

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет

Кафедра

Вычислительной техники

«Вычислительная техника»

Специальность: 09.05.01 «Применение и эксплуатация
автоматизированных систем специального
назначения»

Специализация: «Эксплуатация вычислительных машин,
комплексов, систем и сетей специального назначения»

ДИПЛОМНЫЙ ПРОЕКТ

на тему

СИСТЕМА УПРАВЛЕНИЯ ПОТОЧНОЙ
ЛИНИЕЙ ПРОИЗВОДСТВА

Студент

М.Е. Еськов
(подпись, дата)

Еськов Максим Михайлович
(ФИО полностью)

Руководитель

Д.А. Тарасов
(подпись, дата)

Д.А. Тарасов
(фамилия, инициалы)

Нормоконтролёр

А.С. Бычков
(подпись, дата)

А.С. Бычков
(фамилия, инициалы)

Рецензент

главный редактор
(должность, место работы)

А.В. Васильков
(подпись, дата)

А.В. Васильков
(фамилия, инициалы)

Работа допущена к защите (протокол зас. кафедры от 22.05.26 г. № 14)

Заведующий кафедрой

М.А. Митрохин
(подпись)

М.А. Митрохин
(фамилия, инициалы)

Работа защищена с отметкой удовлетворительно (протокол заседания ГЭК от

10 июня 2026 № 6)

Секретарь ГЭК

Е.И. Гурин

Е.И. Гурин

Пенза 2026

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет

Вычислительной техники

Кафедра

«Вычислительная техника»

«Утверждаю»

Заведующий кафедрой

М.А. Митрохин

« 11 » 02 2026г.

**ЗАДАНИЕ
НА ДИПЛОМНЫЙ ПРОЕКТ**

1. Студент Еськов Максим Михайлович гр. 21ВОЭ1 факультета вычислительной техники

2. Тема работы «Система управления поточной линией производства»

Тема утверждена приказом ПГУ № 122/о от " 11 " февраля 2026 г.

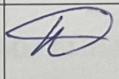
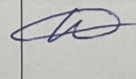
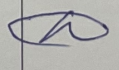
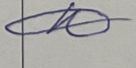
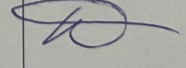
3. Руководитель дипломной работы к.т.н., доцент кафедры ВТ ПГУ Тарасов Д.А.

4. Задание на работу (назначение разработки, исходные данные и т.п).
Разработать систему управления электроприводами горизонтального и наклонного конвейеров для передвижения сыпучих грузов. На пульте управления разместить две кнопки: «Пуск» и «Стоп». При пуске первым включается наклонный конвейер, через 10 секунд — горизонтальный. Наклонный конвейер в нормальном режиме работает 3 минуты, при этом за 20 секунд до его остановки первым выключается горизонтальный. При кратковременном нажатии «Стоп» первым останавливается горизонтальный конвейер, через 20 секунд — наклонный; при удержании «Стоп» более 2 секунд оба останавливаются мгновенно. Разработка ведётся в OwenLogic на языке функциональных блоков, также создать панель оператора (HMI).

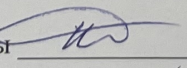
5. Перечень подлежащих разработке вопросов

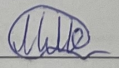
Требуется произвести анализ предметной области: изучить существующие системы управления конвейерными линиями для сыпучих грузов, выявить их достоинства и недостатки в части алгоритмов пуска и остановки, а также средств визуализации (HMI). Также необходимо изучить среду программирования OwenLogic, язык функциональных блоков (FBD), принципы построения временных диаграмм и логики управления электроприводами, методы реализации аварийных остановок и защиты оборудования. Разработать алгоритм работы линии в нормальном и аварийном режимах, создать программу управления в OwenLogic и спроектировать панель оператора для отображения состояния конвейеров и приёма команд.

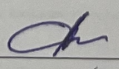
6. Календарный график выполнения работы

№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1	Анализ темы, аналогов, постановка задачи	126	11.02.26 – 02.03.26	
2	Выбор инструментов разработки выбор элементной базы (для аппаратной части)	215	02.03.26 – 01.04.26	
3	Получение и анализ результатов	286	01.04.26 – 29.04.26	
4	Описание архитектуры приложения	185	29.04.26 – 16.05.26	
5	Оформление отчёта	52	16.05.26 – 21.05.26	
	Общий объём часов	864		

Дата выдачи задания "11 " февраля 2026 г.

Руководитель работы 
(подпись, дата)

Задание к исполнению принял студент 
(подпись, дата)

Работу к защите допустить
Декан факультета  А.Р. Фионов
05.06.2026 (подпись, дата)

Реферат

Дипломный проект содержит 85 листов, 108 рисунков, 13 таблиц, 7 источников, 3 приложения.

СИСТЕМА УПРАВЛЕНИЯ ПОТОЧНОЙ ЛИНИЕЙ ПРОИЗВОДСТВА.

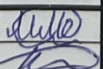
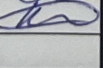
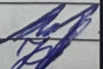
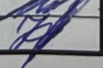
Цель работы – разработать программу, позволяющую автоматизировать организацию управления поточной линией производства на базе ПЛК ОВЕН ПР200 и панели оператора ОВЕН СП270.

Объектом разработки является управляющая программа, которая должна обеспечивать переключение состояний работы двигателей после нажатия кнопок управления. Система должна обладать графическим пользовательским интерфейсом на панели оператора.

Разработка системы производилась в OwenLogic. Логика системы был создана языке функциональных блоков.

По итогам разработки системы была реализована системы выполняющая в полном объеме поставленную задачу и взаимодействие панели оператора с контроллером.

Созданная система может быть внедрена в любые производственные организации.

					ПГУ 1.090501.03.001 ПЗ		
Изм	Лист	№ докум.	Подпись	Дата			
Разраб.		Еськов М.М.		22.05.20			
Пров.		Тарасов Д.А.		28.05.20			
Н. Контр.		Бычков А.С.		28.05.20			
Утв.		Митрохин М.А.		28.05.20			
Система управления поточной линией производства Пояснительная записка					Литера	Лист	Листов
						4	85
					ФВТ, каф. ВТ гр. 21В0Э1		

Содержание

Перечень сокращений и обозначений.....	8
Введение.....	9
1 Система управления поточной линией производства	10
1.1 Анализ предметной области.....	10
1.1.1 Обзор аналогов	11
1.1.2 Анализ требований к разработке	14
1.1.3 Выбор архитектуры программы.....	15
1.1.4 Выбор языка программирования	16
1.1.5 Выбор среды разработки	16
1.2 Проектирование и разработка	30
1.2.1 Конфигурирование ПЛК ОВЕН ПР200.....	30
1.2.2 Разработка управляющей программы	35
1.2.3 Конфигурирование панели оператора ОВЕН СП270	44
1.3 Тестирование результатов разработки	49
1.3.1 Тестирование и отладка программы в редакторе OWEN Logic	49
1.3.2 Тестирование и отладка интерфейса панели оператора ОВЕН СП270	52
1.3.3 Тестирование проекта на стенде целевой платформы	53
2 Охрана труда	62
2.1 Анализ негативных факторов, воздействующих на оператора ПЭВМ и способы их устранения.....	62

2.1.1 Оборудование рабочего места	62
2.1.2 Освещенность рабочего места	64
2.1.3 Электробезопасность	65
2.1.4 Пожарная безопасность	65
2.1.5 Режим труда и отдыха.....	66
2.2 Меры безопасности при работе с ПЛК ОВЕН ПР200 и панелью оператора ОВЕН СП270	68
3 Экономика	69
3.1 Цель и задачи экономического обоснования.....	69
3.2 Расчет затрат на разработку системы	69
3.2.1 Трудоемкость разработки	69
3.2.2 Расчет заработной платы разработчика	70
3.2.3 Стоимость оборудования и комплектующих	70
3.2.4 Прочие затраты	71
3.2.5 Полная себестоимость разработки.....	71
3.3 Капитальные затраты на внедрение.....	71
3.4 Расчет эксплуатационных расходов	72
3.5 Оценка экономической эффективности	72
3.5.3 Расчёт годовой экономии	73
3.5.4 Чистый годовой экономический эффект.....	74
3.5.5 Срок окупаемости проекта	75
3.6 Анализ чувствительности	75

3.7 Вывод по экономической части	75
Заключение	76
Список использованных источников	77
Приложение А.....	79
Приложение А.1 – UML–диаграмма деятельности для пользователя ...	79
Приложение А.2 – UML–диаграмма деятельности администратора	80
Приложение А.3 – UML–диаграмма использования для пользователя .	81
Приложение А.4 – UML–диаграмма сценариев для администратора....	82
Приложение Б – Схема программ.....	83
Приложение В – Презентация.....	86

Перечень сокращений и обозначений

В настоящей работе применяют следующие сокращения и обозначения

ЖКИ – жидкокристаллический индикатор

ОВЕН СП270 – панель оператора фирмы ОВЕН модель СП270

ПЗУ – постоянное запоминающее устройство

ПК – персональный компьютер

ПЛК – программируемый логический контроллер

ПЛК ОВЕН ПР200 – программируемый логический контроллер фирмы ОВЕН модель ПР200

ПЭВМ – персональная электронно–вычислительная машина

FBD (Function Block Diagram) – графический язык диаграмм функциональных блоков, который предназначен для визуального программирования

IEC 61131 – международный стандарт, описывающий языки программирования для программируемых логических контроллеров

Modbus RTU – коммуникационный протокол, основан на архитектуре ведущий–ведомый

OWEN Logic – среда программирования для создания алгоритмов работы программируемых логических контроллеров

RS-485 (англ. Recommended Standard 485) – стандарт физического уровня для промышленных сетей

SCADA (Supervisory Control And Data Acquisition) – программный пакет, предназначенный для разработки или обеспечения работы в реальном времени систем сбора, обработки, отображения и архивирования информации об объекте мониторинга или управления

USB (Universal Serial Bus) – последовательный интерфейс для подключения периферийных устройств к вычислительной технике

					ПГУ 1.090501.03 ПЗ	Лист
						8
Изм.	Лист	№ докум.	Подпись	Дата		

Введение

Потребность в организации контроля доступа в помещениях возникает почти в любой отрасли человеческой деятельности. Принимаемые решения должны учитывать конкретные условия и доступное оборудование, а на производстве — опираться на уровень автоматизации и существующие ресурсы. Разрешить данную задачу позволяет автоматизация контроля доступа, которая одновременно усиливает защитные меры и сокращает затраты на содержание персонала.

При внедрении автоматизации структура управляющей системы значительно усложняется и включает комбинацию разнообразного оборудования. Взаимодействие с оператором также качественно меняется: команды подаются, а данные отображаются посредством множества устройств. В этом контексте человек становится неотъемлемым элементом контура управления, а само взаимодействие именуется человеко–машинным интерфейсом.

В случае, когда такой интерфейс ориентирован на управление автоматизированным технологическим процессом, он получает название SCADA–система (диспетчерское управление и сбор данных). На практике это понятие трактуется значительно шире, поскольку функционал современных SCADA–систем выходит далеко за рамки буквального перевода термина.

В контуре управления SCADA–система получает информацию от программируемых логических контроллеров. Те, в свою очередь, опрашивают различные составляющие системы.

1 Система управления поточной линией производства

1.1 Анализ предметной области

Системы управления поточной линией производства активно используется на предприятиях и заводах с целью упрощения перемещения грузов по конвейеру. Разрабатываемая программа для ПЛК (программируемого логического контролера) OWEN PR200 должна обеспечить простое и понятное управление поточной линией производства. Эффективность и качество работы напрямую зависит от надежности системы управления.

Проект реализован для автоматизации управления поточной линией производства и обеспечения эффективного использования проекта работниками предприятий. Управление реализовано с помощью панели оператора, в котором реализован интуитивно понятный интерфейс для эффективной работы сотрудников предприятия.

У программы присутствуют особенности, позволяющие ей быть востребованной и способной к конкуренции:

- Легкость в использовании;
- безотказность работы;
- возможность работы без подключения к сети;
- легкость встраивания в инфраструктуру;
- обширность использования;
- возможность запоминания данных в контроллер.

Итогом разработки является система позволяющая упростить работу по перемещению грузов с использованием поточной линии производства по различным объектам. Все это позволяет получить востребованную систему.

1.1.1 Обзор аналогов

На данный момент насчитывают много различных поточных линий производства. Каждая из них имеет как достоинства так и недостатки. Наиболее распространенным является ленточный тип, которые имеет преимущество в виде высокой производительности, простоты конструкции, возможности транспортировки на большие расстояния. Также присутствуют и недостатки у данного типа, например необходимость обслуживания ленты и роликов. Следующим типом является цепной. Данный тип имеет ряд преимуществ: высокая прочность, возможность перемещения тяжелых грузов. Но у данного тип имеются и недостатки, такие как: повышенный уровень шума и износ механических элементов. Еще одним популярным типом является винтовой, у которого имеются так же ряд преимуществ: компактность конструкции, удобство транспортировки мелких материалов. Но есть у данного типа и недостатки в виде ограниченной длины транспортировки и сравнительно высокого энергопотребления. Краткий анализ представлен в таблице 1.

Таблица 1 – Анализ преимуществ и недостатков конкурентов

Производитель	Преимущество	Недостаток
Ленточный	Высокая производительность, простота конструкции, возможность транспортировки на большие расстояния	Необходимость регулярного обслуживания ленты и роликов
Цепной	Высокая прочность,	Повышенный уровень шума и износ

	возможность перемещения тяжелых грузов	механических элементов
Винтовой	Компактность конструкции, удобство транспортировки мелких материалов	Ограниченная длина транспортировки и сравнительно высокое энергопотребление

Ленточный тип поточной линии производства – это тип конвейерного оборудования, предназначенный для приема сыпучих или кусковых материалов с одного транспортера и передачи их на другой транспортер, а также для формирования штабеля или равномерного распределения материала по площади. На рисунке 1 представлен пример ленточной линии производства.

Ленточный тип может иметь различные области применения, такие как:

- Горнодобывающая промышленность;
- отвалообразование;
- металлургия;
- строительная индустрия.



Рисунок 1 – Ленточный тип

Цепной тип поточной линии производства – это тип конвейерного оборудования, в котором тяговым и несущим органом служит одна или несколько цепей, движущихся по направляющим. В отличие от ленточного перегружателя, здесь отсутствует гибкая лента, материал перемещается непосредственно по металлическому настилу или в ковшах, закреплённых на цепях. На рисунке 2 представлен пример цепного типа поточной линии производства.

Цепной тип может иметь различные сферы применения, такие как:

- Горнорудная и горнодобывающая промышленность;
- металлургия;
- литейное производство;
- цементная и строительная промышленность.



Рисунок 2 – Цепной тип

Винтовой тип поточной линии производства – это вид непрерывного транспорта, в котором перемещение материала осуществляется с помощью вращающегося винта, заключённого в жёлоб или открытого. В отличие от ленточных и цепных перегружателей, рабочий орган здесь полностью закрыт или имеет корытообразную форму, что позволяет транспортировать сыпучие, пылящие, пастообразные или даже влажные материалы с высокой

герметичностью. На рисунке 3 представлен пример винтового типа поточной линии производства.

Винтовой тип используется в различных сферах, таких как:

- Сельское хозяйство и комбикормовая промышленность;
- строительная индустрия и производство стройматериалов;
- химическая и фармацевтическая промышленность;
- пищевая промышленность;
- металлургия и горнорудная промышленность.



Рисунок 3 – Винтовой

На основе проведенного анализа был сделан вывод существуют различные виды поточных линий производства которые направлены на выполнение определенных задач в различных сферах.

1.1.2 Анализ требований к разработке

Итогом проекта является создание системы поточной линии производства на базе программируемого логического контроллера, которая предназначена для различных сфер, таких как металлургия, горнодобывающая промышленность с требованиями безопасного перемещения предметов. Разрабатываемая программа должно обеспечить высокую эффективность и

безопасность работников при работе с ней в условиях промышленности.

Требуется следующий функционал системы:

- Запуск и остановка горизонтального конвейера;
- запуск и остановка наклонного конвейера;
- последовательный запуск конвейеров;
- обеспечение безопасной остановки оборудования;
- обработка сигналов датчиков;
- световая индикация состояния системы.

Для управления разрабатываемой системой используется панель оператора с созданным на ней интерфейсом в конфигураторе.

1.1.3 Выбор архитектуры программы

Требуется создать обмен данными направленный в две стороны, обмен данными будет происходить между человеком и управляемой программой.

Реализацию можно распределить на шаги выполнения:

- Сначала программа анализирует что происходит на входах системы после чего передает информацию в выделенную память под кол-во входов;
- далее программа перебирает полученную информацию по определенному алгоритму;
- после того как информация была перебрана, готовый результат будет передан на выходные контакты контроллера;
- после прохождения последнего этапа система переходит в первоначальное состояние и начинает цикл заново.

То сколько времени будет потрачено на прохождение всех шагов будет зависеть от сложности создаваемого алгоритма. Функциональная схема работы прибора представлена на рисунке 4.

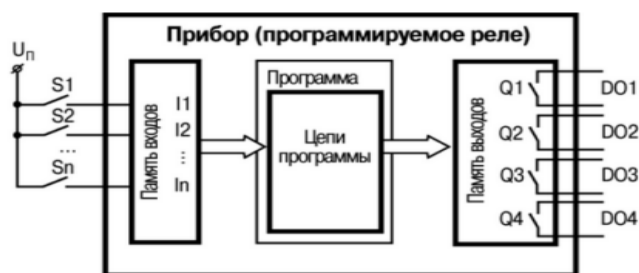


Рисунок 4 – Функциональная схема работы прибора

Обмен данными производится через панель оператора. К панели оператора по интерфейсу RS485 подключается контроллер в режиме slave. Данные передаются с использованием витой пары что обеспечивает стабильную передачу данных.

1.1.4 Выбор языка программирования

В качестве языка программирования был выбран язык функциональных блоков. Этот язык создан для программирования логических контроллеров. Удобство данного языка программирования заключается в том что, пользователю не нужно писать множество строк кода. Программирование происходит путем размещения объектов на рабочем холсте программы. Одним из главных плюсов является то что программа собирается как конструктор. Система создается путем соединения входных и выходных блоков.

1.1.5 Выбор среды разработки

Средой для разработки была выбрана среда программирования OwenLogic. В то время как для создания панели оператора используется конфигуратор СП200.

OwenLogic – это среда разработки, специализированная на создании алгоритмов для логических контроллеров и близких к ним по функциональности устройств. К их числу относятся приборы серий ПР1хх, ПР200, а также панель оператора ИПП120, выпускаемые компанией ОВЕН. Полный перечень устройств, поддерживаемых данной средой, представлен на

официальном сайте производителя .

С помощью OWEN Logic разработчик может создавать программы автоматизации, реализующие любую необходимую логику работы оборудования, и загружать их в память прибора. Программирование системы происходит с использование функциональных блоков. Это облегчает процесс написания алгоритма работы программ.

Среда разработки доступна для установки на официальном сайте, с этим можно ознакомиться на рисунке 5.

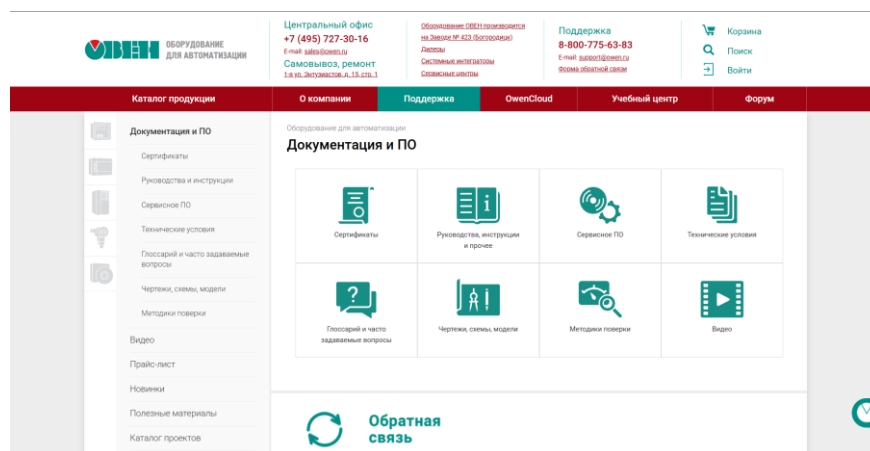


Рисунок 5 – Главная страница OWEN

На главной странице производителя помимо установки архива с программой можно ознакомиться технической документацией, информацией о производителе, а также актуальной информацией об различных сферах компании, для того что бы началась установка архива с программным обеспечением требуется нажать кнопку скачать, это показано на рисунке 6.

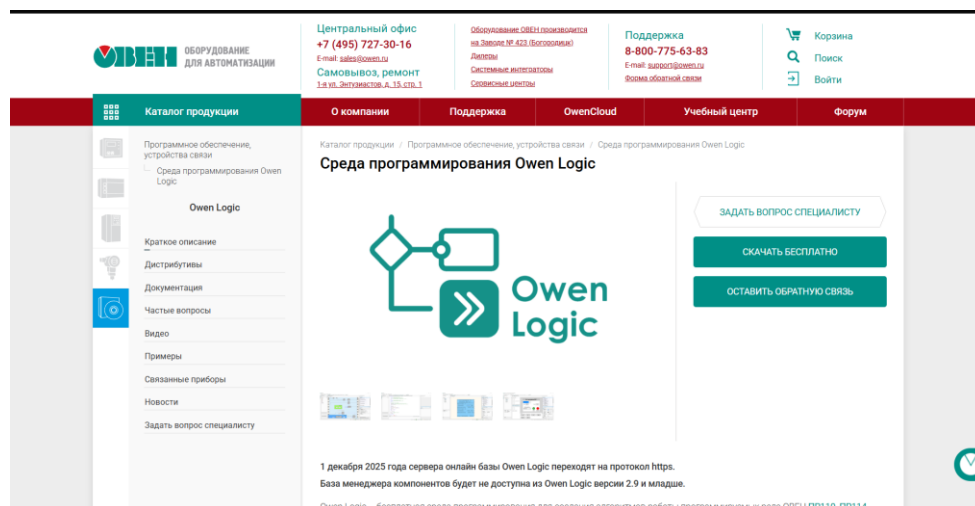


Рисунок 6 – Загрузка среды программирования на сайте

Что бы началась загрузка пользователь должен ознакомиться с лицензионным соглашением, после чего если пользователь согласен, поставить галочку и нажать кнопку скачать, после чего начнется установка архива, все это можно увидеть на рисунке 7.

☐ Я принимаю условия лицензионного соглашения

Скачать

Рисунок 7 – Подтверждение с согласия с условиями лицензионного соглашения

Как только была нажата кнопка скачать и было потрачено немного времени на установку архива можно приступить установке OwenLogic. Ознакомится с процессом установки можно на рисунке 8.

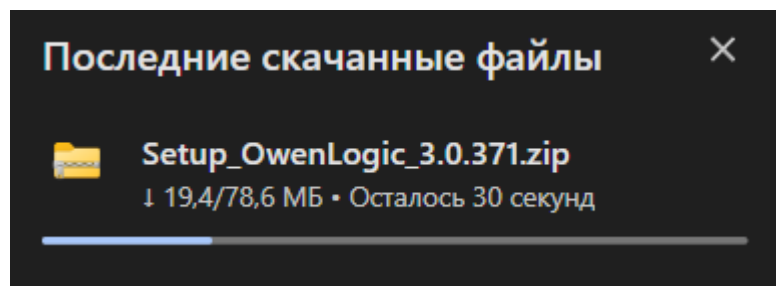


Рисунок 8 – Загрузка архива с установочными файлами

После установки архива его требуется распаковать, после распаковке архива пользователю будет предоставлен установочный exe файл с шагами установки, с началом установки можно ознакомиться на рисунке 9.

