МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет	*Кафедра*
вычислительной техники	«Вычислительная техника»

Направление подготовки: 09.03.01 «Информатика и вычислительная техника»

Профиль подготовки: «Вычислительные машины, комплексы, системы и сети»

БАКАЛАВРСКАЯ РАБОТА

на тему

ПРОГРАММА УПРАВЛЕНИЯ ПРИВОДОМ ПРОДОЛЬНОГО ПЕРЕМЕЩЕНИЯ СУППОРТА ТОКАРНОГО СТАНКА

Студент 2.06.2025 (подпись, дата) Киреев Д.А.

Руководитель 2.06.2025 (подпись, дата) Тарасов Д.А.

Нормоконтролёр 2.06.2025 (подпись, дата) Бычков А.С.

Работа допущена к защите (протокол заседания кафедры от 28.05.25 г. № 16)

Заведующий кафедрой _____ (подпись) Митрохин М.А.

Работа защищена с отметкой Хорошо (протокол заседания ГЭК от 09.06.2025 г. № 9)

Секретарь ГЭК _____ (подпись) Гурин Е.И.

Пенза 2025

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

Факультет	*Кафедра*
вычислительной техники	«Вычислительная техника»

«Утверждаю»
Заведующий кафедрой
_____М.А. Митрохин
«06» 05 2025г.

ЗАДАНИЕ
НА БАКАЛАВРСКУЮ РАБОТУ

1. Студент Киреев Данила Андреевич гр. 21ВВВ1 факультета вычислительной техники.

2. Тема работы «Программа управления приводом продольного перемещения суппорта токарного станка» Тема утверждена приказом ПГУ №601/о от "06" мая 2025 г.

3. Руководитель бакалаврской работы к.т.н., доцент Тарасов Д.А.

4. Задание на работу (назначение разработки, исходные данные и т.п.)

Разработать программу для контроллера, управляющую преобразователем частоты, а тот в сою очередь приводом продольного перемещения суппорта токарного станка, который совершает возвратно-поступательное движение. Вся программа создается на базе ПЛК ОВЕН ПР200 и панели оператора ОВЕН СП270, с использованием преобразователя частоты ВЕСПЕР Е2 – 8330. Объектом разработки является управляющая программа на базе ПЛК, которая должна обеспечивать перемещение суппорта на заданной рабочей скорости, которое начинается по кратковременному нажатию кнопки «ПУСК/СТОП», остановка – по срабатыванию одного из концевых выключателей. Задание команд управления следует осуществлять посредством воздействия на элементы графического интерфейса экрана панели оператора.

5. Перечень подлежащих разработке вопросов

Требуется произвести анализ предметной области, изучить аналоги существующих систем управления токарных станков, способы разработки программ управления, изучить способы проектирования подключаемых устройств.

6. Календарный график выполнения работы

№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1	Изучение предметной области	15%	06.05.2025-07.05.2025	
2	Анализ требований	4%	08.05.2025-09.05.2025	
3	Выбор технологий, применяемых при проектировании и разработке	15%	10.05.2025-11.05.2025	
4	Описание алгоритма	13%	12.05.2025-13.05.2025	
5	Разработка управляющей программы	45%	14.05.2025-28.05.2025	
6	Оформление пояснительной записки	8%	29.05.2025-30.05.2025	

Дата выдачи задания "06" мая 2025 г.

Руководитель бакалаврской работы 06.05.2025 Тарасов Д.А.
(подпись, дата)

Задание к исполнению принял студент 06.05.2025 Киреев Д.А.
(подпись, дата)

Работу к защите допустить
Декан факультета 06.06.2025 Фионова Л.Р.
(подпись, дата)

Реферат

Бакалаврская работа содержит 103 листа, 105 рисунков, 4 таблицы, 10 источников, 3 приложения.

ПРОГРАММА УПРАВЛЕНИЯ ПРИВОДОМ ПРОДОЛЬНОГО ПЕРЕМЕЩЕНИЯ СУППОРТА ТОКАРНОГО СТАНКА

Цель работы – разработать программу для контроллера, которая управляет преобразователем частоты, а тот в свою очередь приводом продольного перемещения суппорта токарного станка, который совершает возвратно-поступательное движение. Вся программа создается на базе ПЛК ОВЕН ПР200 и панели оператора ОВЕН СП270, с использованием преобразователя частоты ВЕСПЕР Е2 – 8330.

