instance.GetBit("CT 2A Right", MemoryType.Output);
68. MemoryBit ct3BL = MemoryMap.Instance.GetBit("CT 3B Left", MemoryType.Output);
69. MemoryBit ct3BR = MemoryMap.Instance.GetBit("CT 3B Right", MemoryType.Output);
70. MemoryBit ct4BL = MemoryMap.Instance.GetBit("CT 4B Left", MemoryType.Output);
71. MemoryBit ct4BR = MemoryMap.Instance.GetBit("CT 4B Right", MemoryType.Output);
72. MemoryBit ctAL = MemoryMap.Instance.GetBit("CT A Left", MemoryType.Output);
73. MemoryBit ctAR = MemoryMap.Instance.GetBit("CT A Right", MemoryType.Output);
74. MemoryBit ctBL = MemoryMap.Instance.GetBit("CT B Left", MemoryType.Output);
75. MemoryBit ctBR = MemoryMap.Instance.GetBit("CT B Right", MemoryType.Output);
76. MemoryBit rStop1 = MemoryMap.Instance.GetBit("StopR 1 Out", MemoryType.Input);
77. MemoryBit rStop1A = MemoryMap.Instance.GetBit("StopR 1A In", MemoryType.Input);
78. MemoryBit rStop2 = MemoryMap.Instance.GetBit("StopR 2 In", MemoryType.Input);
79. MemoryBit rStop2Aout = MemoryMap.Instance.GetBit("StopR 2A Out", MemoryType.Input);
80. MemoryBit rStop2AfromB = MemoryMap.Instance.GetBit("StopR 2A In from B",
MemoryType.Input);
81. MemoryBit rStop2AfromA = MemoryMap.Instance.GetBit("StopR 2A In from A",
MemoryType.Input);
82. MemoryBit rStopA = MemoryMap.Instance.GetBit("StopR A", MemoryType.Input);
83. MemoryBit rStopB = MemoryMap.Instance.GetBit("StopR B", MemoryType.Input);
84. MemoryBit rStop3in = MemoryMap.Instance.GetBit("StopR 3 In", MemoryType.Input);
85. MemoryBit rStop3out = MemoryMap.Instance.GetBit("StopR 3 Out", MemoryType.Input);
86. MemoryBit rStop3Bin = MemoryMap.Instance.GetBit("StopR 3B In", MemoryType.Input);
87. MemoryBit rStop3BtoB = MemoryMap.Instance.GetBit("StopR 3B Out to B", MemoryType.Input);
88. MemoryBit rStop3BtoA = MemoryMap.Instance.GetBit("StopR 3B Out to A", MemoryType.Input);
89. MemoryBit rStop4B = MemoryMap.Instance.GetBit("StopR 4B", MemoryType.Input);
90. MemoryBit movingXA = MemoryMap.Instance.GetBit("Moving X A", MemoryType.Input);
91. MemoryBit movingZA = MemoryMap.Instance.GetBit("Moving Z A", MemoryType.Input);
92. MemoryBit atLoadA = MemoryMap.Instance.GetBit("At Load A", MemoryType.Input);