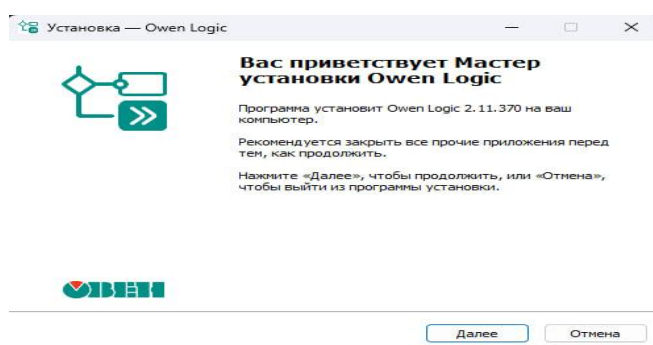


Рисунок 9 – Мастер установки OWEN LOGIC

Далее пользователю потребуется указать место для установки, сделать он это сможет с использованием кнопки обзор, после выбора места потребуется нажать кнопку далее, после чего будет установлен OwenLogic, пользователь выбирает куда будет установлена программа для дальнейшей работы с ней, процесс выбора места для установки показан на рисунке 10.

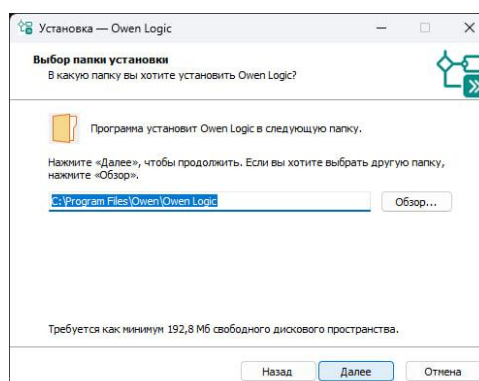


Рисунок 10 – Выбор папки для среды установки

После выбора места для установки перед пользователем будет меню, в котором пользователь может указать дополнительные настройки которые может установить в случае необходимости, доступные настройки показаны на рисунке 11.

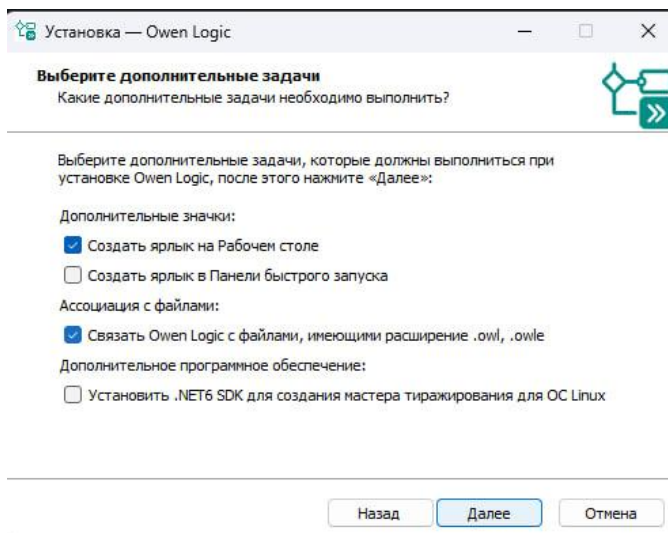


Рисунок 11 – Дополнительные настройки среды программирования

Далее пользователь сможет увидеть что будет установлено на его устройство а также путь для установки, окно с информацией показано на рисунке 12.

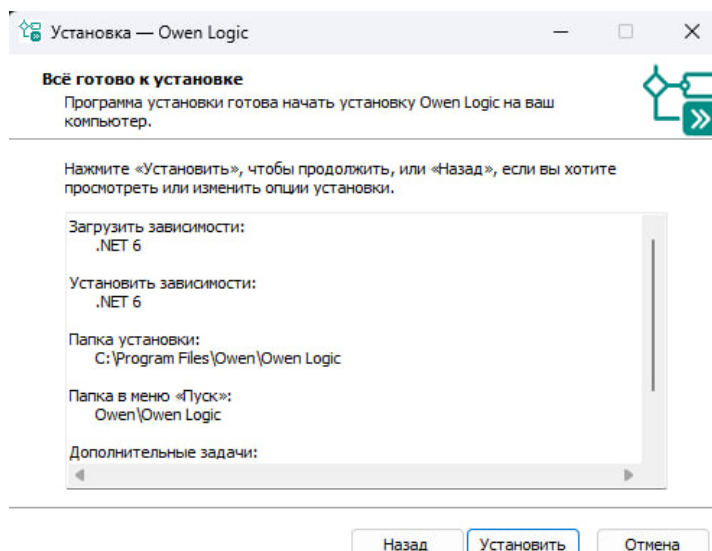


Рисунок 12 – Просмотр настроек установки

На рисунке 13 можно ознакомиться с областью работы программы OwenLogic.

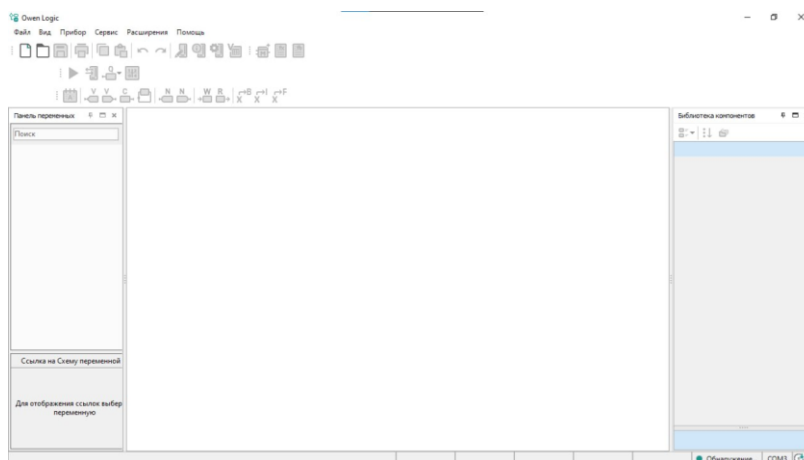


Рисунок 13 – Рабочая область среды программирования

Для того что бы начать работать в OwenLogic потребуется создать новый проект в котором можно будет создать логику для системы, все это можно увидеть на рисунке 14.

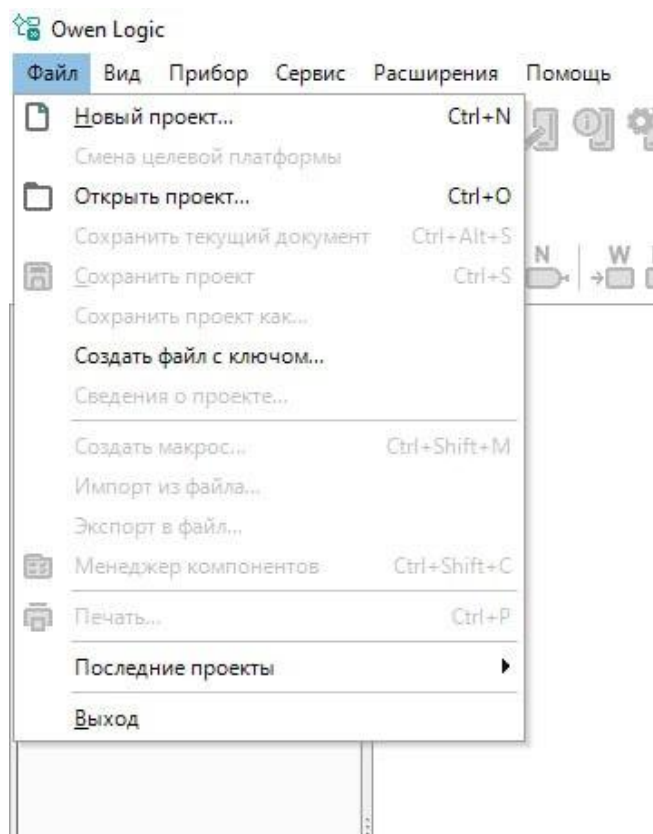


Рисунок 14 – Меню создания проекта

На рисунке 15 можно увидеть основные инструменты разработки OwenLogic.

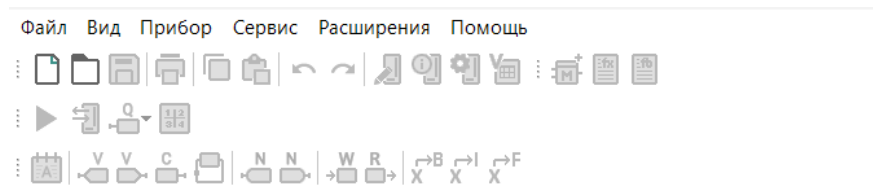


Рисунок 15 – Панель инструментов OWEN Logic

Важно указать, что как и все остальные функции программы, панель инструментов также поддается настройке, если пользователю покажет неудобной панель инструментов то он сможет создать всё для себя, всё это показано на рисунке 16.

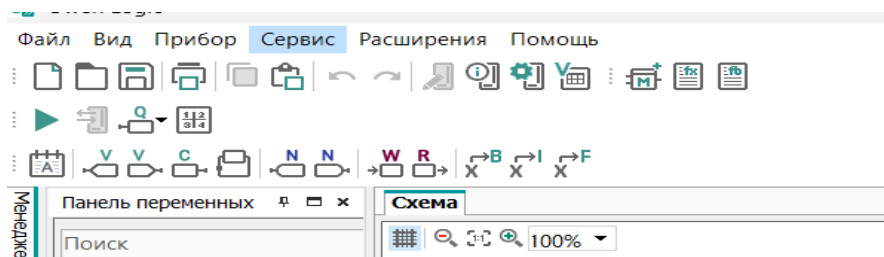


Рисунок 16 – Панель инструментов после настройки

После того как была запущена эмуляция пользователю будет дополнена панель инструментов, это можно увидеть на рисунке 17.

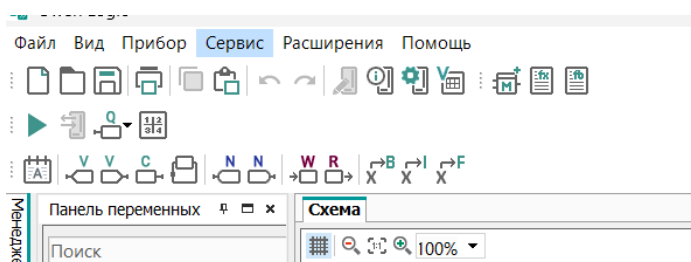


Рисунок 17 – Панель инструментов в режиме эмуляции

Для того что создать логику пользователь должен использовать рабочий холст программы на котором сможет располагать нужные ему компоненты схемы для создания логики, что показано на рисунке 18, а на рисунке 19 можно ознакомиться с измененным холстом.

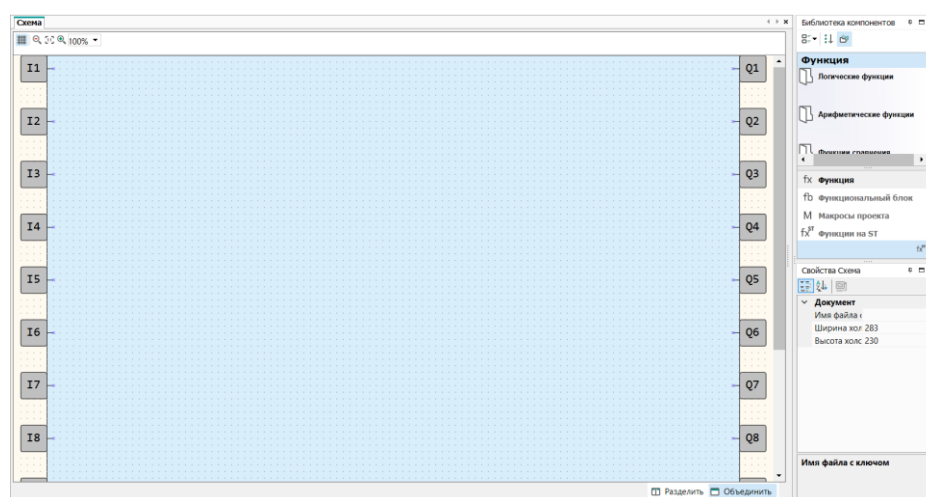


Рисунок 18 – Рабочий холст проекта среды OWEN Logic

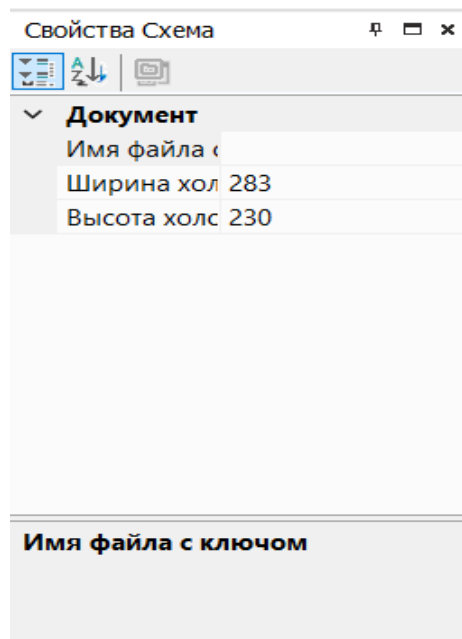


Рисунок 19 – Свойства рабочего холста проекта

Одним из первых что видит пользователь является библиотека компонентов показанная на рисунке 19, также присутствует экран в котором собраны функциональные блоки, нельзя не упомянуть про панель переменных, в ней располагаются все переменные системы, показана рисунке 20.

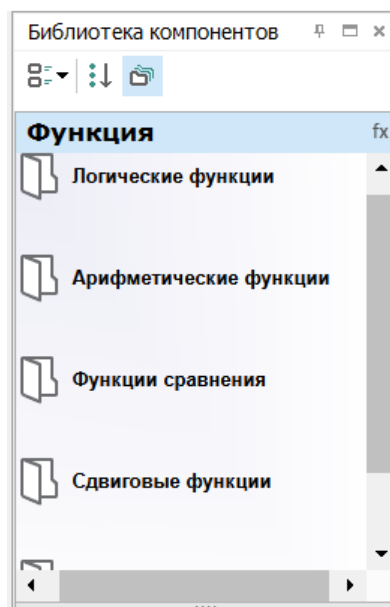


Рисунок 20 – Библиотека компонентов

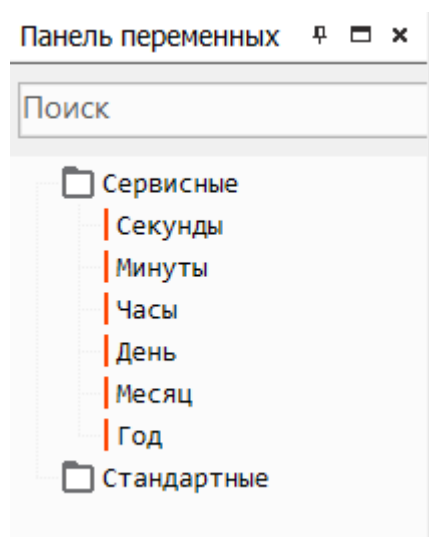


Рисунок 21 – Панель переменных

Что бы подключить контроллер к компьютеру потребуется наличие USB порта в котором потребуется изменить настройки прибора, с настройками можно ознакомиться на рисунке 22.

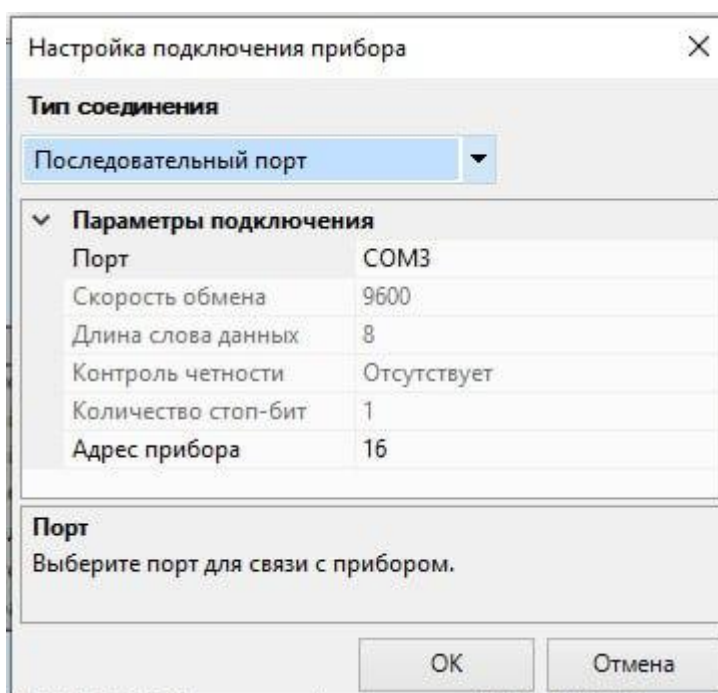


Рисунок 22 – Настройка подключения прибора

Для того что бы быстро и грамотно создавать различные системы пользователю рекомендуется использовать встроенное руководство

пользователя, что можно увидеть на рисунке 23.

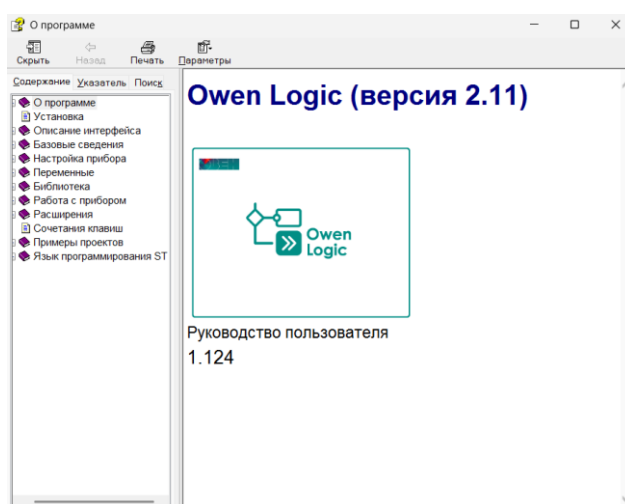


Рисунок 23 – Руководство пользователя

Что бы создать графический интерфейс системы была установлена программа конфигуратор СП200.

Конфигуратор СП200 – это ПО, которое служить для создания конфигурации панели оператора

Конфигуратор позволяет создать проект в котором можно будет реализовать создание панели оператора, сложность панели будет зависеть от самого разработчика так как программа предоставляет широкий доступ инструментов. После того как был создан проект его можно загрузить на панель оператора, на который будет передана конфигурация.

Для упрощения установки пользователю предоставлена инструкция для установки, с которой он может ознакомиться ниже. На рисунке 24 виден мастер настройки конфигуратора.

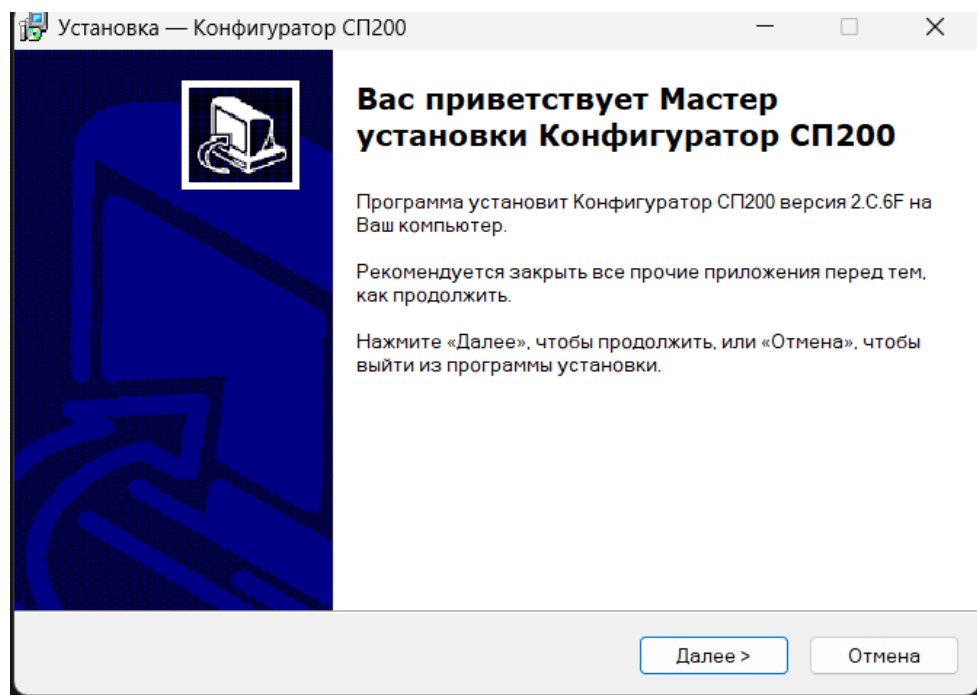


Рисунок 25 – Мастер установки конфигуратора СП200

На рисунке 26 пользователю показано окно, в котором он может выбрать нужный ему путь для установки.

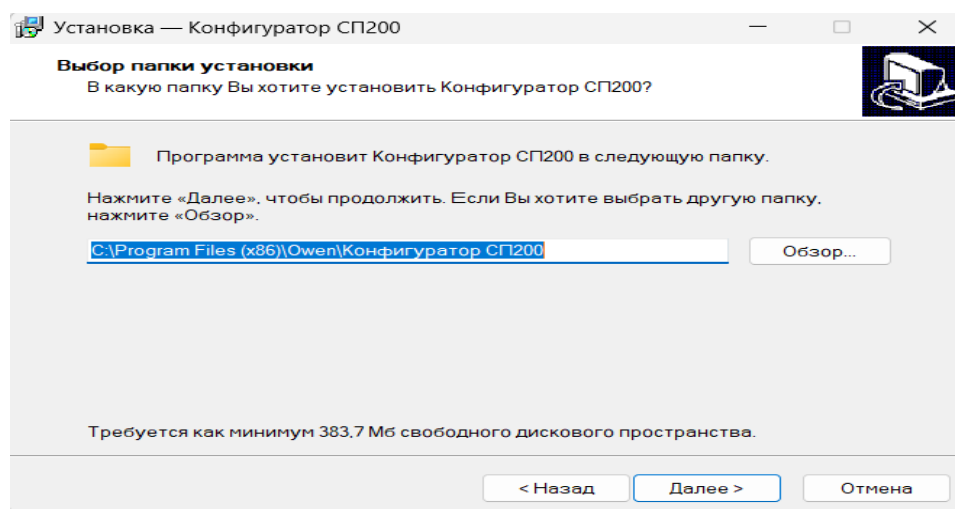


Рисунок 26 – Выбор пространства для установки конфигуратора СП200

После выбора места для установки у пользователя будет возможность создания ярлыка на рабочем столе, ознакомится с созданием можно на рисунке 27.

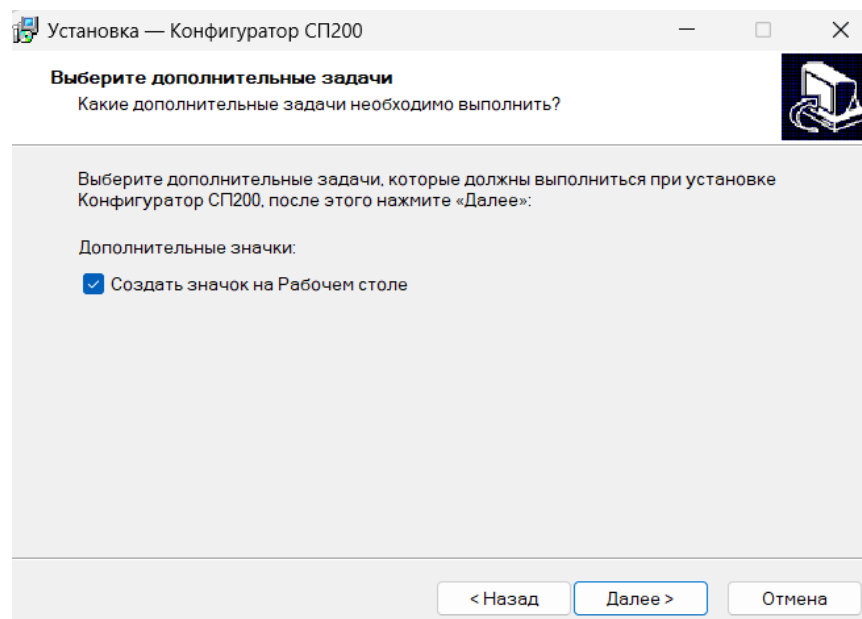


Рисунок 27 – Создание ярлыка configurатора на рабочем столе

На рисунке 28 можно увидеть выбранные пользователем настройки для configurатора.