Объектом разработки является управляющая программа на базе ПЛК, которая должна обеспечивать перемещение суппорта на заданной рабочей скорости, которое начинается по кратковременному нажатию кнопки «ПУСК/СТОП», остановка – по срабатыванию одного из концевых выключателей. Задание команд управления следует осуществлять посредством воздействия на элементы графического интерфейса экрана панели оператора.

Разработка проводилась в среде программирования OWEN Logic и в Конфигураторе СП200. В качестве языка программирования был выбран язык функциональных блоков FBD.

В результате проведенной работы разработана программа, соответствующая требованиям поставленной задачи, а именно: интуитивно понятный пользовательский интерфейс и контроль над перемещением привода продольного перемещения суппорта токарного станка.

					ПГУ 1.090301.10.001 ПЗ		
Изм.	Лист	№ докум.	Подпись	Дата			
Разраб.		Киреев Д.А.		2.06.25	*Программа управления приводом продольного перемещения суппорта токарного станка Пояснительная записка*	Лит.	Лист 2 / Листов 104
Провер.		Тарасов. Д.А.		2.06.25			
Н. Контр		Бычков А.С.		2.06.25		*ФВТ, каф ВТ гр. 21ВВВ1*	
Утверд.		Митрохин М.А.		2.06.25			

Содержание

Перечень сокращений и обозначений 5
Введение 6
1 Программа контроля и управления доступом для кодового замка 8
1.2 Анализ предметной области 8
1.2.1 Обзор аналогов 9
1.2.2 Анализ требований к разработке 10
1.2.3 Выбор архитектуры программы 11
1.2.4 Выбор языка программирования 15
1.2.5 Выбор среды разработки 17
1.3 Проектирование и разработка 31
1.3.1 Конфигурирование ПЛК ОВЕН ПР200 31
1.3.2 Разработка управляющей программы в среде OWEN Logic 37
1.3.3 Конфигурирование преобразователя частоты ВЕСПЕР Е2-8300 53
1.3.4 Конфигурирование панели оператора ОВЕН СП270 62
1.4 Анализ результатов 72
1.4.1 Тестирование и отладка проекта в среде OWEN Logic 72
1.4.2 Тестирование на физическом оборудовании 75
2 Охрана труда 81
2.1 Требования к организации рабочих мест пользователей ПЭВМ 81
2.1.1 Оборудование рабочего места 81
2.1.2 Освещенность рабочего места 83
2.1.3 Электробезопасность 83
2.1.4 Микроклимат и шум 83
Заключение 84
Список использованных источников 85
Приложение А – UML – диаграммы 87
Приложение А.1 – UML – диаграмма деятельности 88
Приложение А.2 – UML – диаграмма компонентов 89

Приложение А.3 – UML – диаграмма последовательностей 90
Приложение А.4 – UML–диаграмма развертывания 91
Приложение Б – Схемы функциональных блоков программ.....92
Приложение В – Презентация.....97

Перечень сокращений и обозначений

В настоящей работе применяют следующие сокращения и обозначения

ОВЕН СП270 – панель оператора фирмы ОВЕН модель СП270

ПЗУ – постоянное запоминающее устройство

ПЧ – преобразователь частоты

ФБ – функциональный блок

ПК – персональный компьютер

ПЛК – программируемый логический контроллер

ПЛК ОВЕН ПР200 – программируемый логический контроллер фирмы ОВЕН модель ПР200

ПЧ ВЕСПЕР Е2 – 8330 – преобразователь частоты фирмы ВЕСПЕР модель Е2 – 8330

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

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

Введение

В современной промышленности токарные станки играют ключевую роль в процессах механической обработки деталей. Они позволяют изготавливать сложные формы с высокой точностью и минимальными затратами времени. Одним из важных факторов, определяющих эффективность работы токарных станков, является система управления, которая обеспечивает точное выполнение технологических процессов. В данной работе рассматривается программа управления приводом продольного перемещения суппорта токарного станка, что является актуальной темой в свете современного подхода к автоматизации производственных процессов.