93. MemoryBit atLeftA = MemoryMap.Instance.GetBit("At Left A", MemoryType.Input);
94. MemoryBit atMiddleA = MemoryMap.Instance.GetBit("At Middle A", MemoryType.Input);
95. MemoryBit atRightA = MemoryMap.Instance.GetBit("At Right A", MemoryType.Input);
96. MemoryBit movingXB = MemoryMap.Instance.GetBit("Moving X B", MemoryType.Input);
97. MemoryBit movingZB = MemoryMap.Instance.GetBit("Moving Z B", MemoryType.Input);
98. MemoryBit atLoadB = MemoryMap.Instance.GetBit("At Load B", MemoryType.Input);
99. MemoryBit atLeftB = MemoryMap.Instance.GetBit("At Left B", MemoryType.Input);
100. MemoryBit atMiddleB = MemoryMap.Instance.GetBit("At Middle B", MemoryType.Input);
101. MemoryBit atRightB = MemoryMap.Instance.GetBit("At Right B", MemoryType.Input);
102. MemoryBit cs1 = MemoryMap.Instance.GetBit("CS 1", MemoryType.Input);
103. MemoryBit cs2 = MemoryMap.Instance.GetBit("CS 2", MemoryType.Input);
104. MemoryBit cs3 = MemoryMap.Instance.GetBit("CS 3", MemoryType.Input);
105. MemoryBit cs4 = MemoryMap.Instance.GetBit("CS 4", MemoryType.Input);
106. MemoryBit cs1A = MemoryMap.Instance.GetBit("CS 1A", MemoryType.Input);
107. MemoryBit cs2A = MemoryMap.Instance.GetBit("CS 2A", MemoryType.Input);
108. MemoryBit cs3B = MemoryMap.Instance.GetBit("CS 3B", MemoryType.Input);
109. MemoryBit cs4B = MemoryMap.Instance.GetBit("CS 4B", MemoryType.Input);
110. MemoryBit csA = MemoryMap.Instance.GetBit("CS A", MemoryType.Input);
111. MemoryBit csB = MemoryMap.Instance.GetBit("CS B", MemoryType.Input);
112. MemoryBit rs1In = MemoryMap.Instance.GetBit("RS 1 In", MemoryType.Input);
113. MemoryBit rs1Out = MemoryMap.Instance.GetBit("RS 1 Out", MemoryType.Input);
114. MemoryBit rs1AIn = MemoryMap.Instance.GetBit("RS 1A In", MemoryType.Input);
115. MemoryBit rs1AOut = MemoryMap.Instance.GetBit("RS 1A Out", MemoryType.Input);
116. MemoryBit rs2In = MemoryMap.Instance.GetBit("RS 2 In", MemoryType.Input);
117. MemoryBit rs2Out = MemoryMap.Instance.GetBit("RS 2 Out", MemoryType.Input);
118. MemoryBit rs2AfromA = MemoryMap.Instance.GetBit("RS 2A In From A", MemoryType.Input);

```