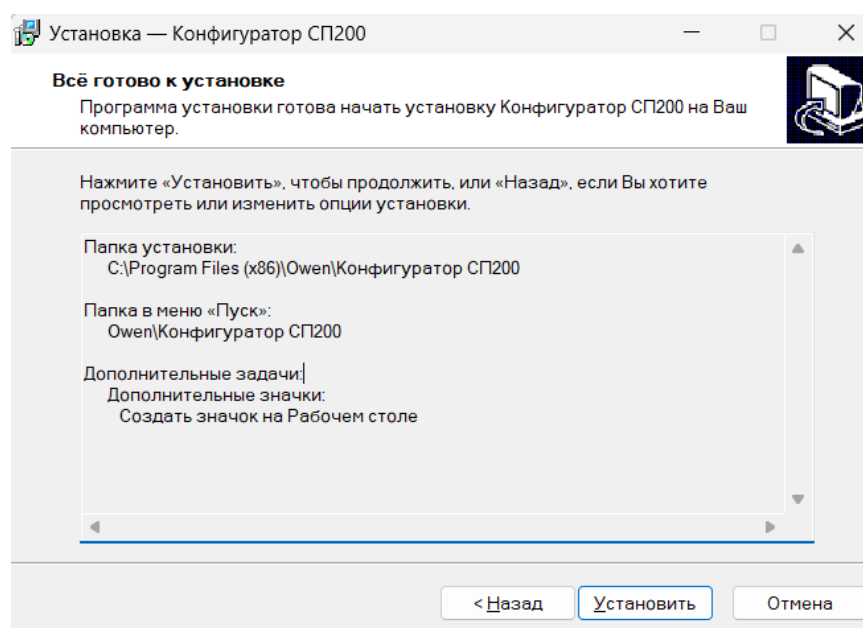


Рисунок 28 – Выбранные настройки для установки

По окончании установки перед пользователем появится ещё один выбор, при желании пользователь может запустить приложение сразу после нажатия кнопки завершить. Увидеть это можно на рисунке 29.

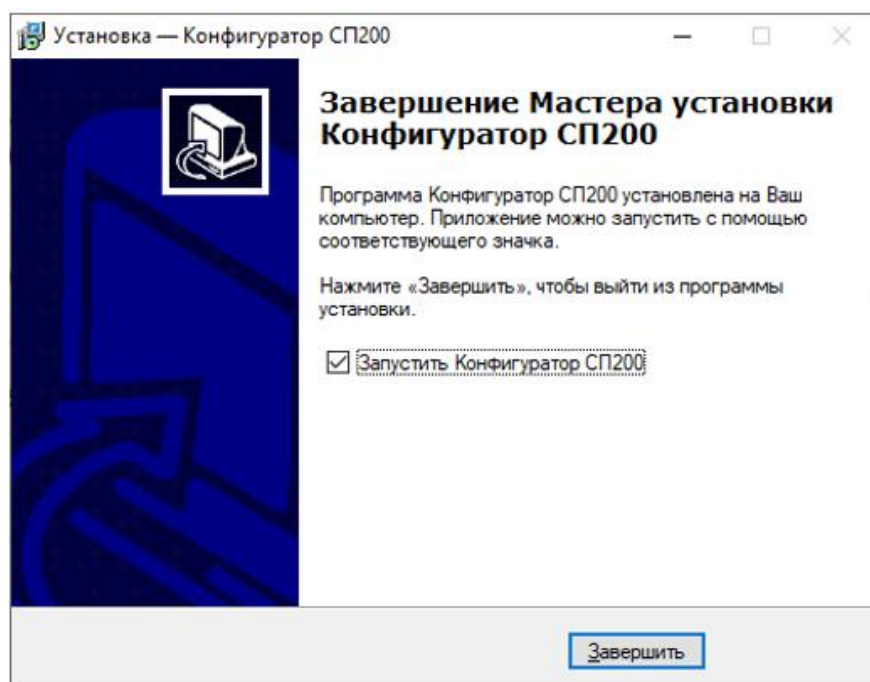


Рисунок 29 – Окно завершения установки

На рисунке 30 новый созданный проект в конфигураторе.

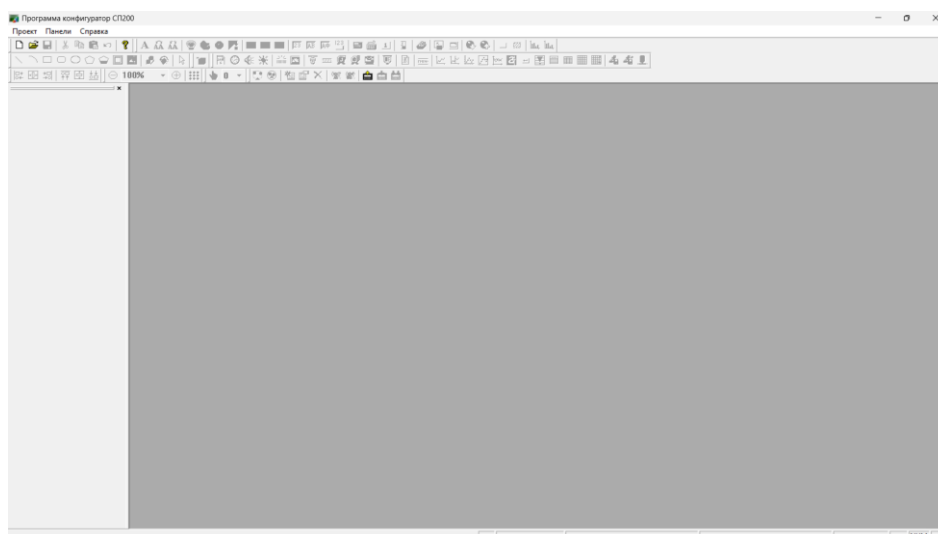


Рисунок 30 – Окно программы конфигуратор СП200

После создания проекта пользователь может ознакомиться с инструментами и при желании создать тестовый проект.



Рисунок 31 – Панель инструментов конфигуратора СП200

Далее на рисунке 32 можно увидеть дерево проекта в котором видно созданный проект.

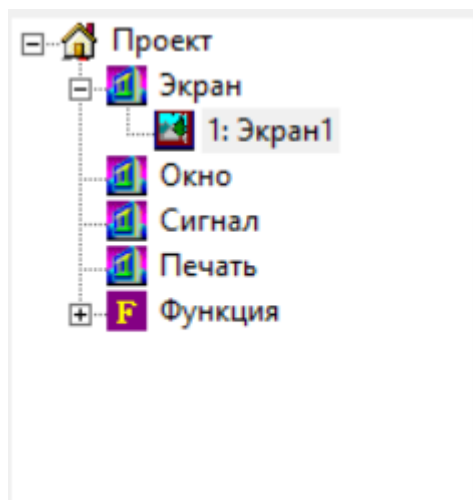


Рисунок 32 – Окно элементов проекта

1.2 Проектирование и разработка

Итогом проектирования будет являться основная программа в которой будет вся логика системы и панель оператора с созданной для неё интерфейсом.

Пользователь нажимает кнопку пуск, после вся поточная линия начинает своё движение

1.2.1 Конфигурирование ПЛК ОВЕН ПР200

Основным элементом проекта является контроллер ПР200 оснащенный дисплеем, то как выглядит прибор можно ознакомиться на рисунке 33. Данный контроллер помещен в корпус 7–din. Может включать в себя до 24 входов и выходов. Если стандартных входов и выходов не достаточно то пользователь может увеличить кол–во входов и выходов используя шину подключения. Также предусмотрена установка до двух интерфейсов

Разработка алгоритма системы была выполнена с помощью функциональных блоков OwenLogic. Для того что бы загрузить алгоритм

пользователю понадобится кабель для подключения.



Рисунок 33 – Прибор ОВЕН ПР200

Был создан новый проект в котором пользователь может выбрать нужную модель контроллера для разработки, в нашем случае ПР200–24.2(4).Х, со этим можно ознакомиться на рисунке 34.

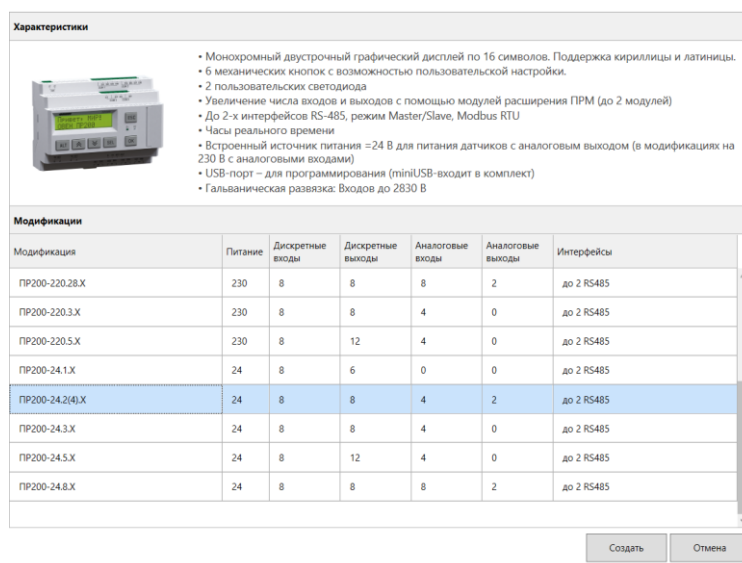


Рисунок 34 – Окно выбора модели контроллера при создании нового проекта

Для того что бы проект работал корректно пользователю потребуется указать адрес, значение которого будет составлять 1, также для дальнейшей

работы потребуется создать сетевые переменные, все это показано на рисунках 35 и 36.

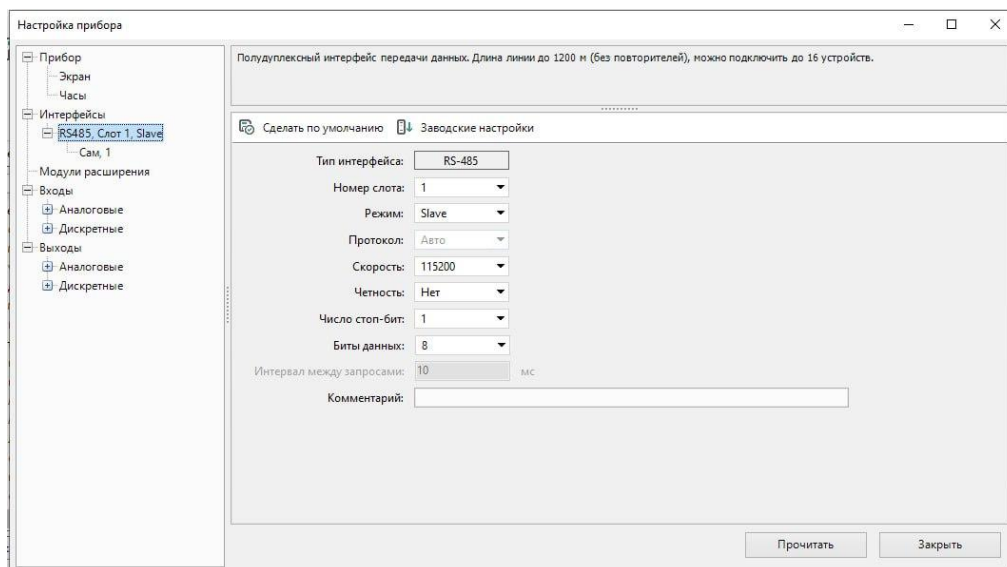


Рисунок 35 – Сетевые настройки прибора

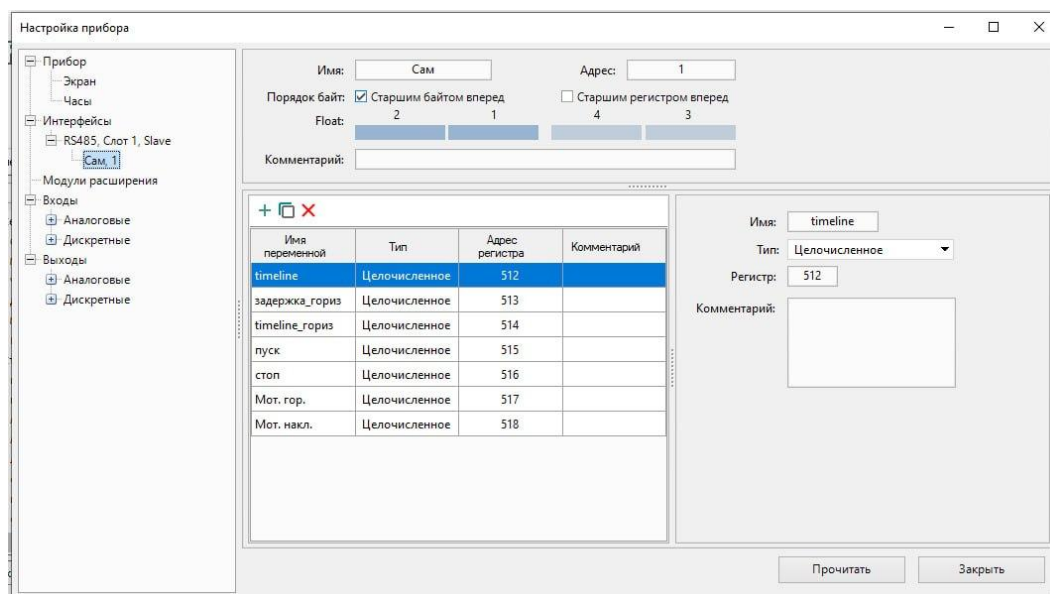


Рисунок 36 – Таблица сетевых переменных прибора

В таблице 2 показаны сетевые переменные с указанием адресов регистров.

Таблица 2 – Таблица сетевых переменных проекта

Имя переменной	Тип переменной	Адрес регистра
----------------	----------------	----------------

timeline	Целочисленное	512
задержка_гориз	Целочисленное	513
timeline_гориз	Целочисленное	514
пуск	Целочисленное	515
стоп	Целочисленное	516
Мот. гор.	Целочисленное	517
Мот. накл	Целочисленное	518

Для того что бы панель оператора начала работать необходимо добавить сетевые переменные, взаимодействие будет происходить по протоколу Modbus RTU и интерфейсу RS485, со всем этим можно ознакомиться на рисунке 37

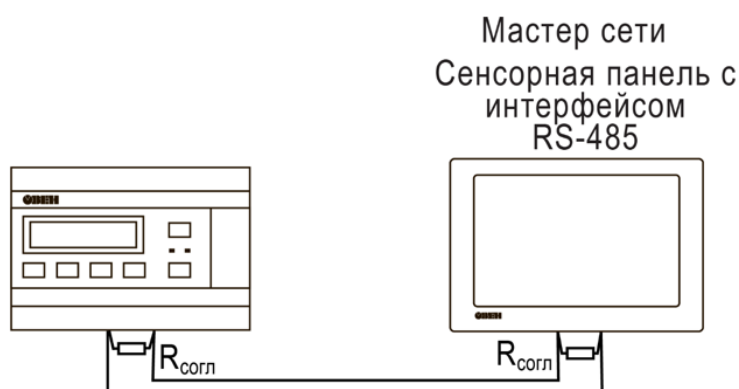


Рисунок 37 – Схема сетевого взаимодействия мастер–панели с ПЛК

Для того что бы реализовать функциональные требования пользователя ему также потребуется создать локальные переменные что бы контроллер работал корректно, это показано на рисунке 38 и таблице 3.

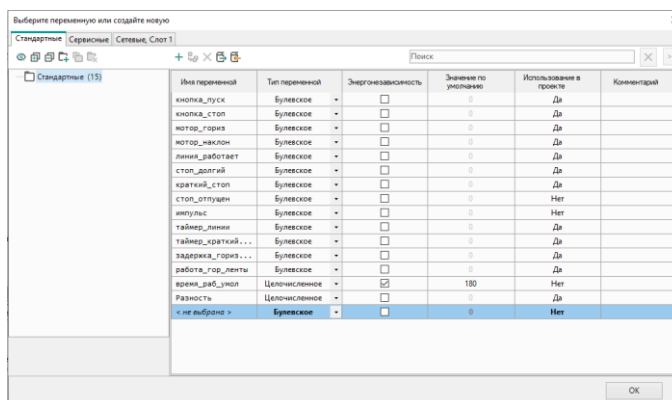


Рисунок 38 – Таблица локальных переменных проекта

Таблица 3 – локальные переменные проекта

Имя переменной	Тип переменной
кнопка_пуск	булевское
кнопка_стоп	булевское
мотор_гориз	булевское
мотор_наклон	булевское
линия_работает	булевское
стоп_долгий	булевское
краткий_стоп	булевское
стоп_отпущен	булевское
Имя переменной	Тип переменной
импульс	булевское
Таймер_линии	булевское
Таймер_краткий_стоп	булевское
Задержка_горизонтальной	булевское
Работа_гор_ленты	булевское
Время_раб_умол	целочисленное
разность	целочисленное

1.2.2 Разработка управляющей программы

Для работы с конвейерной линией необходимо реализовать время работы всей линии, по умолчанию время работы 3 минуты, то есть 180 секунд, через это время конвейер должен остановиться сам.

Время работы всей линии реализовано через TON. Таймер с задержкой включения решает простую задачу, он откладывает прохождение сигнала.

На рисунке 40 представлен готовый фрагмент кода с настроенным таймером работы всей линии.

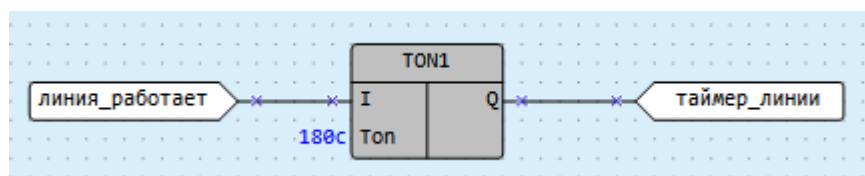


Рисунок 40 – Таймер работы всей линии.

Цикл линии выглядит следующим образом на рисунке 41 и реализован с помощью RS-триггера.

RS-триггер с приоритетом выключения будет использоваться для контроля состояния при поступлении импульсов на определенный вход.

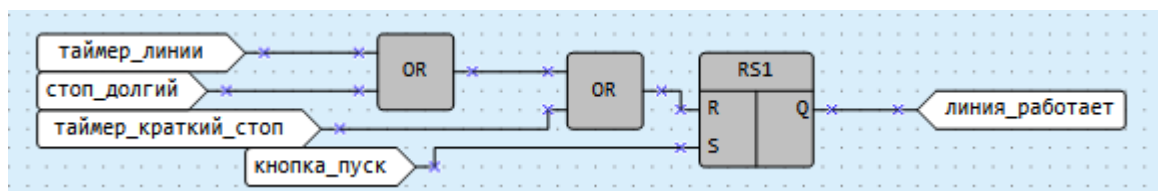


Рисунок 41 – Реализация цикла линии конвейера

Далее нам требуется реализовать работу моторов двух конвейерных лент, горизонтальной и наклонной. Мотор наклонной ленты всегда должен останавливаться раньше горизонтальной, чтобы избежать поломок.

Так как горизонтальный мотор в стандартном режиме всегда останавливается раньше, он реализован простым кодом и представлен на рисунке 42.

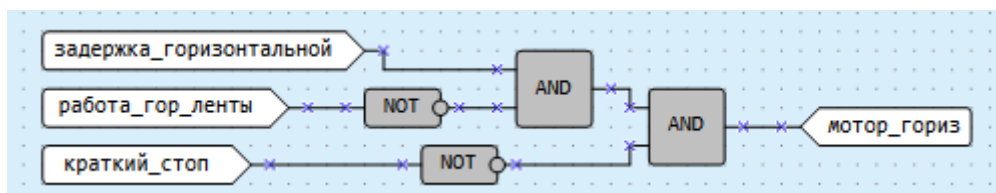


Рисунок 42 – Реализация мотора горизонтальной ленты

Для создания потребовалось использовать AND и NOT и переменные «задержка_горизонтальной» – они позволяют реализовать запуск горизонтальной ленты с задержкой 10 секунд по умолчанию.

«работа_гор_ленты» – время работы горизонтальной ленты.

«краткий_стоп» – переменная, с помощью которой реализуется остановка ленты.

Для реализации наклонного мотора уже были использованы такие логические блоки как OR и AND, а также RS-триггер с приоритетом выключения. На рисунке 43 представлен готовый алгоритм работы мотора наклонной ленты.

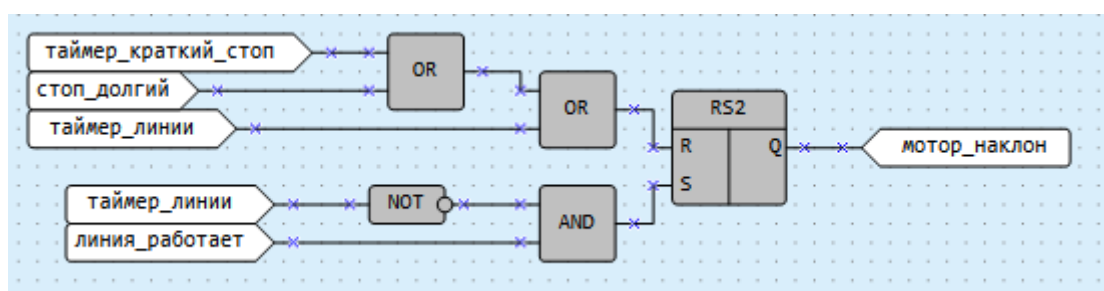


Рисунок 43 – Алгоритм работы мотора наклонной ленты

Были также использованы переменные:

- «таймер_краткий_стоп»;
- «стоп_долгий»;
- «таймер_линии»;
- «работа_линии».

Более подробно про эти переменные и их алгоритмы мы поговорим позже.

Для упрощения схемы, был создан макрос, чтобы структурировать программный код, был создан макрос «конвейер» на рисунке 44.

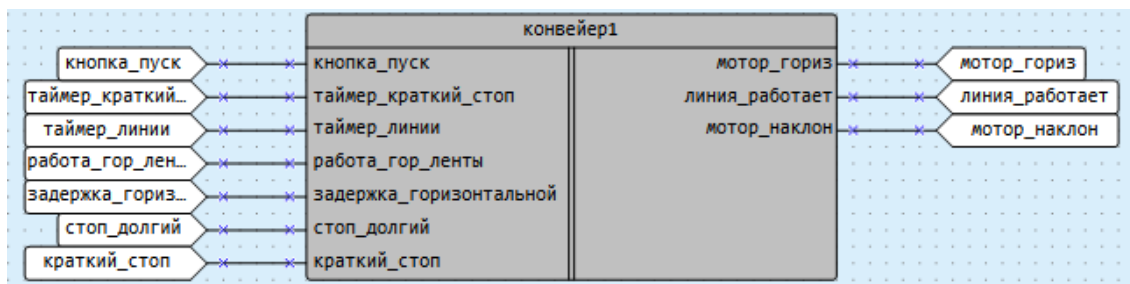


Рисунок 44 – Готовый макрос конвейер

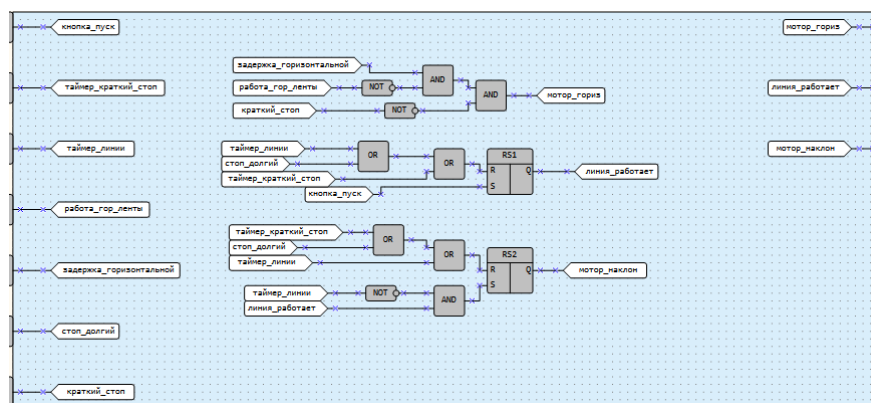


Рисунок 45 – Содержимое макроса «конвейер»

Далее для реализации автоматизированного выключения и задержек, благодаря которым будет реализована поочередность включения и настройка всех таймеров через панель оператора были реализованы следующие методы решения.