Привод продольного перемещения суппорта — это механизм, который обеспечивает выполнение возвратно-поступательных движений. Такие движения необходимы для обработки деталей, находящихся на обрабатывающей системе станка. Эффективность этого привода во многом зависит от качественной работы автоматизированной системы управления. Важнейшей задачей этой системы является поддержание заданных параметров перемещения, такие как скорость, плавность и точность, что в свою очередь значительно влияет на качество обрабатываемых деталей.

В рамках данной работы будет разработан программный комплекс, который будет управлять преобразователем частоты, а тот, в свою очередь, будет контролировать работу привода. Преобразователь частоты позволяет изменять частоту и амплитуду электричества, подаваемого на электродвигатель, что ведет к регулировке скорости и момента вращения. Такое управление обеспечивает возможность качественной настройки работы суппорта в зависимости от требований к процессу обработки.

На сегодняшний день, многие устройства и системы управления в токарных станках остаются признанными и надежными, однако технологии в области автоматизации требуют внедрения новых программных решений, которые способны улучшить эксплуатационные характеристики оборудования.

направление на автоматизацию процессов управления привлекло внимание многих исследователей и разработчиков, и, следовательно, является предметом активного изучения.

1 Программа управления приводом продольного перемещения суппорта токарного станка

1.1 Анализ предметной области

Разработка программы управления приводом продольного перемещения суппорта токарного станка является актуальной задачей, способствующей улучшению производственных процессов и повышению их эффективности. Внедрение современных технологий автоматизации откроет новые возможности для оптимизации работы оборудования и снизит затраты на процессы механической обработки, что в свою очередь окажет положительное влияние на конкурентоспособность предприятия в современных условиях. Управляющая программа для ПЛК ОВЕН ПР200, разработанная в данном проекте с учетом технических и программных характеристик данного контроллера, обладает требуемыми возможностями.

Основными пользователями данной программы являются:

– инженеры и операторы станков; они нуждаются в надежной системе управления для точного и эффективного выполнения токарной обработки. Программа позволяет улучшить качество обработки, увеличить производительность и снизить трудозатраты;

– производственные предприятия; компании, занимающиеся механической обработкой, заинтересованы в автоматизации процессов, что способствует снижению издержек и увеличению конкурентоспособности. Эффективное управление движением суппорта значительно ускоряет процесс обработки;

– разработчики оборудования; инженеры-электроники и программисты, разрабатывающие системы управления для токарных станков, требуют документацию и средства для интеграции современных технологий управления, таких как контроллеры и преобразователи частоты.

1.1.1 Обзор аналогов

В настоящее время существует несколько аналогичных систем управления для токарных станков. К ним относятся:

– Siemens Sinumerik; Sinumerik предлагает универсальную систему управления, включающую в себя интегрированные модули управления движением, которые поддерживают плавное регулирование скорости и точную позиционную систему;

– Fanuc; контроллеры Fanuc обеспечивают высокую производительность и надежность работы, а также простоту интеграции с различными приводами и преобразователями частоты;

– Mitsubishi M70; данная система управления поддерживает как простые, так и сложные задачи, обеспечивая высококачественное управление движением.

Таблица 1 – Анализ преимуществ и недостатков конкурентов

Производитель	Преимущество	Недостаток
Siemens Sinumeric	Высокая точность позиционирования, широкие возможности настройки, поддержка различных типов	Высокая стоимость оборудования и лицензий, сложность настройки и программирования, требующая специализированных навыков
Fanuc	Долговечность, простота в использовании интерфейса, широкая распространенность, низкие требования к обслуживанию	Ограниченная поддержка пользовательских сценариев, закрытая архитектура, меньшая гибкость в настройке
Mitsubishi M70	Наличие функций самодиагностики и коррекции ошибок, соотношение цена\качество, гибкие настройки параметров движения	Некоторые ограничения в настройке под специфические задачи, ограниченная документация

Эти решения предоставляют многофункциональные системы управления, однако каждая из них имеет свои плюсы и минусы, что свидетельствует о существовании потребности в разработке индивидуализированного программного обеспечения для конкретных требований.

1.1.2 Анализ требований к разработке

Целью данной дипломной работы является создание программы – алгоритма на базе ПЛК, с использованием преобразователя частоты и панели оператора, которые будут отвечать за эффективное и надежное управление приводом продольного перемещения суппорта токарного станка.