```

119. MemoryBit rs2AfromB = MemoryMap.Instance.GetBit("RS 2A In From B", MemoryType.Input);
120. MemoryBit rs2AOut = MemoryMap.Instance.GetBit("RS 2A Out", MemoryType.Input);
121. MemoryBit rs3In = MemoryMap.Instance.GetBit("RS 3 In", MemoryType.Input);
122. MemoryBit rs3Out = MemoryMap.Instance.GetBit("RS 3 Out", MemoryType.Input);
123. MemoryBit rs3AtoA = MemoryMap.Instance.GetBit("RS 3B Out to A", MemoryType.Input);
124. MemoryBit rs3AtoB = MemoryMap.Instance.GetBit("RS 3B Out to B", MemoryType.Input);
125. MemoryBit rs3Bin = MemoryMap.Instance.GetBit("RS 3B In", MemoryType.Input);
126. MemoryBit rs4In = MemoryMap.Instance.GetBit("RS 4 In", MemoryType.Input);
127. MemoryBit rs4Out = MemoryMap.Instance.GetBit("RS 4 Out", MemoryType.Input);
128. MemoryBit rs4Bin = MemoryMap.Instance.GetBit("RS 4B In", MemoryType.Input);
129. MemoryBit rs4BOut = MemoryMap.Instance.GetBit("RS 4B Out", MemoryType.Input);
130. MemoryBit rsAIn = MemoryMap.Instance.GetBit("RS A In", MemoryType.Input);
131. MemoryBit rsAOut = MemoryMap.Instance.GetBit("RS A Out", MemoryType.Input);
132. MemoryBit rsAtoB = MemoryMap.Instance.GetBit("RS A Out to B", MemoryType.Input);
133. MemoryBit rsBin = MemoryMap.Instance.GetBit("RS B In", MemoryType.Input);
134. MemoryBit rsBfromA = MemoryMap.Instance.GetBit("RS B In from A", MemoryType.Input);
135. MemoryBit rsBOut = MemoryMap.Instance.GetBit("RS B Out", MemoryType.Input);
136. MemoryInt rfidInCommand = MemoryMap.Instance.GetInt("RFID In Command", MemoryType.Output);
137. MemoryBit rfidInExecuteCommand = MemoryMap.Instance.GetBit("RFID In Execute Command",
MemoryType.Output);
138. MemoryInt rfidInMemoryIndex = MemoryMap.Instance.GetInt("RFID In Memory Index",
MemoryType.Output);
139. MemoryInt rfidInWriteData = MemoryMap.Instance.GetInt("RFID In Write Data", MemoryType.Output);
140. MemoryInt rfidInCommandID = MemoryMap.Instance.GetInt("RFID In Command ID", MemoryType.Input);
141. MemoryInt rfidInStatus = MemoryMap.Instance.GetInt("RFID In Status", MemoryType.Input);
142. MemoryInt rfidInReadData = MemoryMap.Instance.GetInt("RFID In Read Data", MemoryType.Input);
143. MemoryInt rfid2Command = MemoryMap.Instance.GetInt("RFID 2 Command", MemoryType.Output);
144. MemoryBit rfid2ExecuteCommand = MemoryMap.Instance.GetBit("RFID 2 Execute Command",
MemoryType.Output);
145. MemoryInt rfid2MemoryIndex = MemoryMap.Instance.GetInt("RFID 2 Memory Index",
MemoryType.Output);
146. MemoryInt rfid2WriteData = MemoryMap.Instance.GetInt("RFID 2 Write Data", MemoryType.Output);
147. MemoryInt rfid2CommandID = MemoryMap.Instance.GetInt("RFID 2 Command ID", MemoryType.Input);
148. MemoryInt rfid2Status = MemoryMap.Instance.GetInt("RFID 2 Status", MemoryType.Input);
149. MemoryInt rfid2ReadData = MemoryMap.Instance.GetInt("RFID 2 Read Data", MemoryType.Input);
150. MemoryInt rfidOutCommand = MemoryMap.Instance.GetInt("RFID Out Command", MemoryType.Output);
151. MemoryBit rfidOutExecuteCommand = MemoryMap.Instance.GetBit("RFID Out Execute Command",
MemoryType.Output);
152. MemoryInt rfidOutMemoryIndex = MemoryMap.Instance.GetInt("RFID Out Memory Index",
MemoryType.Output);
153. MemoryInt rfidOutWriteData = MemoryMap.Instance.GetInt("RFID Out Write Data", MemoryType.Output);
154. MemoryInt rfidOutCommandID = MemoryMap.Instance.GetInt("RFID Out Command ID",
MemoryType.Input);
155. MemoryInt rfidOutStatus = MemoryMap.Instance.GetInt("RFID Out Status", MemoryType.Input);
156. MemoryInt rfidOutReadData = MemoryMap.Instance.GetInt("RFID Out Read Data", MemoryType.Input);
157. MemoryInt rfidA1Command = MemoryMap.Instance.GetInt("RFID A1 Command", MemoryType.Output);
158. MemoryBit rfidA1ExecuteCommand = MemoryMap.Instance.GetBit("RFID A1 Execute Command",
MemoryType.Output);
159. MemoryInt rfidA1MemoryIndex = MemoryMap.Instance.GetInt("RFID A1 Memory Index",
MemoryType.Output);
160. MemoryInt rfidA1WriteData = MemoryMap.Instance.GetInt("RFID A1 Write Data", MemoryType.Output);
161. MemoryInt rfidA1CommandID = MemoryMap.Instance.GetInt("RFID A1 Command ID",
MemoryType.Input);
162. MemoryInt rfidA1Status = MemoryMap.Instance.GetInt("RFID A1 Status", MemoryType.Input);
163. MemoryInt rfidA1ReadData = MemoryMap.Instance.GetInt("RFID A1 Read Data", MemoryType.Input);
164. MemoryInt rfidA2Command = MemoryMap.Instance.GetInt("RFID A2 Command", MemoryType.Output);

```