На рисунке 46 представлен таймер работы горизонтальной ленты, который по умолчанию должен отключаться за 20 секунд, до остановки наклонного.

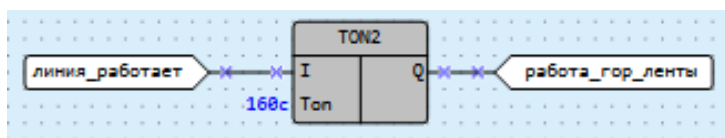


Рисунок 46 – Таймер горизонтальной ленты

Также для настройки этого таймера через панель оператора был сделан

код, продемонстрированный на рисунке 47.

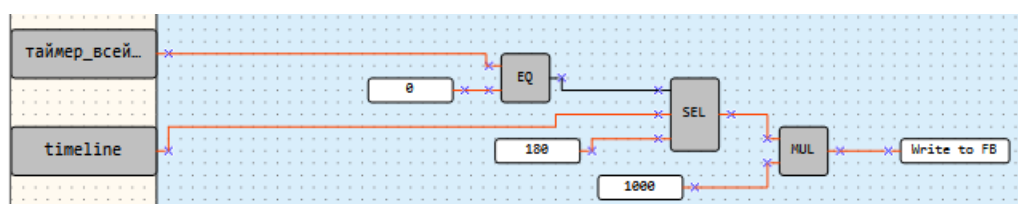


Рисунок 47 – Настройка таймера

Алгоритм был реализован с помощью блоков EQ – операция сравнения, SEL – тернарная условная операция сравнения и MUL – арифметическая операция умножения, а также на выходе блок записи в функциональный блок, а именно задание времени для таймера.

На блок EQ поступает время, заданное оператором, которое сравнивается с 0, если пользователь ничего не ввел, время остается по умолчанию 180 секунд, для того, чтобы получить 180 секунд, используется арифметическая операция умножения, которая позволяет настраивать время в секундах, так как в среде программирования используются миллисекунды.

1 секунда = 1000 миллисекунд, поэтому необходимо умножать на 1000.

Если V1 равен логическому «0», то результатом выполнения операции блока на выходе является входной сигнал V2. Если V1 равен логической «1», то результатом выполнения операции блока на выходе является входной сигнал V3.

- $V1 = 0 \rightarrow Q = V2;$
- $V1 = 1 \rightarrow Q = V3.$

Для выбора целочисленных значений используется блок SEL, для значений с плавающей запятой — блок fSEL.

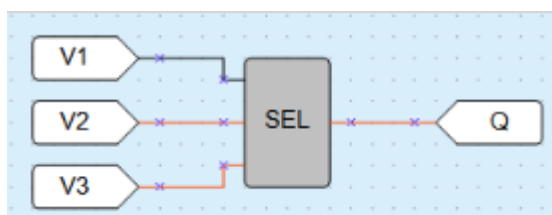


Рисунок 48 – Пример кода с использованием SEL

Результатом операции функции на выходе является произведение входных значений.

Если во время выполнения операции функции значение числа получается больше 4294967295 (32 бита), то биты, выходящие за разрядность 32 бита, отсекаются.

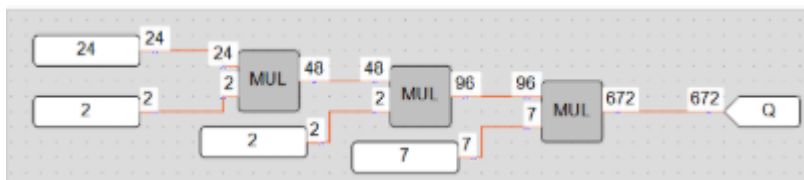


Рисунок 49 – Пример кода с использованием MUL

Результаты выполнения операции блока на выходе сравнения входных значений:

- $V1 = V2 \rightarrow Q = 1;$
- $V1 > V2 \rightarrow Q = 0;$
- $V1 < V2 \rightarrow Q = 0.$

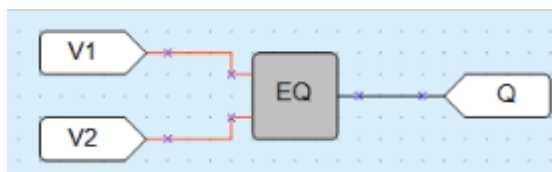


Рисунок 50 – Пример кода с использованием EQ

Аналогичным образом мы создаем настройку для остальных таймеров, которые используются в проекте. Рисунок 51–52

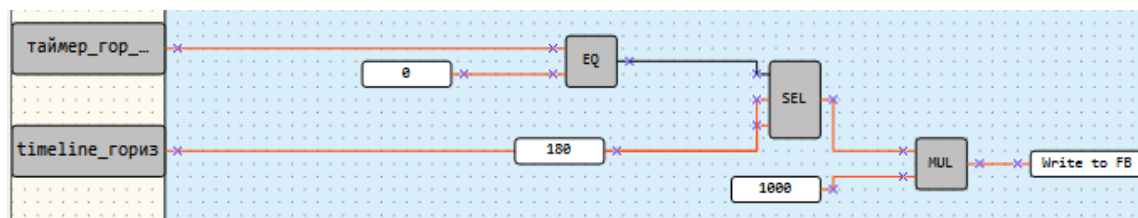


Рисунок 51 – Алгоритм настройки задержки для горизонтальной ленты

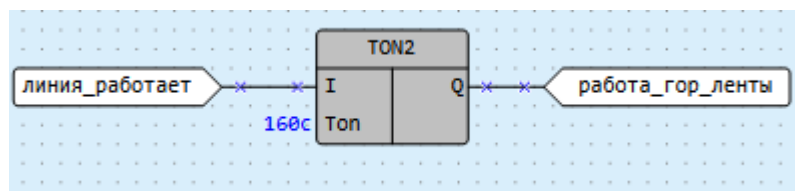


Рисунок 52 – Таймер задержки горизонтальной ленты

Для работы с таймерами был создан отдельный макрос «таймеры», в котором реализованы все вышеперечисленные алгоритмы, для ручной настройки времени работы.

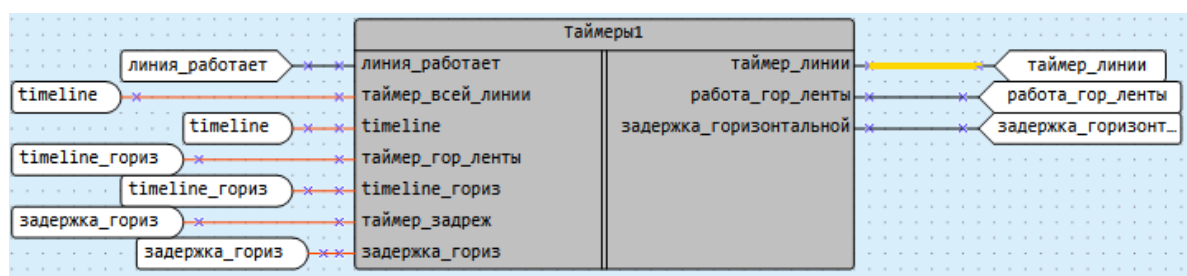


Рисунок 53 – Макрос таймеров

Код внутри макроса продемонстрирован на рисунке 54

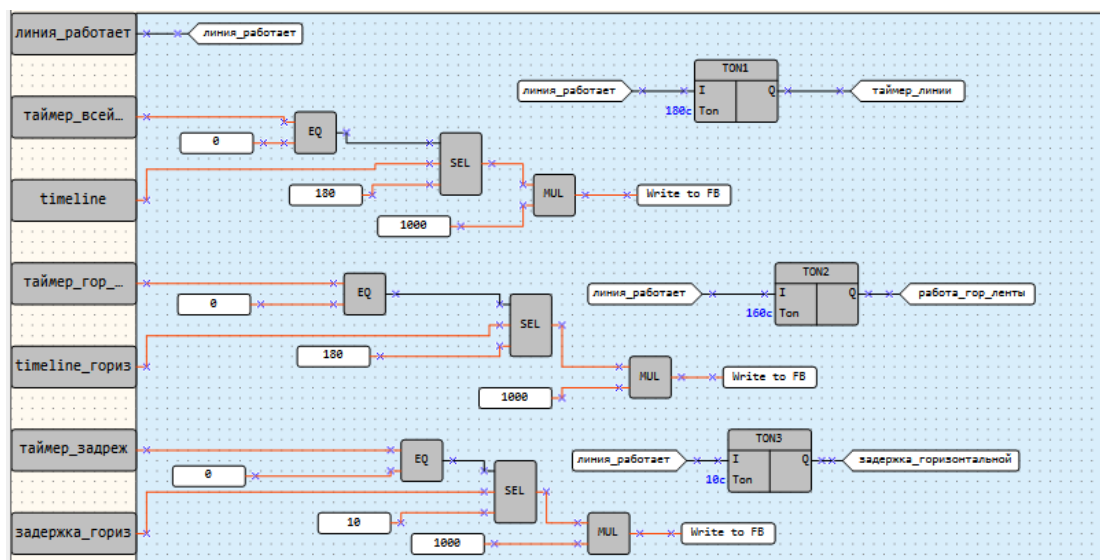


Рисунок 54 – Код макроса «таймеры»

У кнопки стоп есть несколько режимов работы, краткосрочное нажатие и удержание кнопки 2 секунды. При краткосрочном нажатии стол останавливается спустя 20 секунд после нажатия.

Для реализации необходимо сделать регистрацию краткосрочного нажатия и сохранение в переменную.

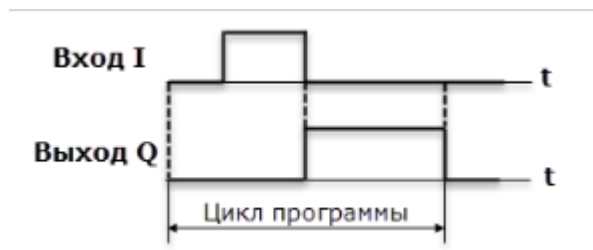


Рисунок 55 – Принцип работы детектора заднего фронта

Реализация логики для регистрации краткосрочного нажатия показана на рисунке таком-то.

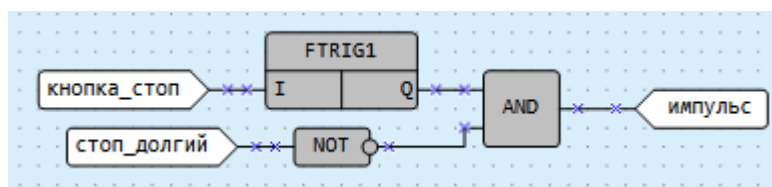


Рисунок 56 – Сохранение импульса краткосрочного нажатия

Для того, чтобы распознать краткосрочное нажатие или долгосрочное, кнопка должна быть нажата не более 2,1 секунд

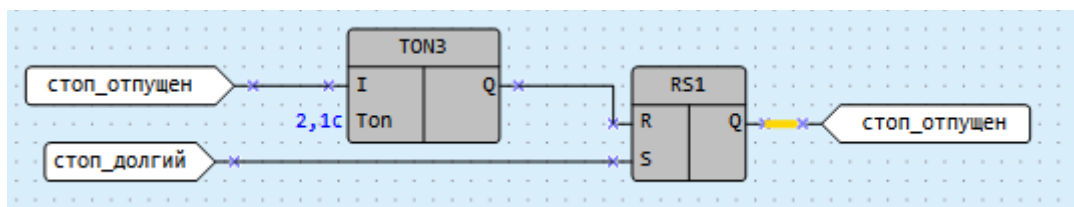


Рисунок 57 – Распознавание краткосрочного нажатия

После того как кнопка отпущена и импульс получен и распознан краткосрочное нажатие кнопки, нужна новая переменная «краткий_стоп», которая запустит процесс остановки моторов линии в нужной нам последовательности. Алгоритм представлен на рисунке 58.

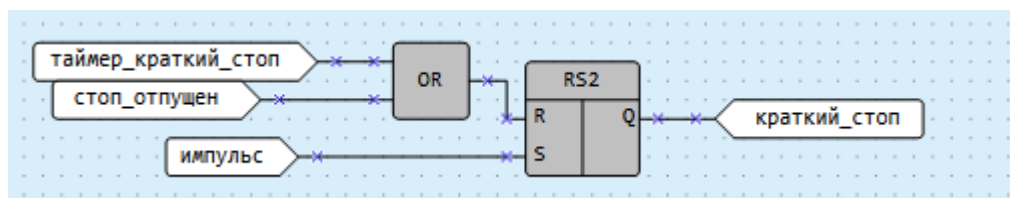


Рисунок 58 – Алгоритм работы краткого нажатия кнопки

После того, как программа распознала краткосрочное нажатие кнопки, запускается таймер, по истечению которого начнут останавливаться моторы в необходимой нам последовательности и сбрасываться таймер работы всей линии. Таймер представлен на рисунке 59.

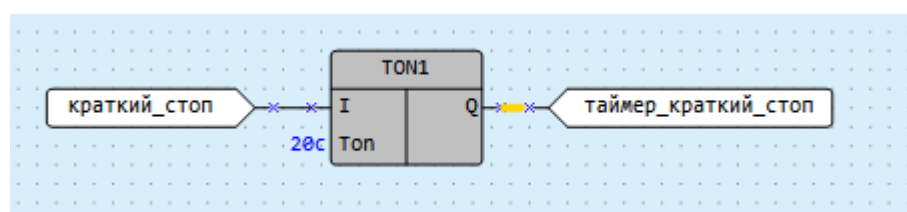


Рисунок 59 – Таймер работы кратковременного нажатия кнопки стоп

Таймер автоматически настраивается при помощи разности таймеров всей линии и таймера горизонтальной ленты, для того чтобы горизонтальная линия останавливалась раньше, реализовано это через блок SUB, пример представлен на рисунке 60.

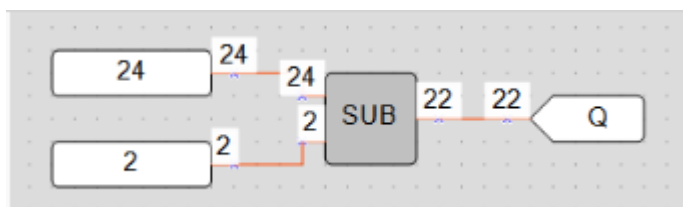


Рисунок 60 – Пример работы разности

На рисунках 61–62 представлена разность между таймером конвейерной линии и горизонтальной ленты.

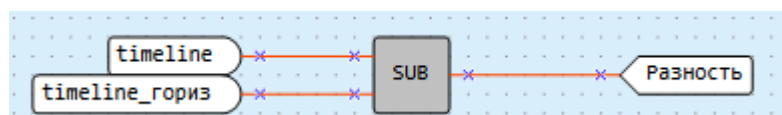


Рисунок 61 – Автоматическая настройка таймера «краткий_стоп»

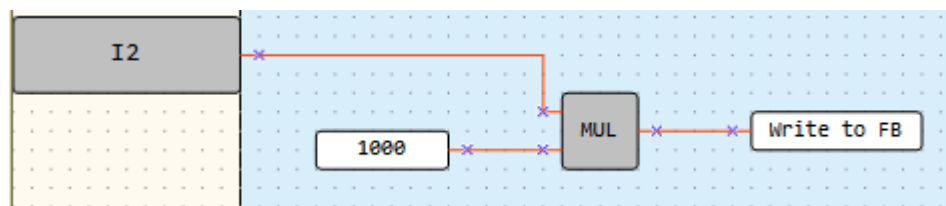


Рисунок 62 – Автоматическая настройка таймера «краткий_стоп»

Регистрация долговременного нажатия реализована проще всего, через таймер, который по умолчанию равен 2 секундам и представлен на рисунке 63. После долгого нажатия, все моторы одновременно останавливаются.

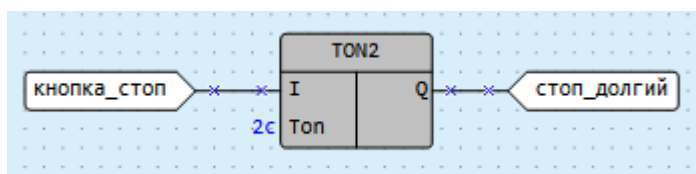


Рисунок 63 – Реализация долго нажатия кнопки «стоп»

Для визуальной составляющей также был создан отдельный макрос, в котором находится весь алгоритм с режимами работы кнопки стоп. На рисунке 64 представлен весь макрос.

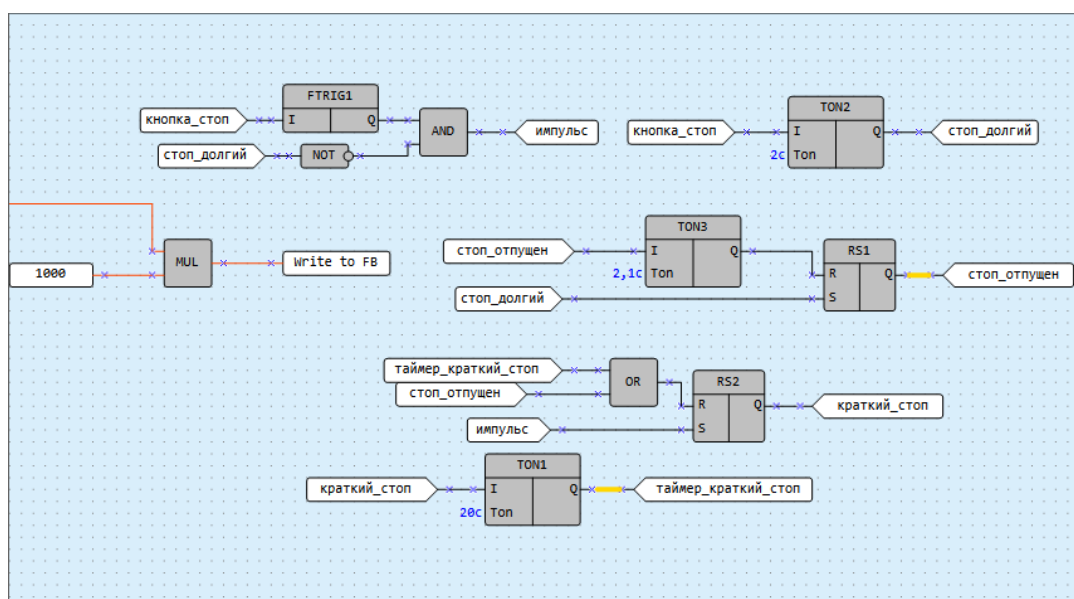


Рисунок 64 – Макрос с логикой кнопки стоп

На основной схеме макрос выглядит следующим образом

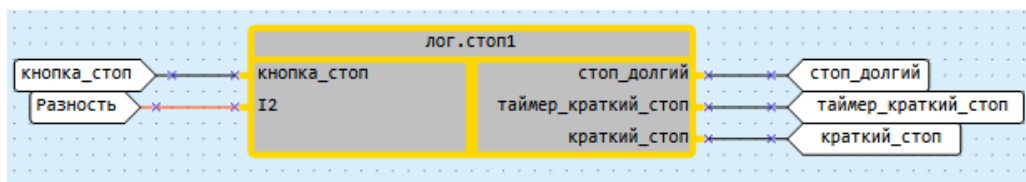


Рисунок 65 – Макрос с логикой кнопки стоп

Также для работы с панелью оператора, необходимо добавить сетевые переменные, на рисунке таком–то представлены сетевые переменные, а также в таблице 66.

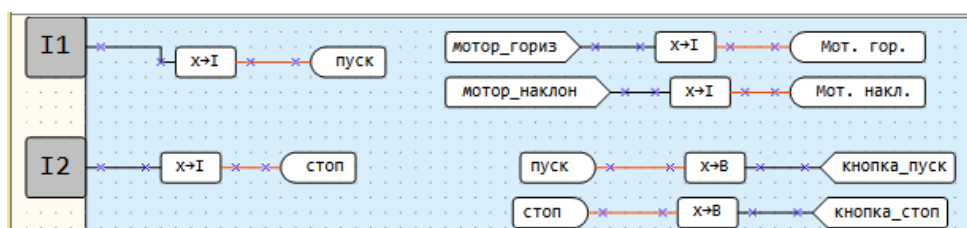


Рисунок 66 – Добавление сетевых переменных

После успешного создания основной программы нужно загрузить её на контроллер. Для того что бы загрузить систему потребуется наличие провода указанного в документации, схема подключения показана на рисунке 67.

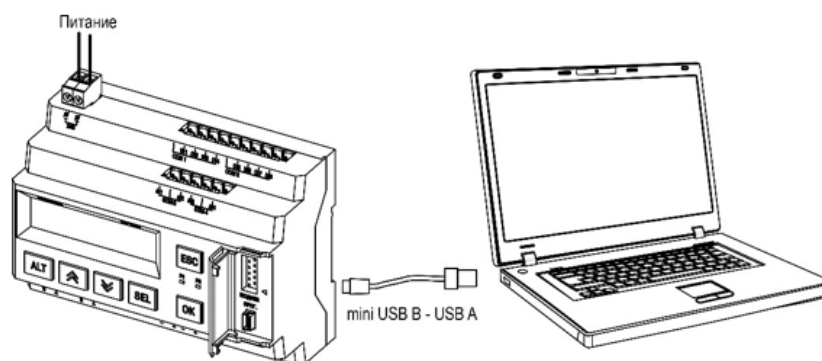


Рисунок 67 – Схема подключения ПЛК к ПК

1.2.3 Конфигурирование панели оператора ОВЕН СП270

Учитывая функциональные требования к управляющей программе, были построены UML–диаграммы двух видов: вариантов использования и диаграмма деятельности.

При проектировании предполагалось что будут использоваться отдельные панели оператора для администратора и пользователя, но для того что бы показать функционал в тестовом стенде будет использоваться одна панель оператора, её будет достаточно для демонстрации функционала. В следствие этого функционал администратора потребовалось перенести на отдельный экран.

На рисунке 68 представлена UML–диаграмма деятельности, на которой показано как пользователь будет взаимодействовать с панелью оператора и контроллером.

Похожая UML–диаграмма с реализацией функциональных требований для администратора представлена на рисунке 69.