Для достижения этой цели необходимо решить ряд задач, таких как: изучение принципов работы контроллеров, преобразователей частоты, панелей оператора. Разработать программное обеспечения для управления, а также провести тестирование и анализ результатов работы производительности системы.

Требуемый функционал программы управления на ПЛК:

– управление преобразователем частоты, который в свою очередь управляет приводом продольного перемещения суппорта токарного станка, совершая возвратно – поступательные движения;

– перемещение суппорта на заданной рабочей скорости начинается по кратковременному нажатию кнопки «ПУСК/СТОП». Прервать цикл работы возможно в любой момент времени кратковременным нажатием той же кнопки;

– остановка по срабатыванию концевого выключателя. Затем привод стоит в течение заданного промежутка времени и быстро возвращает инструмент (реверс на заданной скорости холостого хода);

– остановка по срабатыванию второго концевого выключателя;

– контроль аварийных ситуаций;

– задание команд управления осуществить посредством воздействия на элементы графического интерфейса экрана панели оператора;

– отображение на экране панели оператора значений эксплуатационных параметров протекающего процесса в контролируемой системе.

В качестве устройств для организации взаимодействия между пользователем и управляющей программой будут использоваться панель оператора ОВЕН СП270, программируемый логический контроллер ОВЕН СП200 и преобразователь частоты ВЕСПЕР Е2 – 8330. Общая схема подключения всех устройств между собой изображена на рисунке 1.

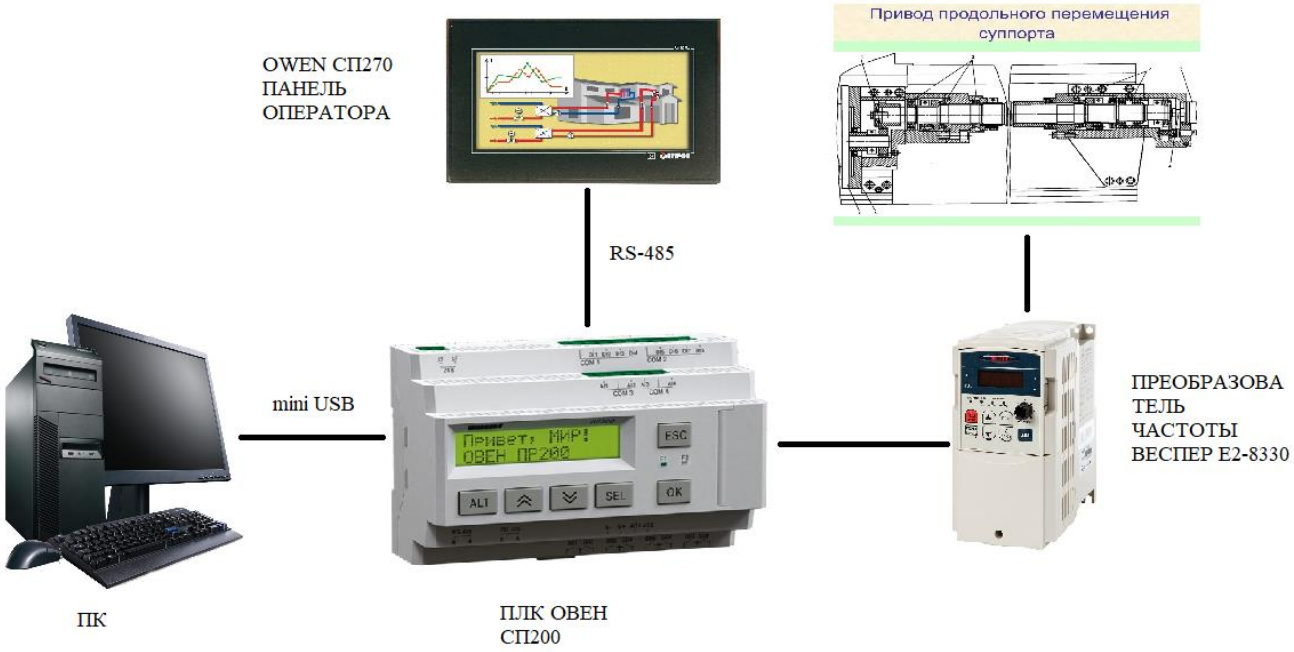


Рисунок 1 – Общая схема подключения устройств проекта

1.1.3 Выбор архитектуры программы

Работу ПЛК в виде последовательно выполняемых шагов:

– значение каждого входа (I1…In) автоматически сохраняется в отдельной ячейке памяти. Для каждого входа предусмотрена своя ячейка.