```

165. MemoryBit rfidA2ExecuteCommand = MemoryMap.Instance.GetBit("RFID A2 Execute Command",
MemoryType.Output);
166. MemoryInt rfidA2MemoryIndex = MemoryMap.Instance.GetInt("RFID A2 Memory Index",
MemoryType.Output);
167. MemoryInt rfidA2WriteData = MemoryMap.Instance.GetInt("RFID A2 Write Data", MemoryType.Output);
168. MemoryInt rfidA2CommandID = MemoryMap.Instance.GetInt("RFID A2 Command ID",
MemoryType.Input);
169. MemoryInt rfidA2Status = MemoryMap.Instance.GetInt("RFID A2 Status", MemoryType.Input);
170. MemoryInt rfidA2ReadData = MemoryMap.Instance.GetInt("RFID A2 Read Data", MemoryType.Input);
171. MemoryInt rfidA3Command = MemoryMap.Instance.GetInt("RFID A3 Command", MemoryType.Output);
172. MemoryBit rfidA3ExecuteCommand = MemoryMap.Instance.GetBit("RFID A3 Execute Command",
MemoryType.Output);
173. MemoryInt rfidA3MemoryIndex = MemoryMap.Instance.GetInt("RFID A3 Memory Index",
MemoryType.Output);
174. MemoryInt rfidA3WriteData = MemoryMap.Instance.GetInt("RFID A3 Write Data", MemoryType.Output);
175. MemoryInt rfidA3CommandID = MemoryMap.Instance.GetInt("RFID A3 Command ID",
MemoryType.Input);
176. MemoryInt rfidA3Status = MemoryMap.Instance.GetInt("RFID A3 Status", MemoryType.Input);
177. MemoryInt rfidA3ReadData = MemoryMap.Instance.GetInt("RFID A3 Read Data", MemoryType.Input);
178. MemoryInt rfidA4Command = MemoryMap.Instance.GetInt("RFID A4 Command", MemoryType.Output);
179. MemoryBit rfidA4ExecuteCommand = MemoryMap.Instance.GetBit("RFID A4 Execute Command",
MemoryType.Output);
180. MemoryInt rfidA4MemoryIndex = MemoryMap.Instance.GetInt("RFID A4 Memory Index",
MemoryType.Output);
181. MemoryInt rfidA4WriteData = MemoryMap.Instance.GetInt("RFID A4 Write Data", MemoryType.Output);
182. MemoryInt rfidA4CommandID = MemoryMap.Instance.GetInt("RFID A4 Command ID",
MemoryType.Input);
183. MemoryInt rfidA4Status = MemoryMap.Instance.GetInt("RFID A4 Status", MemoryType.Input);
184. MemoryInt rfidA4ReadData = MemoryMap.Instance.GetInt("RFID A4 Read Data", MemoryType.Input);
185. MemoryInt rfidB1Command = MemoryMap.Instance.GetInt("RFID B1 Command", MemoryType.Output);
186. MemoryBit rfidB1ExecuteCommand = MemoryMap.Instance.GetBit("RFID B1 Execute Command",
MemoryType.Output);
187. MemoryInt rfidB1MemoryIndex = MemoryMap.Instance.GetInt("RFID B1 Memory Index",
MemoryType.Output);
188. MemoryInt rfidB1WriteData = MemoryMap.Instance.GetInt("RFID B1 Write Data", MemoryType.Output);
189. MemoryInt rfidB1CommandID = MemoryMap.Instance.GetInt("RFID B1 Command ID",
MemoryType.Input);
190. MemoryInt rfidB1Status = MemoryMap.Instance.GetInt("RFID B1 Status", MemoryType.Input);
191. MemoryInt rfidB1ReadData = MemoryMap.Instance.GetInt("RFID B1 Read Data", MemoryType.Input);
192. MemoryInt rfidB2Command = MemoryMap.Instance.GetInt("RFID B2 Command", MemoryType.Output);
193. MemoryBit rfidB2ExecuteCommand = MemoryMap.Instance.GetBit("RFID B2 Execute Command",
MemoryType.Output);
194. MemoryInt rfidB2MemoryIndex = MemoryMap.Instance.GetInt("RFID B2 Memory Index",
MemoryType.Output);
195. MemoryInt rfidB2WriteData = MemoryMap.Instance.GetInt("RFID B2 Write Data", MemoryType.Output);
196. MemoryInt rfidB2CommandID = MemoryMap.Instance.GetInt("RFID B2 Command ID",
MemoryType.Input);
197. MemoryInt rfidB2Status = MemoryMap.Instance.GetInt("RFID B2 Status", MemoryType.Input);
198. MemoryInt rfidB2ReadData = MemoryMap.Instance.GetInt("RFID B2 Read Data", MemoryType.Input);
199. MemoryInt rfidB3Command = MemoryMap.Instance.GetInt("RFID B3 Command", MemoryType.Output);
200. MemoryBit rfidB3ExecuteCommand = MemoryMap.Instance.GetBit("RFID B3 Execute Command",
MemoryType.Output);
201. MemoryInt rfidB3MemoryIndex = MemoryMap.Instance.GetInt("RFID B3 Memory Index",
MemoryType.Output);
202. MemoryInt rfidB3WriteData = MemoryMap.Instance.GetInt("RFID B3 Write Data", MemoryType.Output);
203. MemoryInt rfidB3CommandID = MemoryMap.Instance.GetInt("RFID B3 Command ID",
MemoryType.Input);
204. MemoryInt rfidB3Status = MemoryMap.Instance.GetInt("RFID B3 Status", MemoryType.Input);
205. MemoryInt rfidB3ReadData = MemoryMap.Instance.GetInt("RFID B3 Read Data", MemoryType.Input);

```

Приложение Д

(обязательное)

Листинг кода программы управления складом