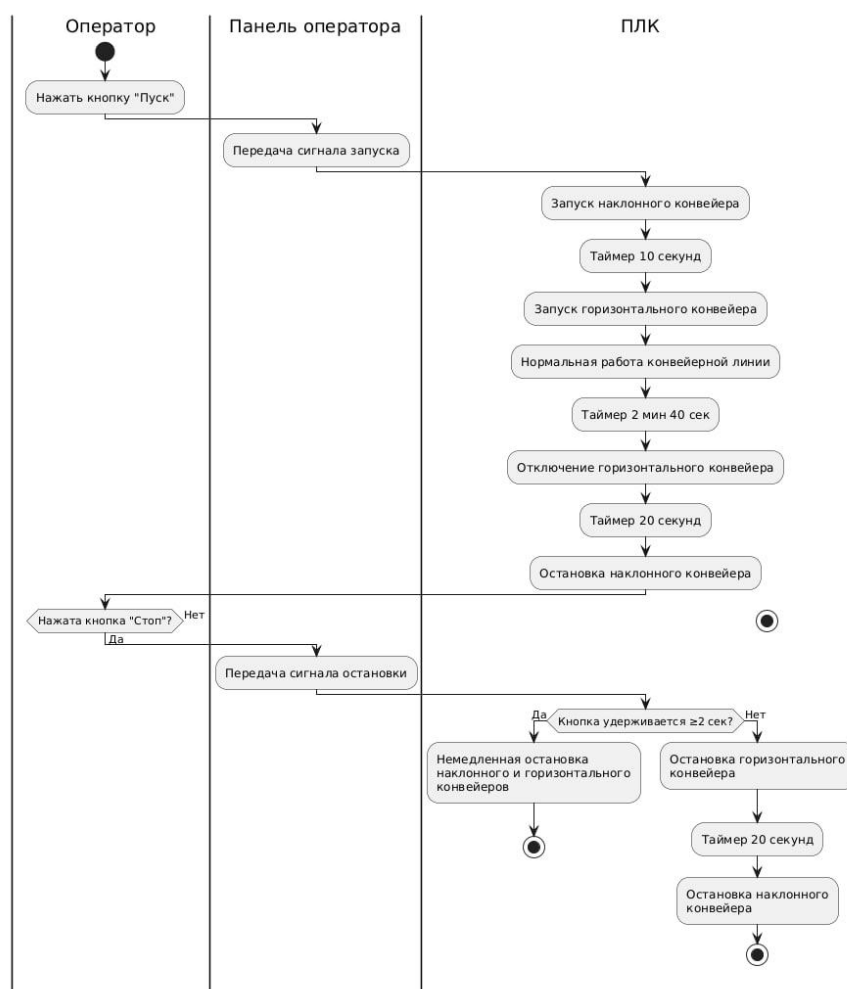


Рисунок 68 – UML диаграмма для пользователя

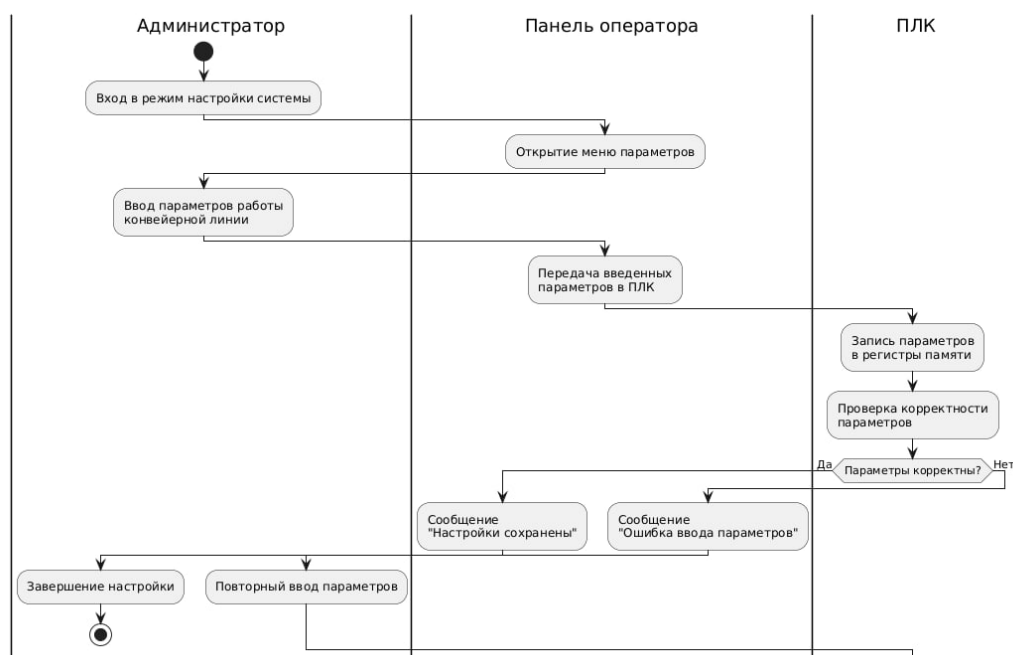


Рисунок 69 – UML диаграмма для администратора

На рисунках 70–71 показаны UML–диаграммы вариантов использования.



Рисунок 70 – UML–диаграмма сценариев использования для пользователя



Рисунок 71 – UML–диаграмма сценариев использования для администратора

После анализа всей информации был начат этап создания конфигурации панели оператора.

Началом было создание проекта в конфигураторе, как показано на рисунке 72.

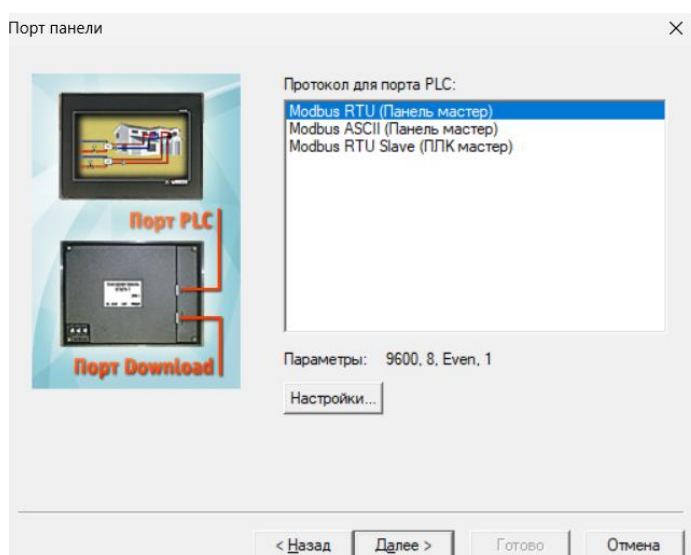


Рисунок 72 – Выбор протокола связи с ПЛК

Для того что бы можно было подключить панель оператора к

контроллеру потребовалось выбрать определенные параметры, параметры показаны на рисунке 73.

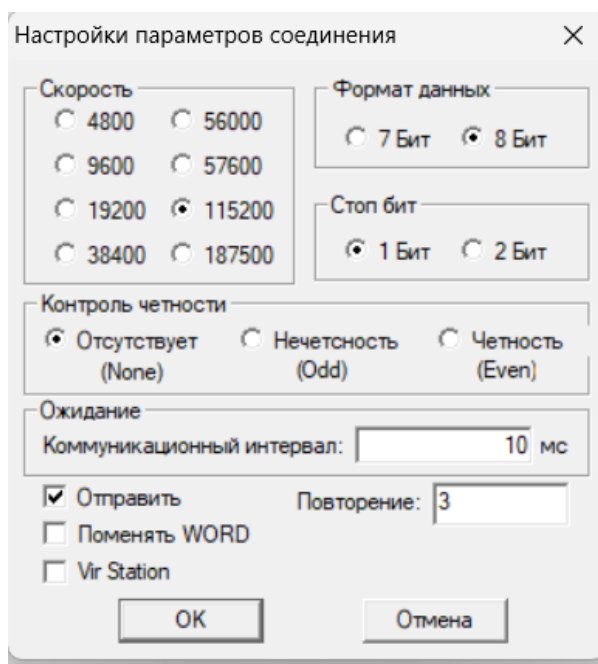


Рисунок 73 – Настройки параметров соединения

В момент выбора порта на рисунке 74 был выбран тот что порт не используется.

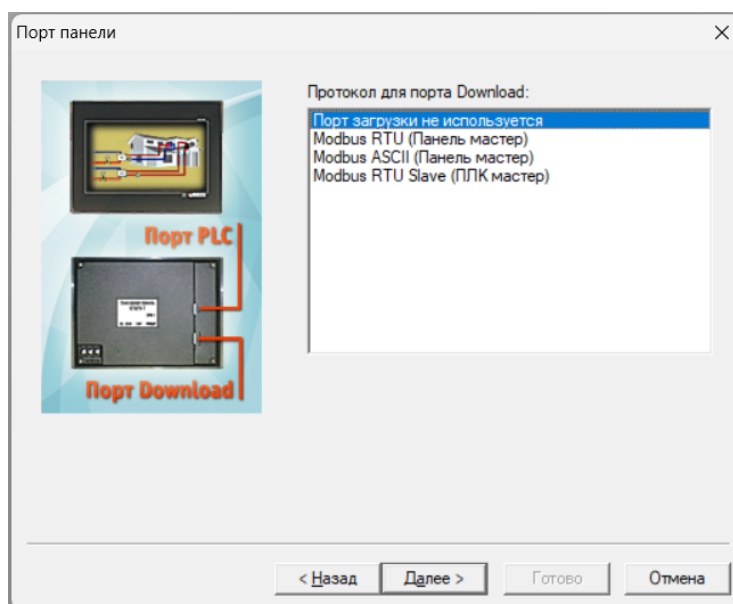


Рисунок 74 – Выбор протокола для порта загрузки

После создания проекта открывается главная рабочая область, продемонстрированная на рисунке 75.

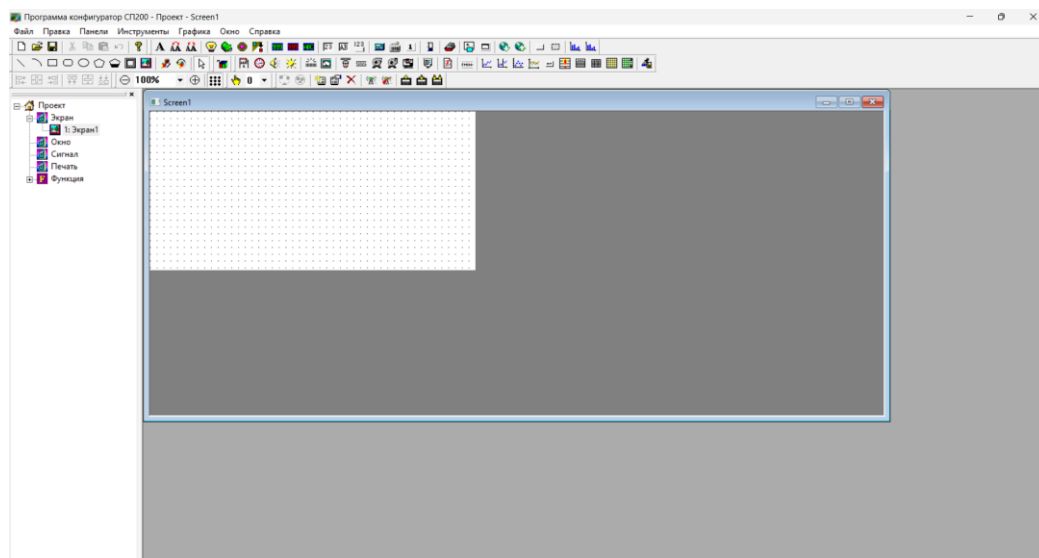


Рисунок 75 – Окно нового проекта Конфигуратора СП200

1.3 Тестирование результатов разработки

1.3.1 Тестирование и отладка программы в редакторе OWEN Logic

С целью проверки корректности программы требуется организовать тестирование программы с целью проверки на соответствие поставленной задаче. Его результаты представлены на рисунке 76

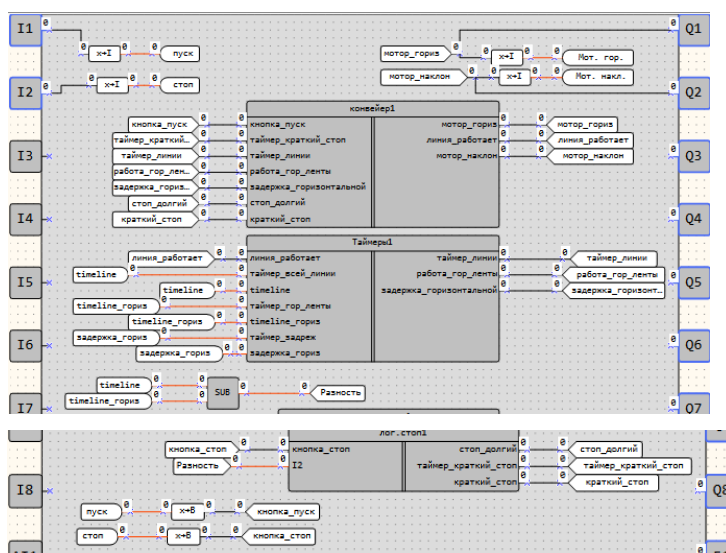


Рисунок 76 – Проект в режиме симуляции в Owen Logic

На рисунке 77 представлена работа кнопки пуск, на вход будет подаваться сигнал после чего будет включен мотор наклонной ленты.

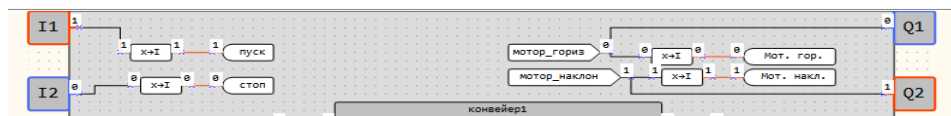


Рисунок 77 – Сигнал на I1

Спустя 10 секунд должен активироваться мотор горизонтальной линии на выходе Q1. Что показано на рисунке 78.



Рисунок 78 – Запуск горизонтальной ленты спустя 10 секунд

Вся линия должна быть остановлена спустя 3 минуты работы, на рисунках 79–80 показана настройка таймера для всей линии.

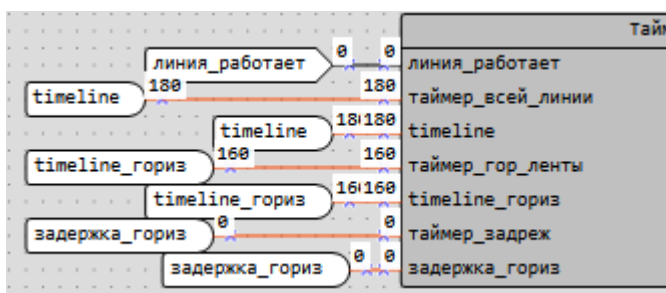


Рисунок 79 – Настройка таймера линии

Рисунок 80 – Ввод времени

На 20 секунд раньше должен останавливаться мотор горизонтальной ленты, а наклонный останавливается спустя 3 минуты. На рисунках 81–82 продемонстрирована вся последовательность выключений моторов.

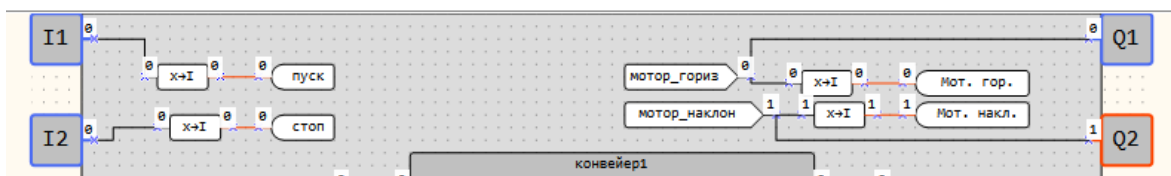


Рисунок 81 – Остановка горизонтальной ленты

Наклонный таймер останавливается по истечении 3 минутного таймера или по времени, которое введет пользователь.

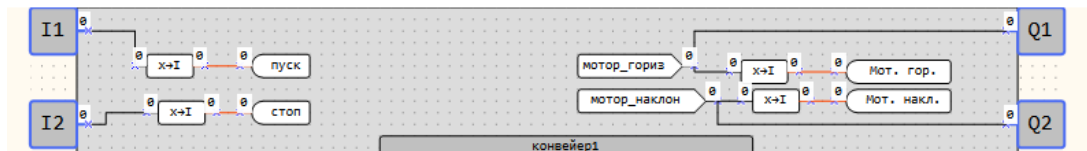


Рисунок 82 – Остановка наклонной ленты

Остановка по таймеру

При срабатывании входа I2 и удержании его больше 2 секунд конвейер должен одновременно остановить свои линии на выходах Q1 и Q2. Результат тестирования предоставлен на рисунке 83.

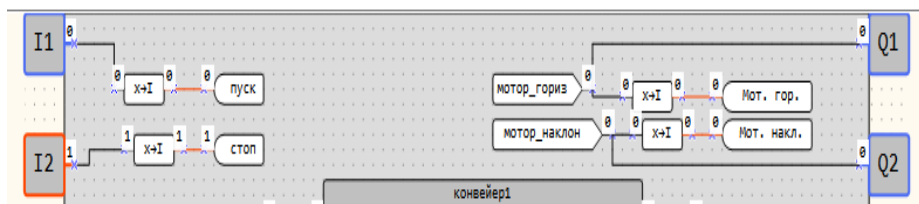


Рисунок 83 – Остановка сразу двух конвейеров

Для того что бы правильно остановить один из конвейеров требуется нажать кнопку стоп на входе I2. На рисунках 84–85 показано, что при нажатии кнопки стоп на входе I2, сразу останавливается горизонтальная конвейерная лента, чтобы в последующем она не нагружала наклонную.

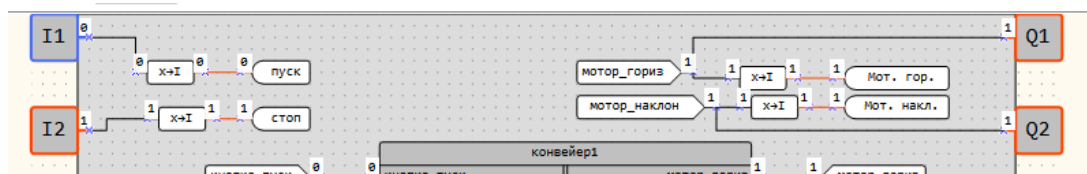


Рисунок 84 – Кратковременное нажатие кнопки стоп

Спустя 20 секунд останавливается наклонная конвейерная лента на

выходе Q2. На рисунке 85 показан результат после ожидания 20 секунд



Рисунок 85 – Остановка конвейерной линии из двух лент

Для настроек таймера можно задать сетевым переменным необходимое время в секундах. На рисунке 86 показаны сетевые переменные, которые можно изменить через панель оператора.

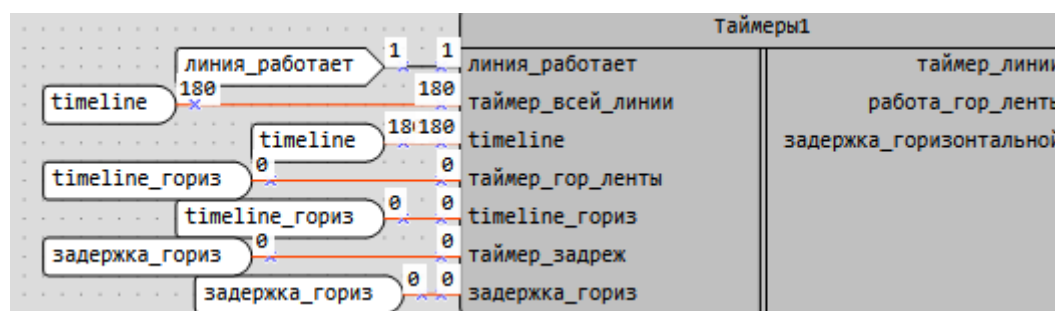


Рисунок 86 – Настройка таймеров

Результаты тестирования программы в среде программирования OwenLogic, в режиме симуляции, показали корректную работу управляющей программы для ПЛК ПР200.

1.3.2 Тестирование и отладка интерфейса панели оператора ОВЕН СП270

После окончания разработки была начата проверка панели оператора, проверка была произведена в режиме эмуляции offline, что показано на рисунках 87,88.

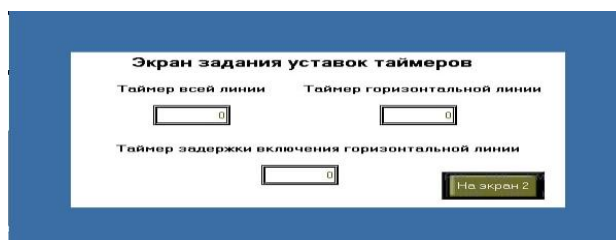


Рисунок 87 – Экран 1 в режиме симуляции Offline

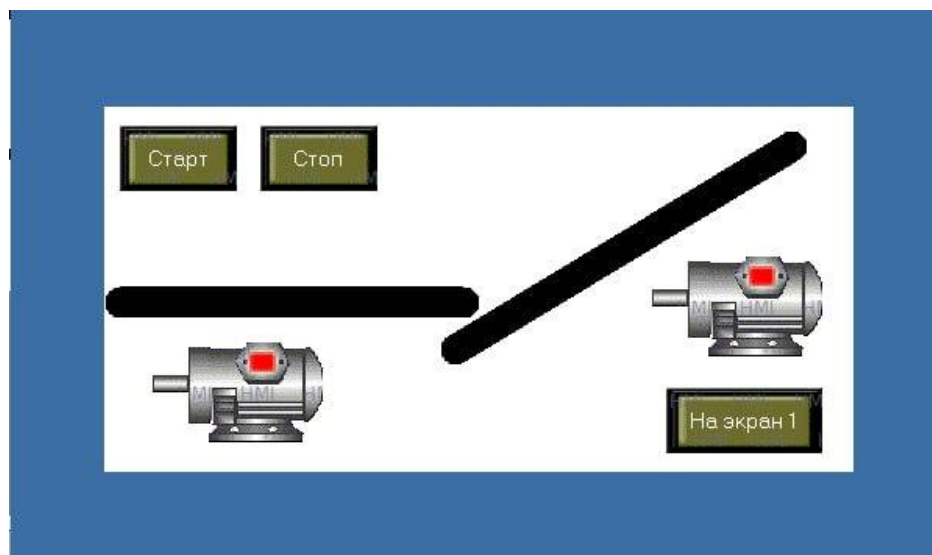


Рисунок 88 – Экран 2 в режиме симуляции Offline

1.3.3 Тестирование проекта на стенде целевой платформы

После начала этапа отладки была произведена проверка на тестовом стенде. На рисунке 89 показаны контроллер с загруженными программами, панель оператора и интерфейс панели оператора.



Рисунок 89 – Лабораторный стенд целевой платформы

Перед пользователем на экране 1 показаны поля для выбора уставок таймеров для работы, это видно на рисунке 90.

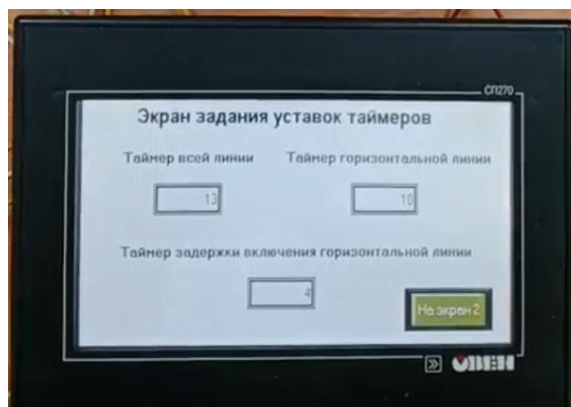


Рисунок 90 – Экран с указанием уставок таймеров

После нажатия поля таймер всей линии перед пользователем будет выдано окно с возможностью указанию нужного числа для задания времени всей линии, что показано на рисунке 91 и 92.

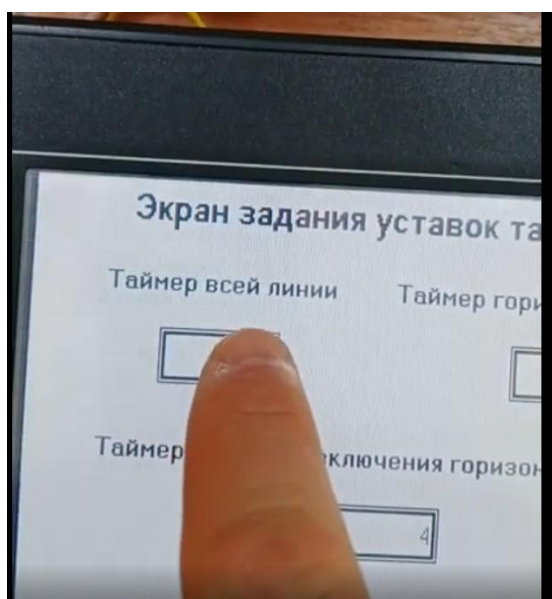


Рисунок 91 – Выбор поля таймер всей линии



Рисунок 92 – Меню с указанием числа для таймера всей линии

Следующим этапом пользователь должен указать таймер горизонтальной линии, перед пользователем будет показан интерфейс в котором он может указать значение для поля, все это показано на рисунках 93 и 94.