– программа считывает значения из ячеек памяти входов и выполняет над ними логические операции в соответствии с алгоритмом работы;

– после обработки всей программы результаты записываются на физические выходы прибора (для включения выходных элементов Q1…Q4);

– переход к первому шагу данной последовательности (после выполнения всех предыдущих шагов обработки программы цикл работы прибора повторяется с первого шага) [1].

Архитектура управляющей программы для прибора определяется количеством доступных каналов ввода-вывода. Время цикла программы, выполняемой на ПЛК, увеличивается пропорционально ее сложности. Логика работы программы визуализирована на функциональной схеме, представленной на рисунке 2 [1].

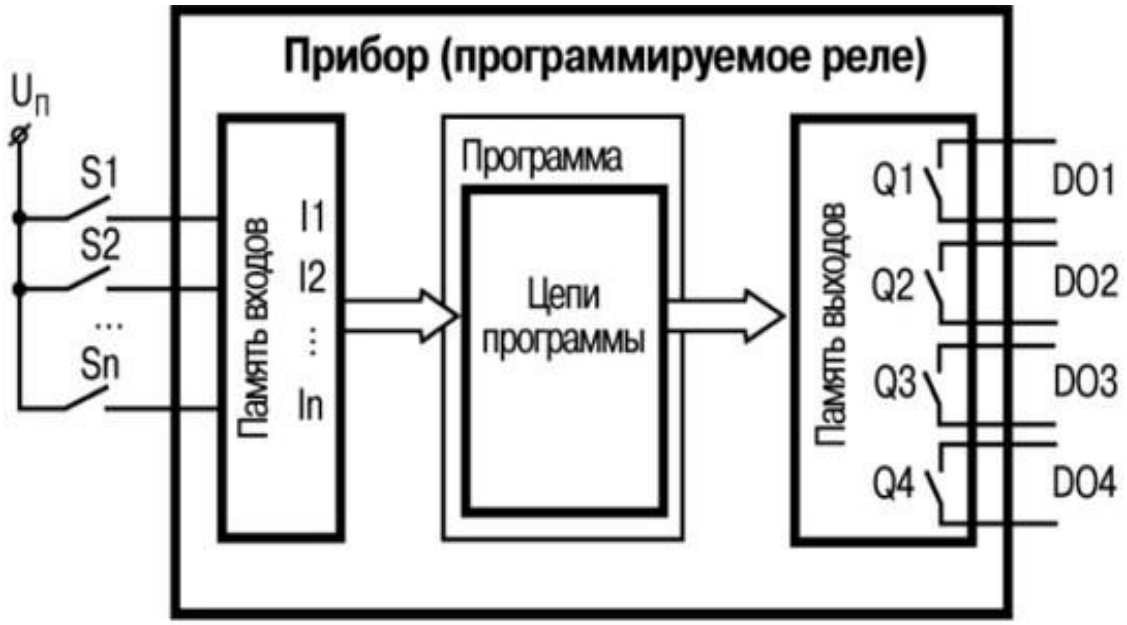


Рисунок 2 – Функциональная схема выполнения программы ПЛК

Необходимый сетевой обмен данными между контроллером и панелью оператора будет организован по физическому интерфейсу RS-485, объединив указанные элементы в сеть. В RS-485 для передачи и приёма данных используется витая пара. Передача данных осуществляется с помощью дифференциальных сигналов. Обмен данными между ними будет осуществляться через протокол Modbus RTU. В качестве ведущего устройства (master) при обмене будет выступать панель оператора, а ведомым (slave) устройством – контроллер.

Разберем обмен данными между панелью оператора и контроллером.

Интерфейс RS-485 предполагает наличие двух типов устройств: управляющих (master) и подчиненных (slave). Master-устройство координирует обмен данными в сети, а slave-устройства выполняют его команды, например, принимают и сохраняют данные или отправляют информацию по запросу.