```

using EngineIO;

namespace Controllers
{
    public class AutomatedWarehouse : Controller
    {
        MemoryBit entryConveyor = MemoryMap.Instance.GetBit("Entry Conveyor",
MemoryType.Output);
        MemoryBit loadConveyor = MemoryMap.Instance.GetBit("Load Conveyor",
MemoryType.Output);
        MemoryInt targetPosition = MemoryMap.Instance.GetInt("Target Position",
MemoryType.Output);
        MemoryBit forksLeft = MemoryMap.Instance.GetBit("Forks Left",
MemoryType.Output);
        MemoryBit lift = MemoryMap.Instance.GetBit("Lift", MemoryType.Output);
        MemoryBit forksRight = MemoryMap.Instance.GetBit("Forks Right",
MemoryType.Output);

        MemoryBit movingX = MemoryMap.Instance.GetBit("Moving X", MemoryType.Input);
        MemoryBit movingZ = MemoryMap.Instance.GetBit("Moving Z", MemoryType.Input);
        MemoryBit atLoad = MemoryMap.Instance.GetBit("At Load", MemoryType.Input);
        MemoryBit atLeft = MemoryMap.Instance.GetBit("At Left", MemoryType.Input);
        MemoryBit atMiddle = MemoryMap.Instance.GetBit("At Middle", MemoryType.Input);
        MemoryBit atRight = MemoryMap.Instance.GetBit("At Right", MemoryType.Input);

        FTRIG ftMovXZ = new FTRIG();
        RTRIG rtAtLoad = new RTRIG();
        RTRIG rtAtLeft = new RTRIG();
        FTRIG ftMovZ = new FTRIG();
        RTRIG rtAtMiddle = new RTRIG();
        RTRIG rtAtRight = new RTRIG();

        int currentPos = 0;

        State state = State.State0;

        public AutomatedWarehouse()
        {
            state = State.State1;

            entryConveyor.Value = false;
            loadConveyor.Value = false;

            entryConveyor.Value = false;
            loadConveyor.Value = false;
            targetPosition.Value = 55;
            forksLeft.Value = false;
            lift.Value = false;
            forksRight.Value = false;
        }

        public override void Execute(int elapsedMilliseconds)
        {
            ftMovXZ.CLK(movingX.Value || movingZ.Value);
            rtAtLoad.CLK(!atLoad.Value);
            rtAtLeft.CLK(atLeft.Value);
            ftMovZ.CLK(movingZ.Value);
        }
    }
}

```

```

rtAtMiddle.CLK(atMiddle.Value);
rtAtRight.CLK(atRight.Value);

if (state == State.State0)
{
    targetPosition.Value = 55;

    if (ftMovXZ.Q)
        state = State.State1;
}
else if (state == State.State1)
{
    //Waiting for load...

    if (!atLoad.Value)
        state = State.State2;
}
else if (state == State.State2)
{
    forksLeft.Value = true;

    if (rtAtLeft.Q)
        state = State.State3;
}
else if (state == State.State3)
{
    lift.Value = true;

    if (ftMovZ.Q)
        state = State.State4;
}
else if (state == State.State4)
{
    forksLeft.Value = false;

    if (rtAtMiddle.Q)
        state = State.State5;
}
else if (state == State.State5)
{

```

```

targetPosition.Value =
++currentPos;

    state = State.State6;
}
else if (state == State.State6)
{
    //Moving to destination...

    if (ftMovXZ.Q)
        state = State.State7;
}
else if (state == State.State7)
{
    forksRight.Value = true;

    if (rtAtRight.Q)
        state = State.State8;
}
else if (state == State.State8)
{
    lift.Value = false;

    if (ftMovZ.Q)
        state = State.State9;
}
else if (state == State.State9)
{
    forksRight.Value = false;

    if (rtAtMiddle.Q)
        state = State.State0;
}

    entryConveyor.Value =
loadConveyor.Value = atLoad.Value;
}
}
}

```