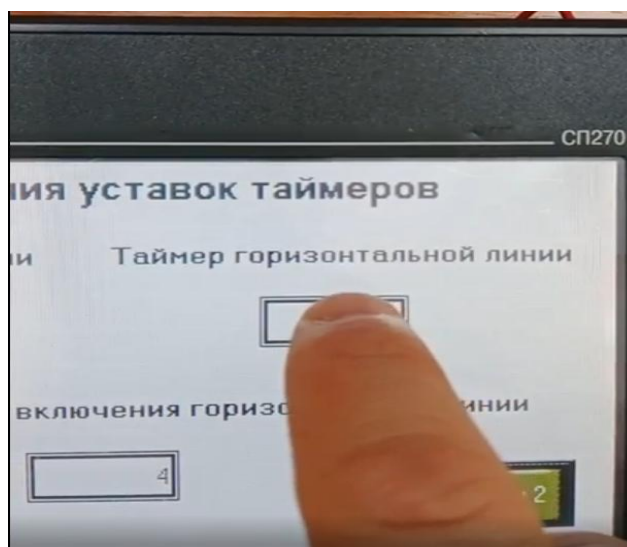


Рисунок 93 – Выбор поля таймер горизонтальной линии

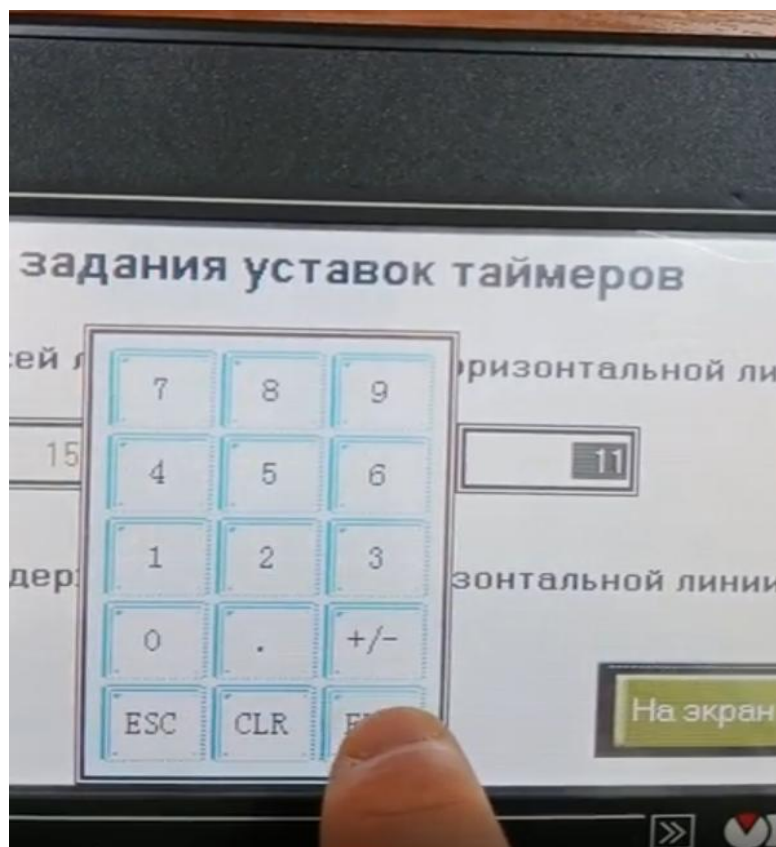


Рисунок 94 – Меню с указанием числа для таймера горизонтальной линии

Далее пользователь может задать нужное ему время задержки включения горизонтальной линии. При нажатии на данное поле пользователю так же будет выдано меню в котором он сможет указать время задержки линии, всё это показано на рисунках 95 и 96.

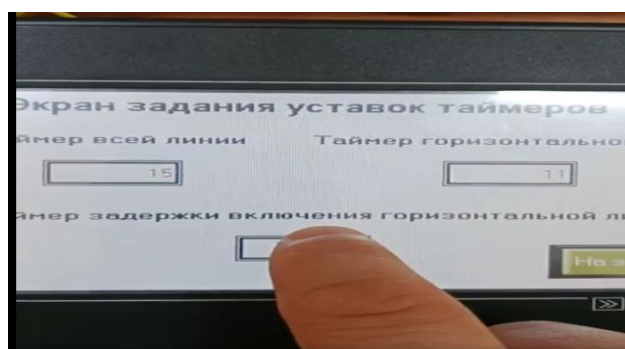


Рисунок 95 – Выбор поля таймер задержки включения горизонтальной линии

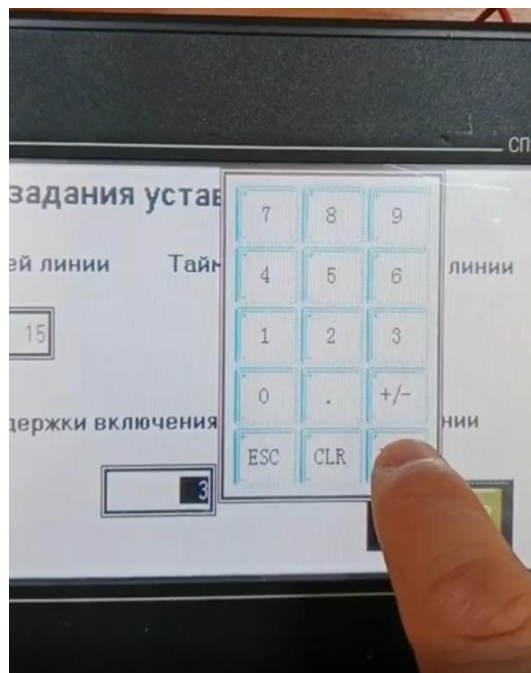


Рисунок 96 – Меню с указанием числа для таймера задержки включения горизонтальной линии

После нажатия кнопки на экран 2 пользователь будет перемещен на следующей экран, в котором будет показана работа всех двигателей, все это показано на рисунках 97 и 98.

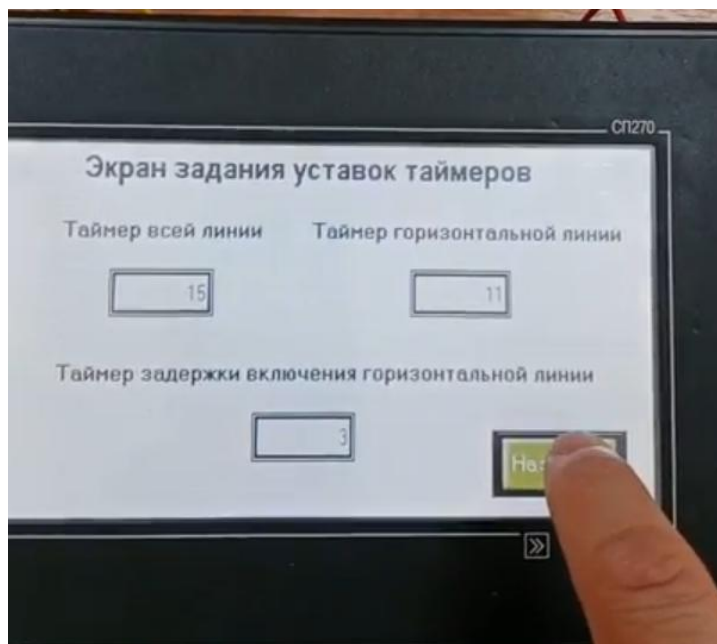


Рисунок 97 – Нажатие кнопки на экран 2

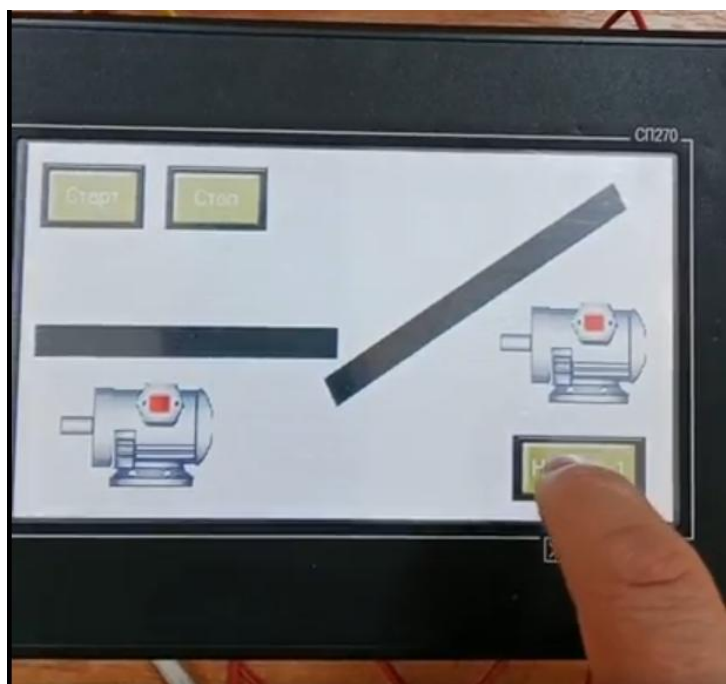


Рисунок 98 – Результат работы кнопки перехода на экран 2

Как только будет нажата кнопка старт будет включен первый двигатель, через время будет включен и второй двигатель, что и показано на рисунке 99 и 100.

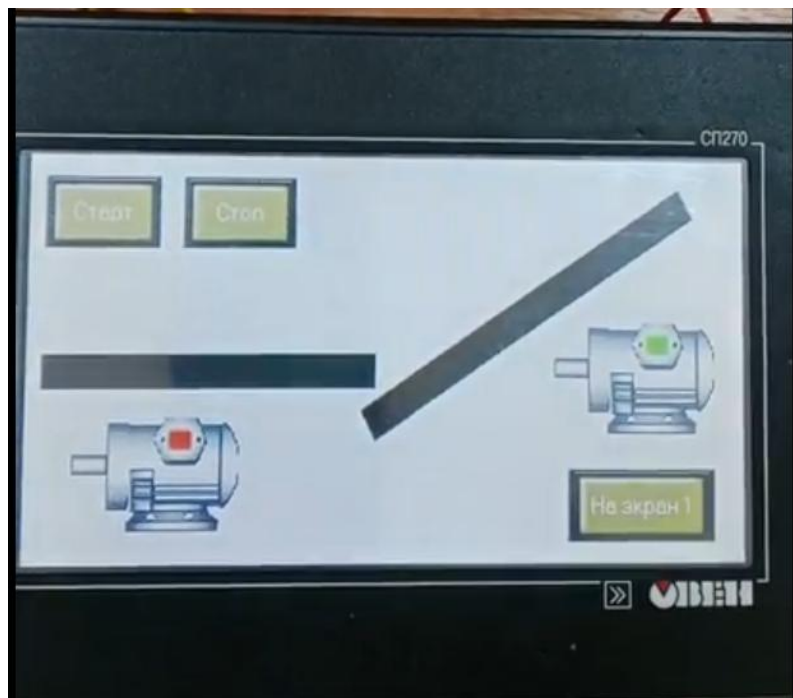


Рисунок 99 – Работа одного двигателя

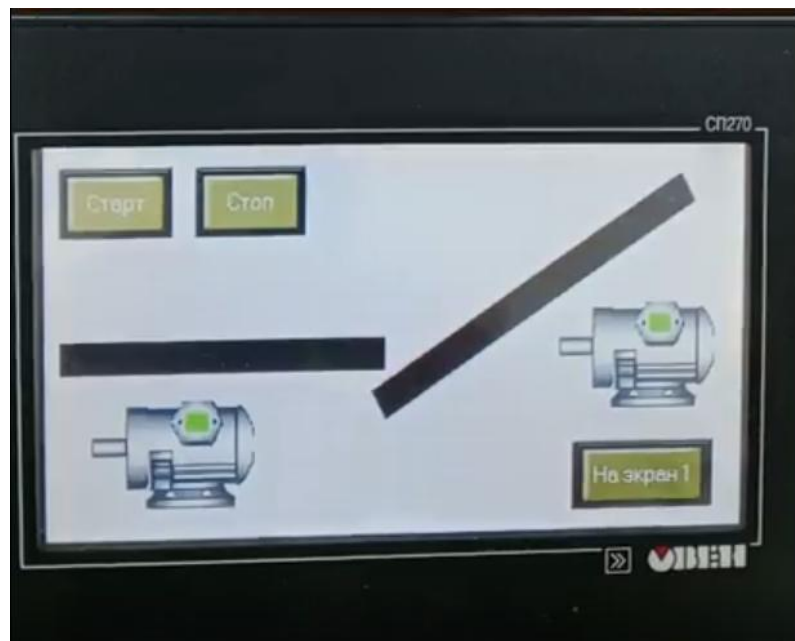


Рисунок 100 – Работа двух двигателей

Через ранее указанное время один из двигателей остановит свою работу, но второй двигатель всё еще продолжит свою работу, что и видно на рисунке 101.



Рисунок 101 – Результат прекращения работы одного из двигателей

Так же через указанное ранее время на экране 1 второй двигатель так же

прекратит свою работу, как и показано на рисунке 102.

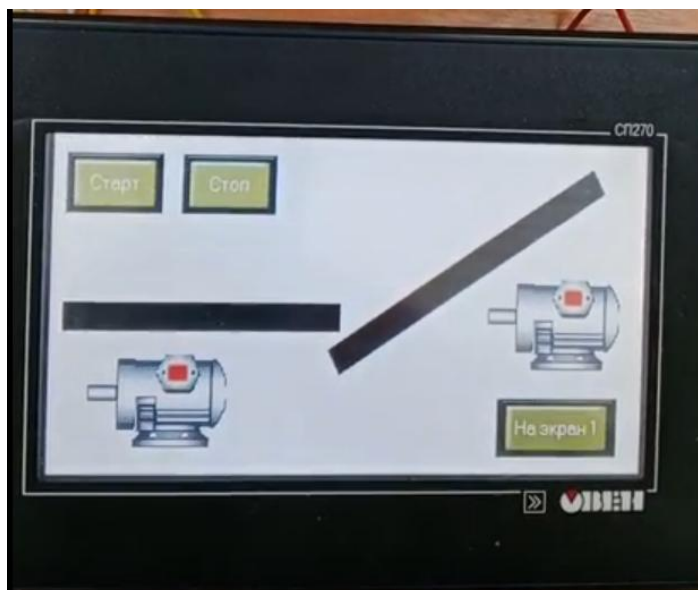


Рисунок 102 – Результат прекращения работы обоих двигателей

Далее после запуска двигателей пользователь может остановить работу двигателей, сначала будет выключен один из двигателей после чего за ним выключится и второй двигатель, все это показано на рисунках 103 и 104.

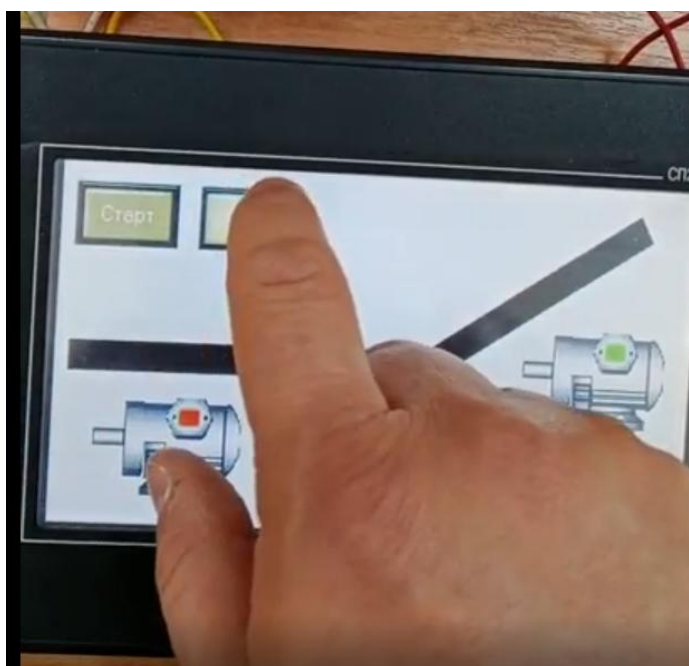


Рисунок 104 – Отключение первого двигателя после нажатия кнопки
СТОП



Рисунок 105 – Отключение второго двигателя после нажатия кнопки
СТОП

Так же после запуска двигателей пользователь может отключить сразу оба двигателя, для этого пользователю потребуется зажать кнопку стоп, после чего двигатели будут отключены, это показано на рисунке 106.



Рисунок 106 – Демонстрация работы отключения обоих двигателей

2 Охрана труда

Этот раздел посвящен вопросам безопасности при работе с компьютером в ходе создания системы, а также при установке контроллера и панелей оператора. В этом разделе будут расписаны меры по обеспечению безопасности здоровья.

2.1 Анализ негативных факторов, воздействующих на оператора ПЭВМ и способы их устранения

Длительное использование компьютера далеко не безобидно для организма. Отсутствие движения, постоянное нахождение в одном положении и огромная нагрузка на зрительный аппарат. Все эти факторы со временем наносят вред состоянию здоровья.

Для того что бы сократить вредное влияние компьютера на здоровье, следует использовать следующие правила:

- Найдите удобный стул и настройте его под себя;
- монитор должен располагаться на уровне глаз;
- исключить блики и тени на дисплее добавив свет;
- устраивайте перерывы в промежутке 45–60;
- ограничить рабочий день в 6–8 часов;
- соблюдение осанки на протяжении рабочего дня.

2.1.1 Оборудование рабочего места

Ниже пользователь может изучить основные требования для правильного рабочего места:

- Дистанция от глаза до дисплея должна быть не менее 600мм и не более 700мм;
- дисплей должен быть чуть ниже уровня глаз;
- у стола должны присутствовать регулировки в диапазоне 650–

800мм;

- клавиатуру размещают на расстоянии 100–300мм от края свободного пространства;

- рабочий стол должен иметь пространство для ног высотой не менее 600 мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног – не менее 650 мм [6].

На рисунке 76 можно ознакомиться с правильно организованным рабочем местом.

Если были выполнены все ранее указанные требования, то рабочее место должно выглядеть, как на рисунке 77.

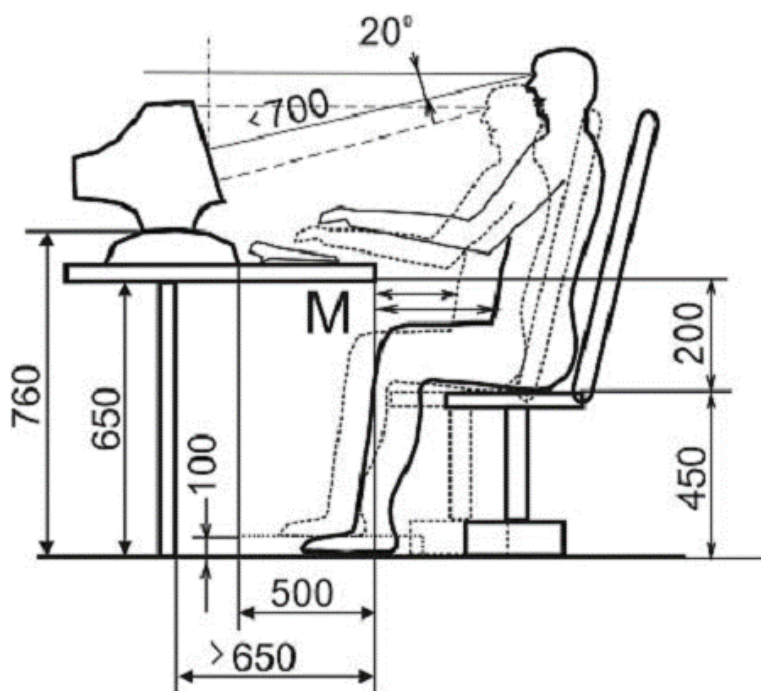


Рисунок 107 – Организация рабочего места

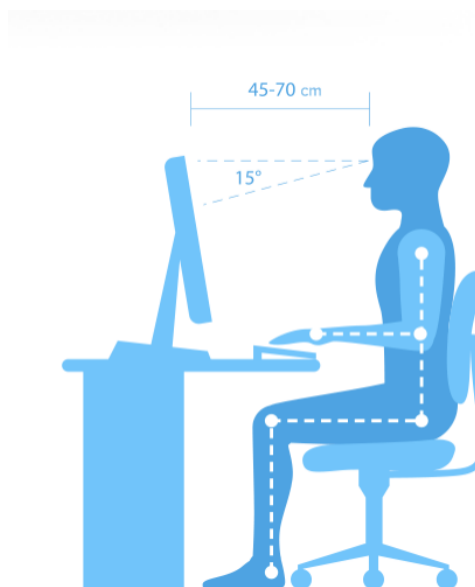


Рисунок 108 – Правильная посадка за ПК

При посадке за компьютер должны соблюдаться следующие требования:

- Запястья должны быть прямыми;
- плечи не должны быть напряжены;
- локти должны быть на всей ширине стола.

2.1.2 Освещенность рабочего места

Нормы освещения задаются исходя из количества зрительного контакта, если речь идёт об местах оборудованных дисплеями, то нужно соблюдать следующие значения: при комбинированной схеме – 400 люкс, а при общей – 200 люкс.

Если у рабочего места недостаточная или не правильная освещенность, то это может повлечь за собой различного рода последствия, одним из является нарушение зрения. Что бы этого не произошло требуется обеспечить правильный и достаточный свет для предотвращения вреда здоровью.

В основном в качестве источника света используют потолочные светильники на люминесцентных лампах. Для предотвращения бликов можно использовать настольные лампы.

2.1.3 Электробезопасность

К электробезопасности предъявляются требования по ГОСТ 12.1.019–2017 «ЭЛЕКТРОБЕЗОПАСНОСТЬ».

Компьютер – это совокупность устройств, запитанных от сети переменного тока с частотой 50Гц и напряжением 230 В. Такое напряжение несёт в себе угрозу для жизни человека, именно поэтому требуется соблюдать правила при работе с компьютером для сохранения безопасности:

- Питание всего оборудования происходит только от одной фазы;
- питание устройств должно осуществляться исключительно через розетки европейского стандарта с наличием заземления, использование различных переходников не допускается;
- розетки, используемые для подключения компьютерной техники, должны быть обозначены с указанием допустимого напряжения;
- номинальное напряжение электросети следует обозначать на стене или специальном щите рядом с розеткой, используя яркую краску и крупный шрифт.

Каждый сотрудник использующий компьютер обязан соблюдать правила электробезопасности поскольку их игнорирование ведет к опасным последствиям. Требуется проверять в каком состоянии находится проводка а так же снять напряжение если у устройства какие-то проблемы.

2.1.4 Пожарная безопасность

При работе с компьютером или любым другим электрооборудованием пожары чаще всего возникают из-за:

- Проблем с проводкой;
- возгорание горючих материалов;
- плохой контакт.

Пожарные меры профилактики проводятся согласно федеральному

закону от 21.12.1994 №69–ФЗ (ред. от 28.09.2010) «О пожарной безопасности». Что бы обеспечить безопасность на предприятиях следует выполнять ряд мер по предостережению воспламенений. Что бы повысить противопожарную безопасность:

- Специально предназначенные помещения должны быть всегда открыты;
- доступ к устройствам пожаротушения должен быть открыт всегда;
- должно быть реализовано отключение вентиляции при обнаружении пожара;
- обязательно наличие пожарной сигнализации.

2.1.5 Режим труда и отдыха

Каждый работник чья деятельность связана с компьютером должен соблюдать установленный режим труда и отдыха. Соблюдение данного режима обязательно для поддержки здоровья. Частота и продолжительность перерывов назначается в соответствии с видом выполняемой работы. Эффективность работы напрямую зависит от правильно составленного графика отдыха.