RS-485 – это двухпроводной последовательный интерфейс. Каждое устройство, использующее этот интерфейс, имеет две клеммы, обозначенные как A и B. Для организации сети RS-485 необходимо соединить одноименные клеммы всех устройств между собой: A к A, B к B. Топология сети должна быть линейной, когда устройства последовательно подключаются к ведущему устройству (master). Разветвленные топологии, такие как "звезда" или "треугольник", не поддерживаются.

Для корректной работы сети RS-485 необходимо единообразно настроить все подключенные устройства. Ключевые параметры:

– протокол обмена (язык связи); все устройства должны использовать один и тот же протокол, в данном проекте – Modbus RTU; этот протокол определяет правила обмена данными между устройствами;

– скорость передачи данных (бодрейт); скорость передачи данных влияет на надежность связи; для коротких линий связи можно использовать более высокие скорости, а для длинных – более низкие. Важно, чтобы все устройства в сети были настроены на одну и ту же скорость;

– адрес устройства (Slave ID); каждому ведомому (slave) устройству в сети Modbus RTU присваивается уникальный адрес в диапазоне от 1 до 240; этот адрес служит для идентификации устройства в сети; недопустимо, чтобы два устройства имели одинаковый адрес; порядок присвоения адресов не имеет значения, главное – уникальность;

– дополнительные параметры; необходимо также обеспечить идентичность настроек контроля четности, количества стоп-бит и битов данных на всех устройствах сети [1].

Обмен данными с устройством slave требует от master-устройства знания карты памяти slave. Master должен знать адреса регистров,

содержащих интересующую информацию, и указывать операцию (чтение или запись) при обращении к конкретному регистру.

Разберем общие сведения о преобразователе частоты.

Электродвигатели являются ключевым элементом в работе многих промышленных установок, от производственных линий до станков. Управление электродвигателем включает в себя не только его включение и выключение, а также изменение направления вращения, но и точную регулировку скорости.

В современных электроприводах для управления скоростью вращения электродвигателей широко применяются преобразователи частоты (ПЧ). Эти устройства, помимо регулирования скорости, обеспечивают ограничение пусковых токов и защиту двигателя от аварийных ситуаций. Наиболее распространенной архитектурой являются ПЧ с промежуточным звеном постоянного тока, реализованные на базе инверторов напряжения. Структурная схема такого преобразователя представлена на рисунке 3[1].

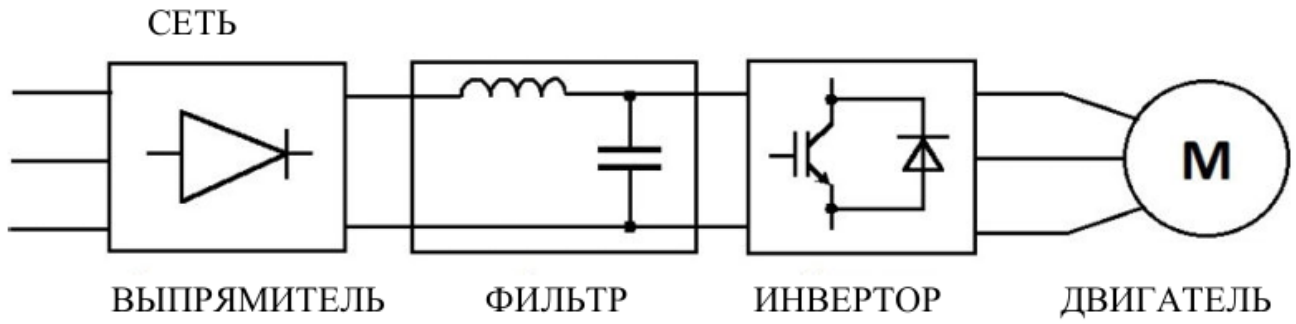


Рисунок 3 – Структурная схема преобразователя частоты
с промежуточным звеном постоянного тока

Переменное напряжение из сети сначала преобразуется в постоянное с помощью диодного выпрямителя и сглаживается фильтром, состоящим из индуктивности и емкости. Затем инвертор преобразует этот постоянный ток обратно в переменный, формируя на выходе напряжение и частоту, необходимые для работы нагрузки. В частности, для управления электродвигателями частотный преобразователь (ПЧ) генерирует

последовательность импульсов постоянного напряжения с регулируемой шириной, которые, проходя через индуктивную обмотку статора, создают ток, близкий к синусоидальному.

Работа частотного преобразователя (ПЧ) конфигурируется посредством параметризации. Набор параметров позволяет активировать или деактивировать различные функции, присваивая им соответствующие значения. Интерфейс пользователя, представленный кнопками на пульте управления, обеспечивает возможность изменения и задания значений параметров для оптимизации характеристик ПЧ. Цифровой дисплей отображает идентификаторы выбранных параметров и их текущие значения, обеспечивая визуальный контроль [1].

Исходя из требуемого функционала, планируемой архитектуры проекта и выбранных устройств был определен набор инструментов и программ, состоящий из: язык функциональных блоков FBD стандарта IEC 61131 – 3, среда разработки OWEN Logic, конфигуратор панели оператора СП200, ОС Microsoft Windows 10, Microsoft Word 2025.

Создание графического интерфейса проекта будет происходить в программе «Конфигуратор СП200». Создание управляющего алгоритма ПЛК ОВЕН ПР200 будет производится в среде разработки OWEN Logic.

1.1.4 Выбор языка программирования

На текущий момент, среди огромного множества языков программирования, предположительно подходящих для разработки текущего проекта, был выбран язык FDB.

Function Block Diagram (FBD) – это графический язык программирования высокого уровня, соответствующий стандарту IEC 61131-3, широко используемый для разработки управляющих программ ПЛК в промышленной автоматизации. Он обеспечивает декларативное описание логики управления посредством соединения функциональных блоков, что позволяет эффективно управлять потоками данных различных типов и инкапсулировать сложные алгоритмы. FBD использует символику булевой алгебры для представления

позволяет эффективно управлять потоками данных различных типов и инкапсулировать сложные алгоритмы. FBD использует символику булевой алгебры для представления логических операций и, благодаря своей графической природе, обеспечивает более эффективное отображение структурной информации по сравнению с языком релейно-контактных схем [4].

Согласно IEC 61131-3, основными элементами языка FBD являются: переменные, функции, функциональные блоки и соединения. Переменные бывают входные, выходные и входные/выходные, пример представлен на рисунке 4.

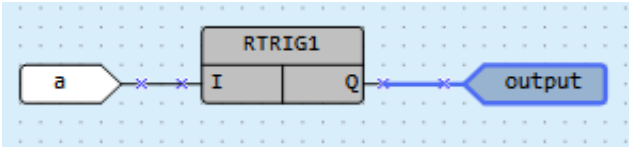


Рисунок 4 – Пример переменных

Данные передаются между функциями и функциональными блоками посредством переменных, которые соединяются с их входными и выходными параметрами. Входные данные поступают в функцию/блок через входные переменные, подключенные к входным параметрам. Выходные данные передаются из функции/блока через выходные переменные, подключенные к выходным параметрам. Возможно также использование одной и той же переменной как для ввода, так и для вывода. Наконец, выходные данные одной функции/блока могут быть напрямую направлены на вход другой, формируя цепочку обработки. Пример возможного алгоритма соединения функциональных блоков представлен на рисунке 5.

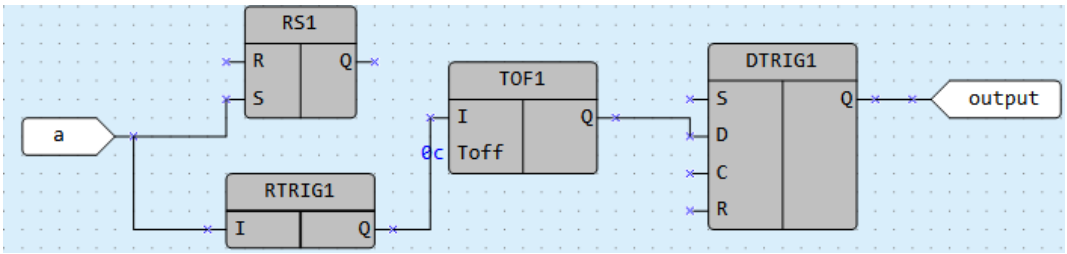


Рисунок 5 – Пример соединения блоков

Блок – это подпрограмма, обрабатывающая какую-либо логику и функционал действия (И, ИЛИ, НЕ, триггеры, таймеры, счётчики и др.) [2].