Принято выделять следующие виды трудовой деятельности, связанные с работой за компьютером:

- Группа А – работа, где основная нагрузка связана со считыванием информации с дисплея;
- группа Б – работа, связанная с вводом информации в систему;
- группа В – творческая деятельность при работе с компьютером.

При выполнении трудовых обязанностей связанных с использованием компьютеров, основной работой с ПЭВМ признается вид деятельности, который занимает не менее 50% рабочего времени в течение смены или полного рабочего дня.

Для видов трудовой деятельности выделяют 3 основные категории тяжести работы с компьютером которые определяются следующим образом:

- группа А – учитывается общее количество знаков, прочитанных за рабочую смену, максимально допустимый объём — 60 000 знаков за смену;
- группа Б – учитывается итоговое количество знаков, как вводимых, так и считываемых в течение всей смены, предельный объём составляет – 40 000 знаков;
- группа В – учёт ведется не от количества введенных знаков, а от времени работы с компьютером, максимальное время работы с компьютером – 6 часов.

Суммарное время регламентированных перерывов для каждой из указанных категорий, указаны в таблице 5.

Таблица 5– Время перерывов

Категория работы с ПЭВМ или ВДТ	Уровень нагрузки за рабочую смену при видах работы с ВДТ			Суммарное время регламентированных перерывов, мин	
	группа А, количество знаков	группа Б, количество знаков	группа В час	при 8– часовой смене	при 12– часовой смене
1	До 20000	До 15000	До 2	50	80
2	До 40000	До 30000	До 4	70	110
3	До 60000	До 40000	До 6	90	140

2.2 Меры безопасности при работе с ПЛК ОВЕН ПР200 и панелью оператора ОВЕН СП270

Во время эксплуатации, технического обслуживания и поверки прибора следует соблюдать требования ГОСТ 12.3.019–80,

«Правил эксплуатации электроустановок потребителей» и «Правил охраны труда при эксплуатации электроустановок» [5].

Любые подключения к прибору и работы по его техническому обслуживанию следует производить только при отключенном питании прибора и подключенных к нему устройств [5].

Попадание влаги на контакты или внутренние компоненты прибора не допускается, также запрещено использование прибора в агрессивных средах где в атмосфере присутствуют пары агрессивных кислот.

Панель оператора работает с максимальным напряжением до 26 В, что составляет безопасное значение для жизни человека. По способу защиты соответствует классу III.

Любые работы по подключению и техническому обслуживанию панели необходимо производить только при отключенном питании и отсутствии напряжения в линиях связи [4].

При проверке панели необходимо соблюдать требования ГОСТ 12.3.019, «Правил эксплуатации электроустановок потребителей» и «Правил техники безопасности при эксплуатации электроустановок потребителей» [4].

3 Экономика

3.1 Цель и задачи экономического обоснования

Целью экономической части дипломной работы является оценка затрат на разработку и внедрение системы управления поточной линией производства.

Для достижения цели решаются следующие задачи:

- Расчёт себестоимости разработки ПО а также его настройки;
- Определение затрат на создание системы;
- оценка текущих расходов;
- расчёт годового экономического эффекта;
- определение срока окупаемости.

3.2 Расчет затрат на разработку системы

Разработка системы включает в себя:

- Проектирование логики управления на языке FBD в среде OwenLogic;
- конфигурирование ПЛК;
- создание графической панели оператора;
- настройка взаимодействия ПЛК;
- тестирование на лабораторном стенде;
- подготовка технической документации.

3.2.1 Трудоемкость разработки

Таблица 6– Трудоемкость разработки

Вид работ	Трудоемкость, чел. часов
Анализ требований и обзор аналогов	10
Разработка алгоритмов управления	25
Конфигурирование ПЛК	14

Создание экранов визуализации	22
Тестирование, отладка, симуляция	45
Создание документации	12
Итого	128 чел.–часов

С учетом того, что рабочий день длится 8 часов, трудоемкость будет составлять 16 рабочих дней.

3.2.2 Расчет заработной платы разработчика

Должность разработчика: инженер–программист АСУ ТП.

Месячная ставка составляет 82000 рублей.

Среднедневная ставка составляет: $82000 / 22 = 3727$ рублей.

Часовая ставка инженера–программиста в час, будет составлять 466 рублей.

Основная заработная плата разработчика составляет 59648 рублей.

Дополнительная заработная плата разработчика будет составлять 8947 рублей.

Важно учитывать страховые взносы, которые составляют 20578 рублей.

По итогам расчетов, затраты на оплату труда инженеру–программисту будут составлять 89173 рублей.

3.2.3 Стоимость оборудования и комплектующих

Таблица 7– Стоимость оборудования и комплектующих

Оборудование	Цена за ед., руб.	Кол–во	Сумма, руб.
ПЛК ОВЕН	30 000	1	30 000
Панель оператора	42 000	1	42 000
Компоненты для сборки стенда	70 000	1	70 000

Итого	оборудование
142000	

3.2.4 Прочие затраты

Таблица 8 – Прочие затраты

Позиции	Сумма, руб.
Электроэнергия на время разработки	2 000
Настройка ПЛК	5 000
Расходные материалы	1 000
Использование ПО OwenLogic	0
Итого прочие затраты	8 000

3.2.5 Полная себестоимость разработки

Таблица 9 – Полная себестоимость разработки

Позиция	Сумма, руб.
Оплата труда разработчика	89173
Оборудование и комплектующие	142000
Прочие затраты	8 000
Итого (себестоимость)	239173

3.3 Капитальные затраты на внедрение

Для того что бы начать внедрение системы потребуется закупка оборудования, его монтаж, настройка, а также обучение оператора.

Таблица 10 – Капитальные затраты на внедрение

Позиция	Сумма, руб.

Себестоимость разработки	239173
Монтаж и настройка	45 500
Обучение одного оператора (2 дня)	4 000
Капитальные затраты	288673

3.4 Расчет эксплуатационных расходов

При расчёте эксплуатационных расходов учитывается электроэнергия, обслуживание и т.д

Таблица 11 – Расчет эксплуатационных расходов

Позиция	Формула/ значение	Сумма, руб. / год
Электроэнергия системы управления	$(0,4 \text{ кВт}) * 5 \text{ ч} * 180 \text{ дн} * 10 \text{ руб./кВт*ч}$	3600
Обслуживание оборудования	$0,02 * 350000$	7000
Амортизация оборудования	$350000 / 7$	50000
Ремонтный фонд	$0,015 * 350000$	3 758
Итого эксплуатационных затрат		65850 руб./ год

3.5 Оценка экономической эффективности

3.5.1 Исходная ситуация

При полу ручном управлении:

– Управление включением или выключением электроприводов осуществляется оператором вручную, без задержек и блокировок. За счёт этого возможны различные ошибки, например: одновременный пуск обоих конвейеров или преждевременное отключение наклонного конвейера;

– отсутствие автоматической задержки в 20 секунд перед остановкой наклонного конвейера, за счёт этого возможно завалы на месте соединения конвейеров;

– при использовании ручного аварийного стопа, оператор не всегда успевает остановить конвейер в нужный момент, из-за этого возможны поломки приводных роликов и других компонентов.

В результате всего перечисленного появляются:

– Перерасход энергии составляет до 25% из некорректных режимов пуска ;

– время на обкатку и настройку линии, на 30% дольше из-за использования ручного контроля;

– отсутствие автоматической аварийной остановки, в среднем за год происходит 4 аварии.

– Средняя стоимость ремонта одной аварии составляет 50000 рублей.

В результате всего этого, эксплуатационные потери в год при стандартном способе использования составляют около 200000 рублей.

3.5.2 После внедрения

– Автоматическое управление по алгоритму ТЗ;

– экономия электричества (около 25–30%) за счёт исключения перегрузок и иных факторов;

– автоматическое контролирование параметров двигателей, автоматическая защита от аварий;

– снижение числа аварий с 4 до 1, а также снижение затрат на ремонт с 50000 рублей до 15000 рублей.

3.5.3 Расчёт годовой экономии

Таблица 12 – Расчёт годовой экономии

					ПГУ 1.090501.03 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		73

Показатель	Без автоматизации (руб./год)	С автоматизацией (руб./год)
Потери от брака или аварий	$4 * 50\,000 = 200\,000$	$1 * 50\,000 = 50\,000$
Электроэнергия электродвигателя	$850\,000 / 0,7 = 1\,214\,286$	850 000
Электроэнергия системы управления	3 500	3 500
Фонд оплаты труда оператора	$1 \text{ чел} * 55\,000 * 12 = 660\,000$	$1 \text{ чел} * 48\,000 * 12 = 576\,000$
Простои, настройка (из-за аварий)	70 000	15 000
Итого прямых затрат	2 147 786	1 494 500

Годовая экономия будет составлять 653 286 рублей.

3.5.4 Чистый годовой экономический эффект

Годовые затраты после внедрения автоматизации будут составлять:

- Эксплуатационные расходы – 813 522 рублей;
- фонд оплаты труда оператора – 528 000 рублей;
- потери от аварий – 22 500 рублей;
- простои и настройка – 10 000 рублей.

Итог после внедрения будет составлять – 1 374 022 рублей в год.

Итог до внедрения будет составлять – 1 837 305 рублей в год.

Чистый годовой экономический эффект будет составлять – 463 283 рублей в год.

Таким образом, внедрение системы позволит экономить 463 283 рублей ежегодно.

3.5.5 Срок окупаемости проекта

Срок окупаемости (PP) = Капитальные затраты / Чистый годовой эффект

$$PP = 288\,673 / 463\,283 = 0,62 \text{ года (примерно 7 месяцев)}$$

3.6 Анализ чувствительности

Таблица 13 – Анализ чувствительности

Сценарий	Чистый эффект, руб./год	Срок окупаемости, лет
Оптимистичный	504 000	0,83
Базовый	420 000	1
Пессимистичный	294 000	1,43

Учитывая все 3 представленных сценария, срок окупаемости проекта остается на высоком уровне и показывает отличный срок окупаемости.

3.7 Вывод по экономической части

Таким образом, разработанная система управления поточной линией производства является эффективной и экономически целесообразной системой. Рекомендуется к внедрению на предприятиях различного назначения.

Заключение

В ходе выполнения задания были приобретены навыки работы с контроллером ПР200, а также с панелью оператора. За время работы были изучены основные понятия и технологии. Разработка производилась в среде программирования OwenLogic. Программирование производилось на языке функциональных блоков

Со временем проект может быть доработан и использоваться на предприятиях с потребностью к данной системе.

					ПГУ 1.090501.03 ПЗ	Лист
						76
Изм.	Лист	№ докум.	Подпись	Дата		

Список использованных источников

1. Конфигурирование и моделирование работы основных элементов промышленной автоматики: учеб.–метод. пособие / Д. А. Тарасов, А. Е. Шаралапов, К. И. Никишин, М. А. Митрохин. – Пенза: Изд–во ПГУ, 2021. – 166 с.

2. RS485 – стандарт промышленных сетей. Какими особенностями обладает микросхема приемопередатчика? [Электронный ресурс] / Режим доступа: <https://habr.com/ru/companies/milandr/articles/540084/> свободный. (дата обращения: 01.05.2026).

3. OWEN Logic. Руководство пользователя. ООО «Производственное Объединение ОВЕН». [Электронный ресурс] / Режим доступа: https://docs.owen.ru/product/programmnoe_obespechenie_owen_logic/986/83304#topic-83304 свободный. (дата обращения: 26.04.26).

4. Панель оператора СП270. Паспорт и руководство по эксплуатации КУВФ.421449.003 РЭ. ООО «Производственное Объединение ОВЕН». [Электронный ресурс] / Режим доступа: https://www.owen.ru/uploads/re_sp270_1312.pdf свободный. (дата обращения: 26.04.26).

5. ПР200. Устройство управляющее многофункциональное. Руководство по эксплуатации. ООО «Производственное Объединение ОВЕН». [Электронный ресурс] / Режим доступа: <https://docs.owen.ru/product/pr200/80/79940#topic-79940> свободный (дата обращения: 15.05.26).

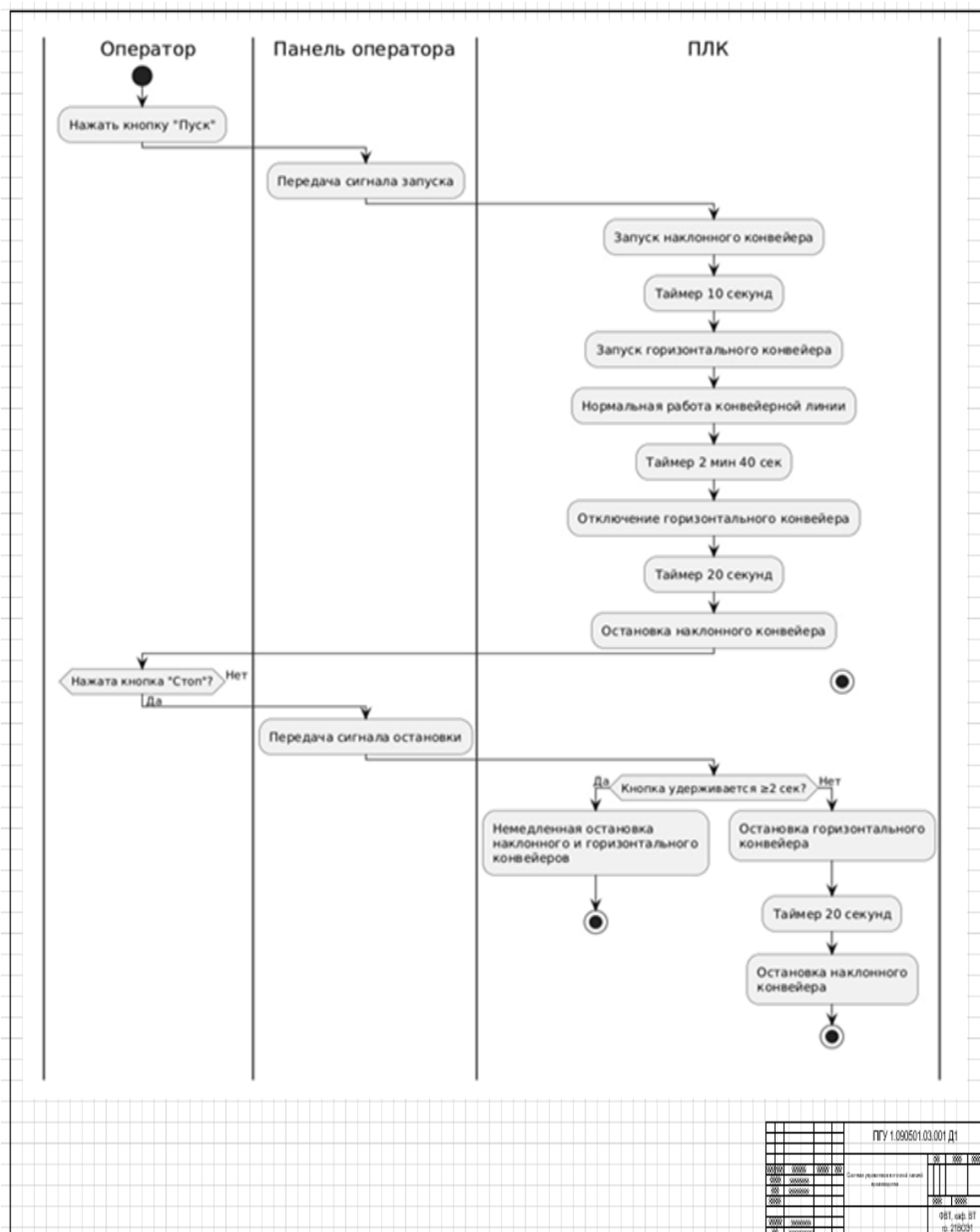
6. Требования по обустройству помещений и рабочих мест, предназначенных для работы на компьютере и прочей офисной технике [Электронный ресурс] / Режим доступа: https://www.consultant.ru/document/cons_doc_LAW_18895/df3eaac6ab12789400

2172933f9cd660e61df962/ свободный. (дата обращения: 19.05.26).

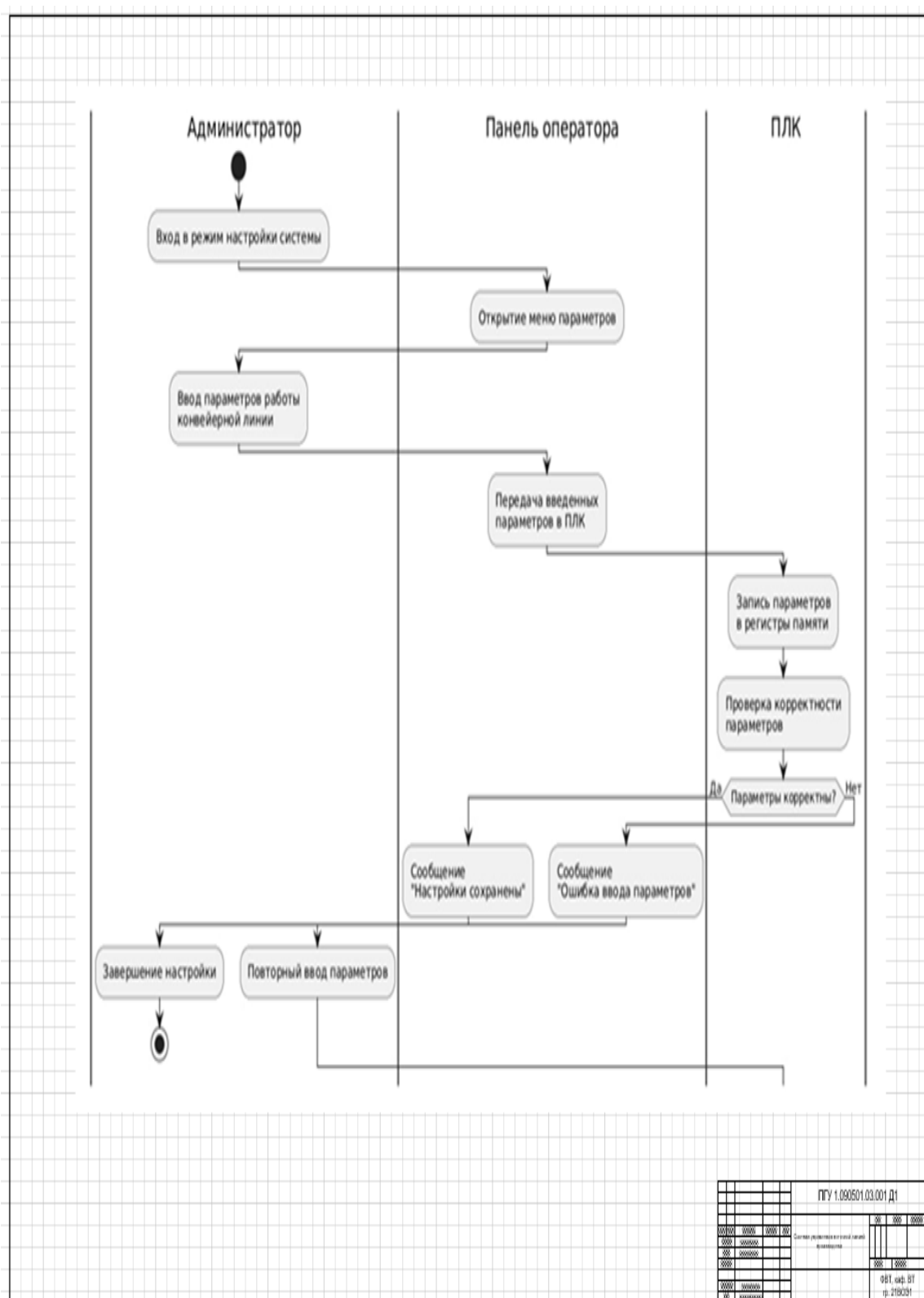
7. ГОСТ IEC 61131-2-2012 Часть 2 Требования к оборудованию и испытания [Электронный ресурс] / Режим доступа: <https://www.reglament.by/wp-content/uploads/docs/gost/gost-iek-61131-2-2012.pdf> свободный. (дата обращения: 14.05.2026)

					ПГУ 1.090501.03 ПЗ	Лист
						78
Изм.	Лист	№ докум.	Подпись	Дата		

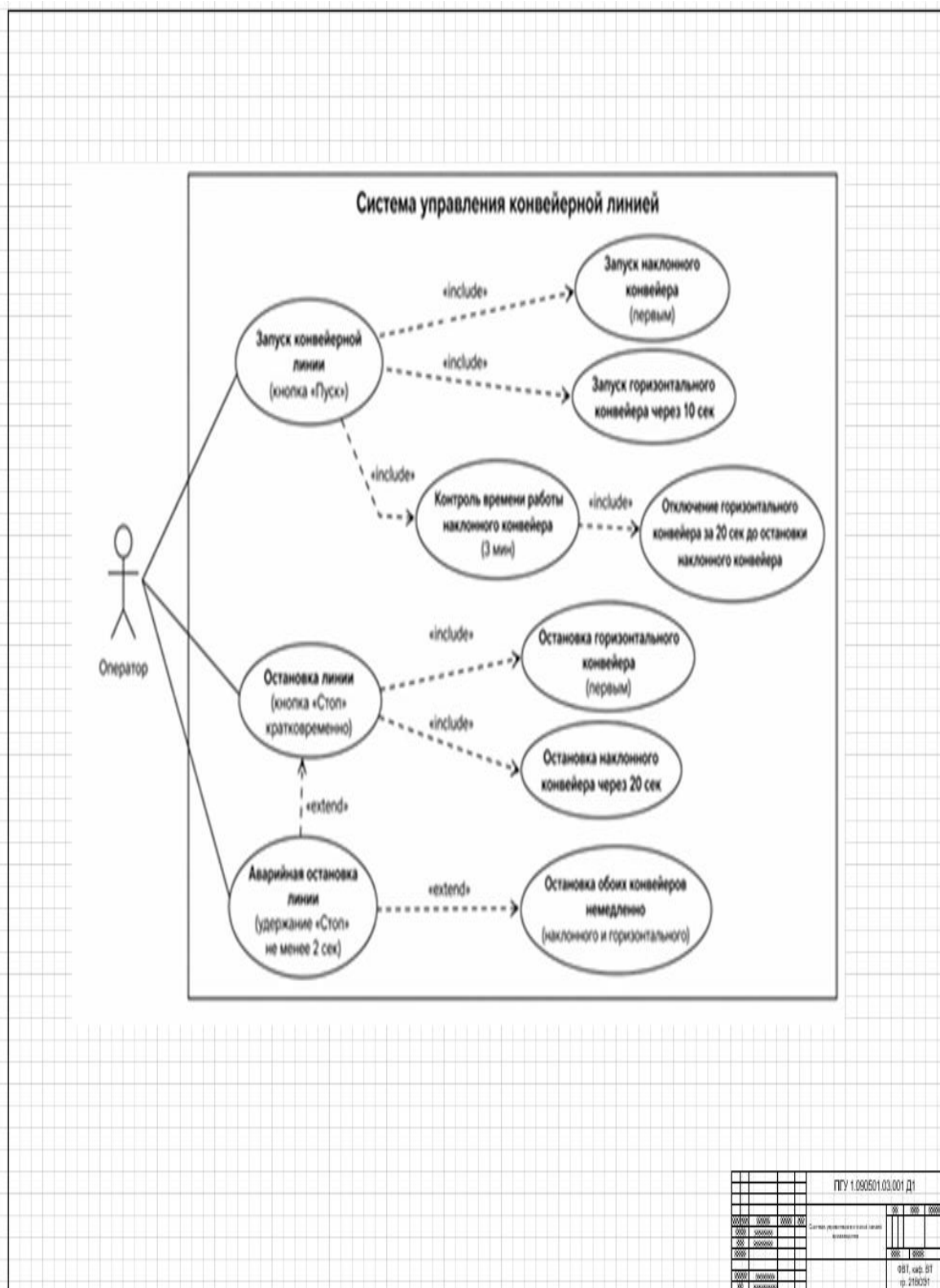
Приложение А (Обязательное) UML–диаграммы



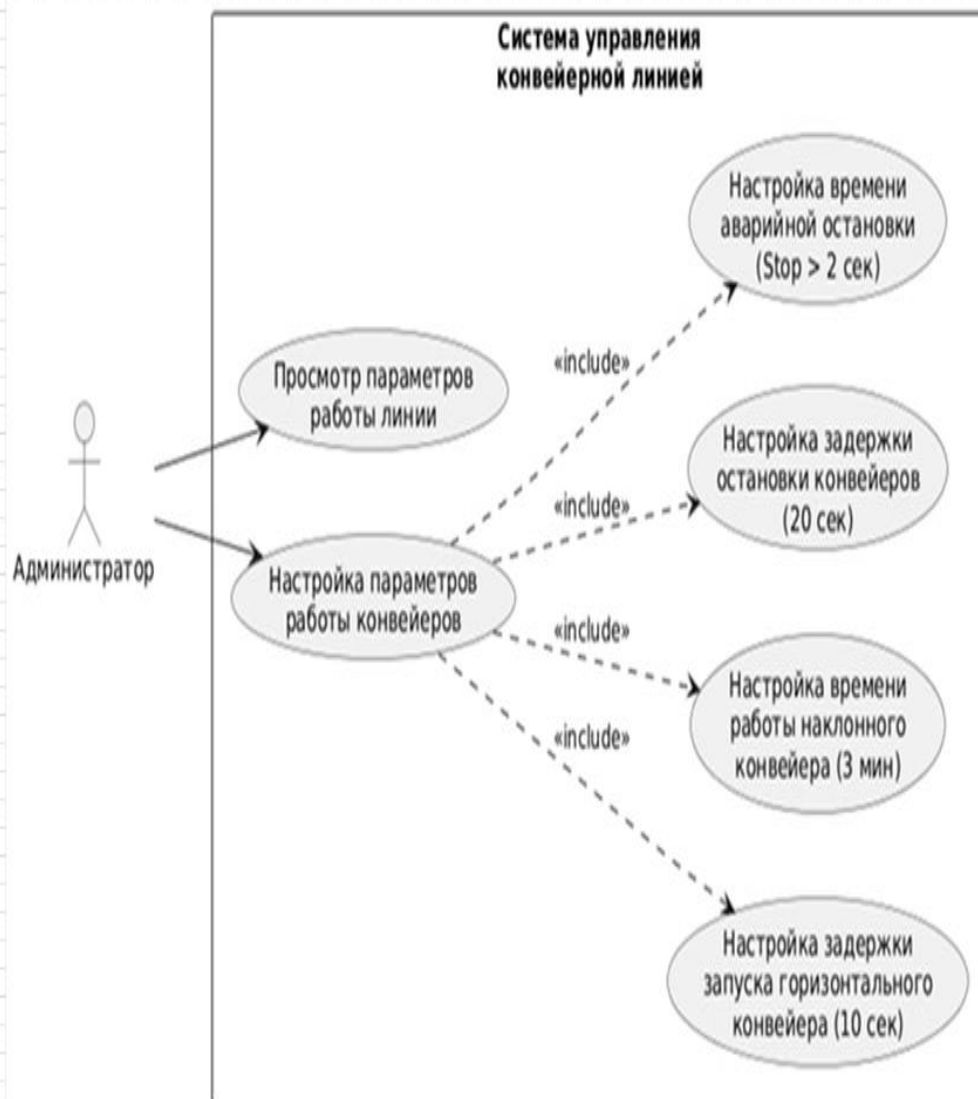
Приложение А.1 – UML–диаграмма деятельности для пользователя



Приложение А.2 – UML–диаграмма деятельности администратора



Приложение А.3 – UML–диаграмма использования для пользователя



ПГУ 1.090501.03.001 Д1			
001001	000000	000000	00
0000	00000000	Счетчик разработанных и тестовых случаев	
000	00000000	всего	
00000		0000	0000
000000	00000000	081, апр 91	
00	0000000000	пр. 2190001	

Приложение А.4 – UML–диаграмма сценариев для администратора

Приложение Б

Схемы функциональных блоков программ

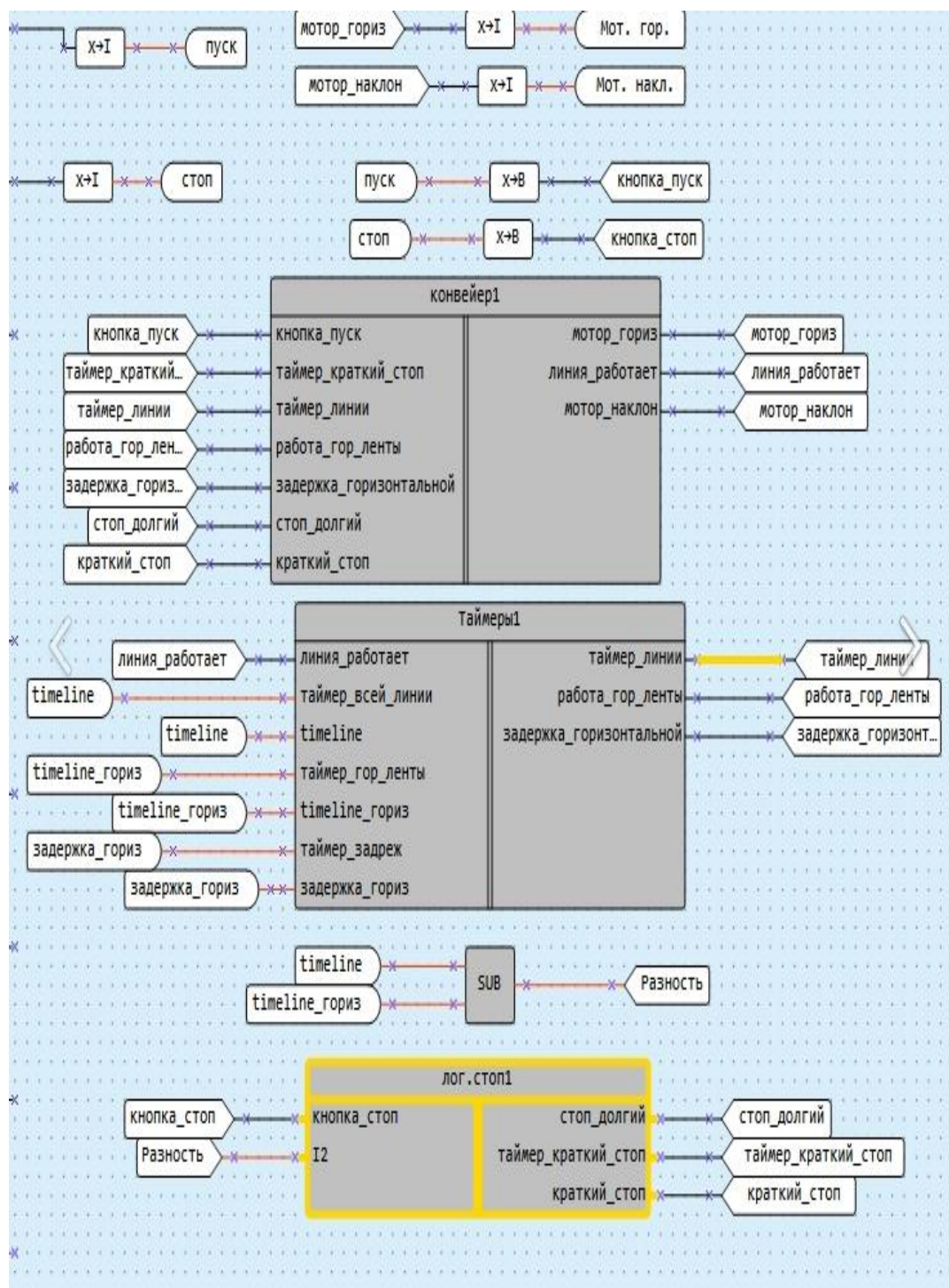


Схема функциональных блоков макроса

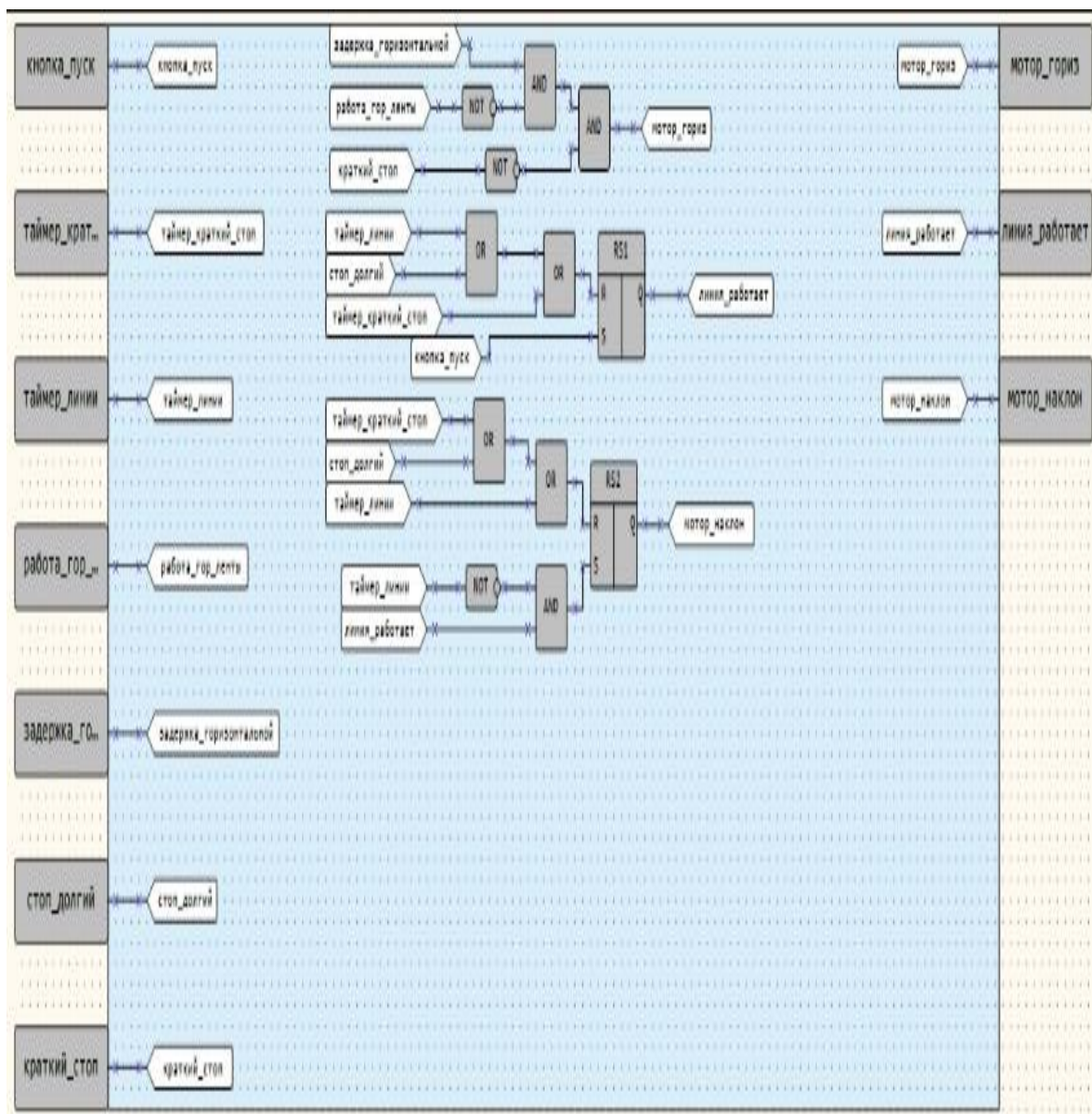


Схема функциональных блоков макроса конвейер

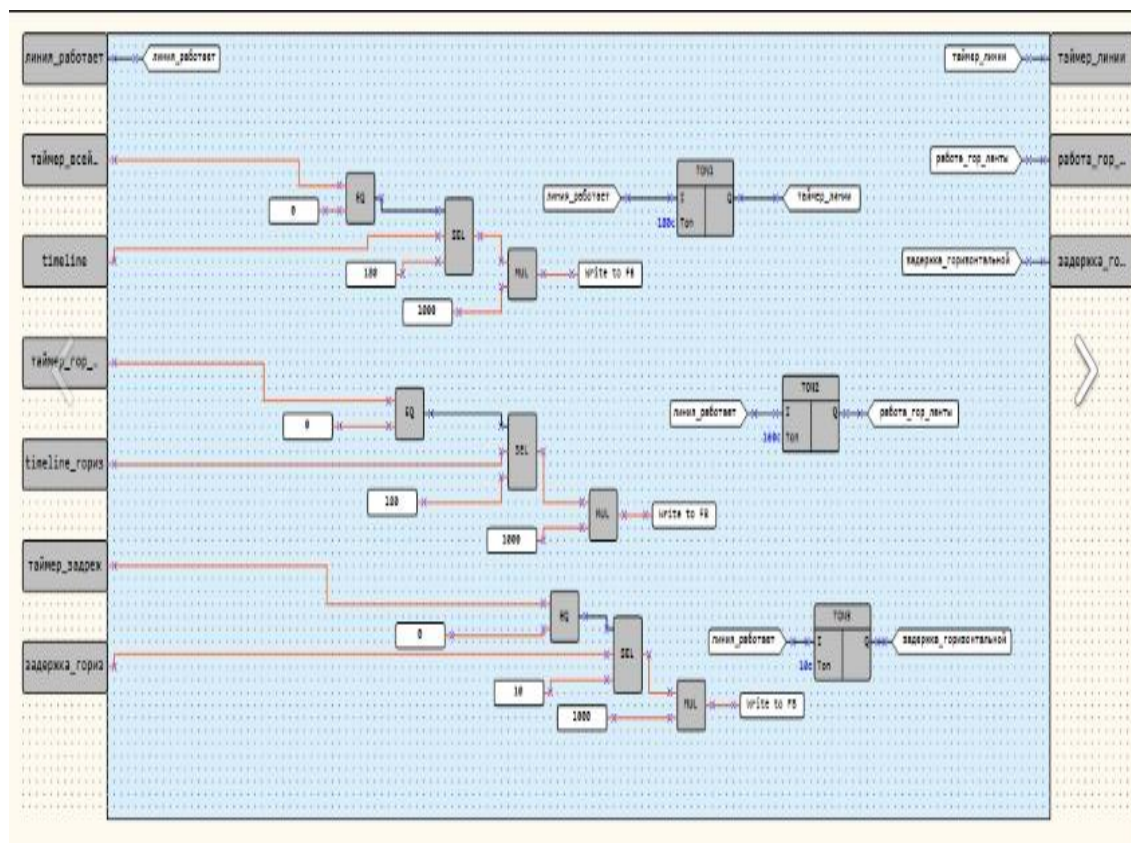


Схема функциональных блоков макроса таймер

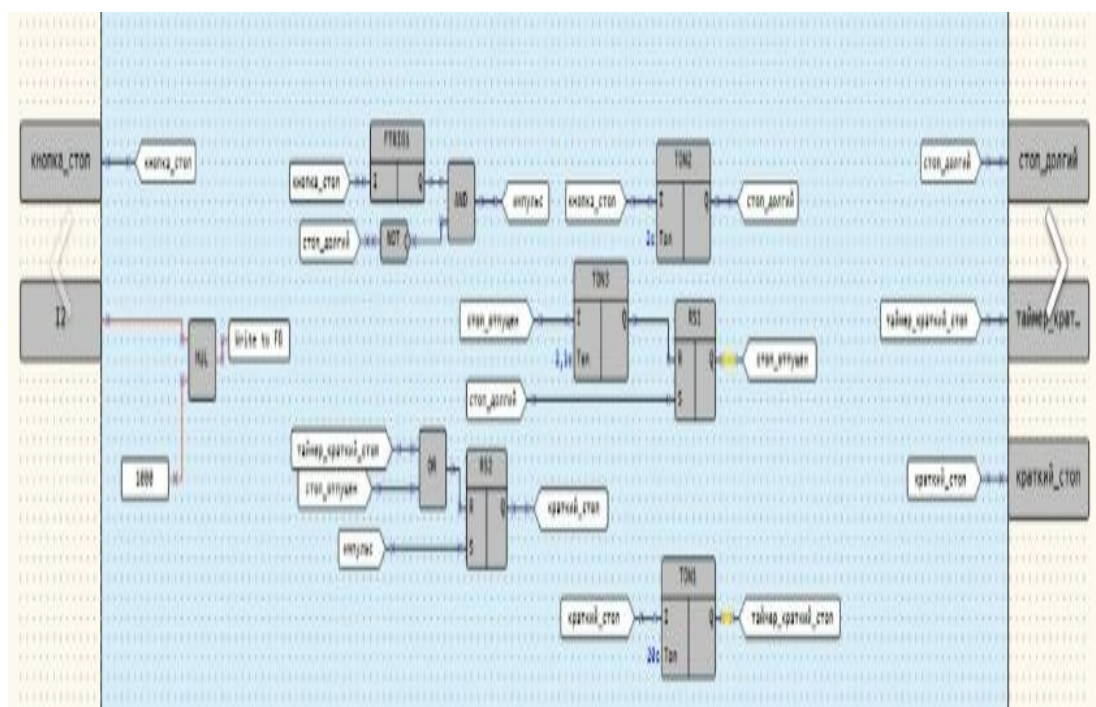


Схема функциональных блоков макроса лог.стоп

Приложение В

(Обязательное)

Программа контроля и управления доступом для кодового замка

Презентация



Пензенский
государственный
университет



Кафедра
вычислительной
техники

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

СИСТЕМА УПРАВЛЕНИЯ ПОТОЧНОЙ ЛИНИЕЙ ПРОИЗВОДСТВА

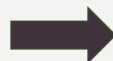
АВТОР РАБОТЫ: СТУДЕНТ ГР. 21ВОЭ1 ЕСЬКОВ М.М.

НАУЧНЫЙ РУКОВОДИТЕЛЬ: ДОЦЕНТ, К.Т.Н. ТАРАСОВ Д.А.

О ПРОГРАММЕ

Проблема

- На производстве, а также в иных неблагоприятных условиях требуется система поточной линии производства для перемещения тяжелых предметов



Разработка

Программа управления
поточной линией производства
для ПЛК ОВЕН PR200

ЦЕЛИ И ЗАДАЧИ

В процессе разработки был определен следующий функционал для реализации:

- Последовательный пуск;
- Цикл нормальной работы;
- Остановка по кратковременному «Стоп»;
- Экстренная остановка обоих конвейеров
- Управление с пульта.

ПОСТАНОВКА ЗАДАЧИ

Взаимодействие с пользователем



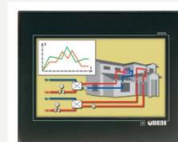
Взаимодействие с администратором



ВЫБОР РЕШЕНИЯ



- Язык разработки:
 - Язык функциональных блоков FBD
- Среда разработки:
 - OWEN Logic
 - Конфигуратор СП200
- Целевая платформа:
 - ПЛК OWEN PR200
 - Панель оператор OWEN СП270



- Другие ресурсы разработки: ПК, ОС Microsoft Windows 11, драйверы для взаимодействия элементов целевой платформы и ПК

ПРОЦЕСС РАЗРАБОТКИ

Диаграмма взаимодействия с пользователем

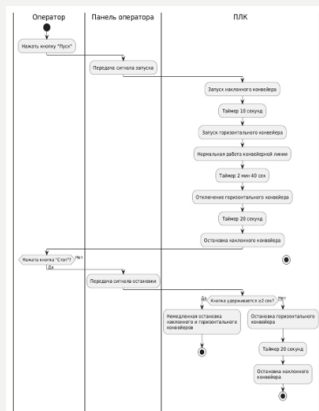
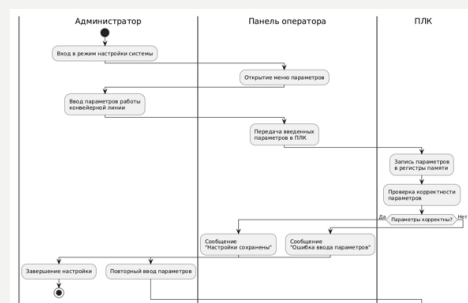
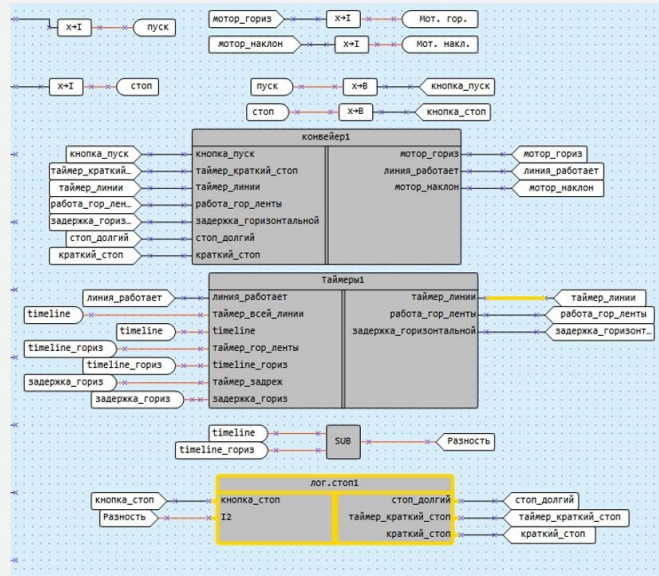


Диаграмма взаимодействия с администратором



Итоговая схема управляющей программы



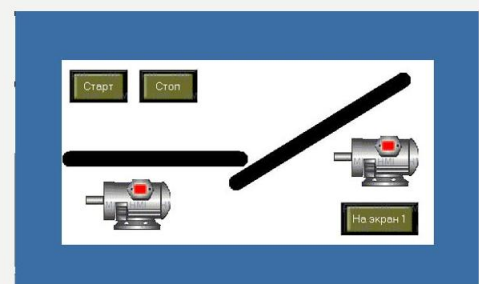
Интерфейс панели оператора

Экран задания уставок таймеров

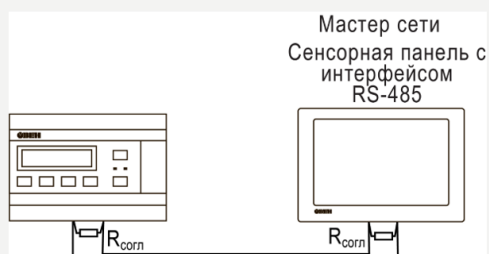
Таймер всей линии Таймер горизонтальной линии

Таймер задержки включения горизонтальной линии

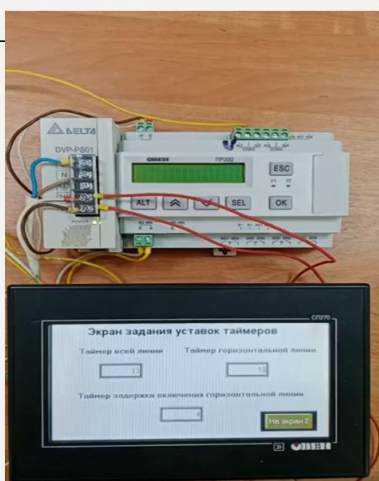
 На экран 2



МОДЕЛЬ ВЗАИМОДЕЙСТВИЯ УСТРОЙСТВ ЦЕЛЕВОЙ ПЛАТФОРМЫ



Тестирование на стенде



ЗАКЛЮЧЕНИЕ

- В рамках разработки проекта были выявлены недостатки существующих аналогов и предложен свой вариант решения в виде реализации текущего проекта.
- Итоговый вариант работоспособен и пригоден для внедрения на производство. В дальнейшем проект может быть расширен путем добавления полноценной панели оператора для администратора, добавлением датчиков и подключением к промышленной сети.

					ПГУ 1.090501.03 ПЗ	Лист
						91
Изм.	Лист	№ докум.	Подпись	Дата		