Пример фрагмента программы: по нажатию кнопка зажигает светодиод на 3 секунды и потом гаснет. Разработана на языке FBD в среде OWEN Logic, представлена на рисунке 6.

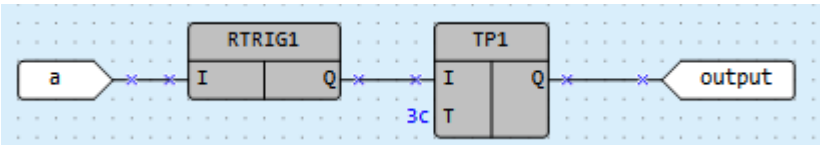


Рисунок 6 – Функциональная программа для ПЛК

1.1.5. Выбор среды разработки

В качестве среды программирования, как было описано выше, была выбрана среда разработки OWEN Logic. Для создания графического интерфейса используется программа Конфигуратор СП200.

Owen Logic – среда программирования, предназначенная для создания алгоритмов работы коммутационных приборов, относящихся к классу программируемых реле, в частности, приборов серий ПР1xx, ПР200 и панели ИПП120 производства компании ОВЕН.

Среда разработки Owen Logic позволяет пользователю разработать программу автоматизации системы по собственному алгоритму и записать ее в энергонезависимую память прибора. Для составления программы используется графический язык FBD стандарта МЭК 61131-3 [2].

В перечень приборов, для программирования которых может использоваться Owen Logic, представленный на сайте компании ОВЕН, входит используемый в проекте ПЛК ОВЕН СП200.

Данная среда программирования бесплатна и доступна для общего пользования на сайте разработчика, демонстрация на рисунке 7.

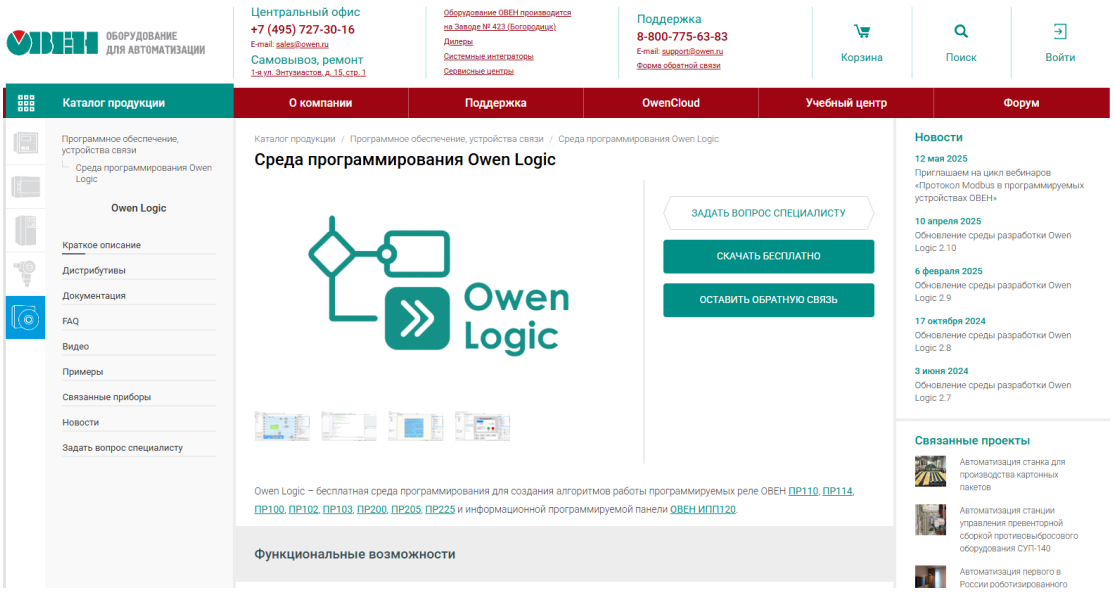


Рисунок 7 – Страница загрузки среды программирования

Также, помимо ссылки на загрузку среды разработки, на сайте OWEN можно найти описание функционала программы, системных требований, техническую документацию и т.д. Нажав кнопку «Скачать бесплатно», сайт запросит принять условия пользовательского соглашения, которое следует внимательно прочитать перед подтверждением, демонстрация на рисунках 8 и 9.

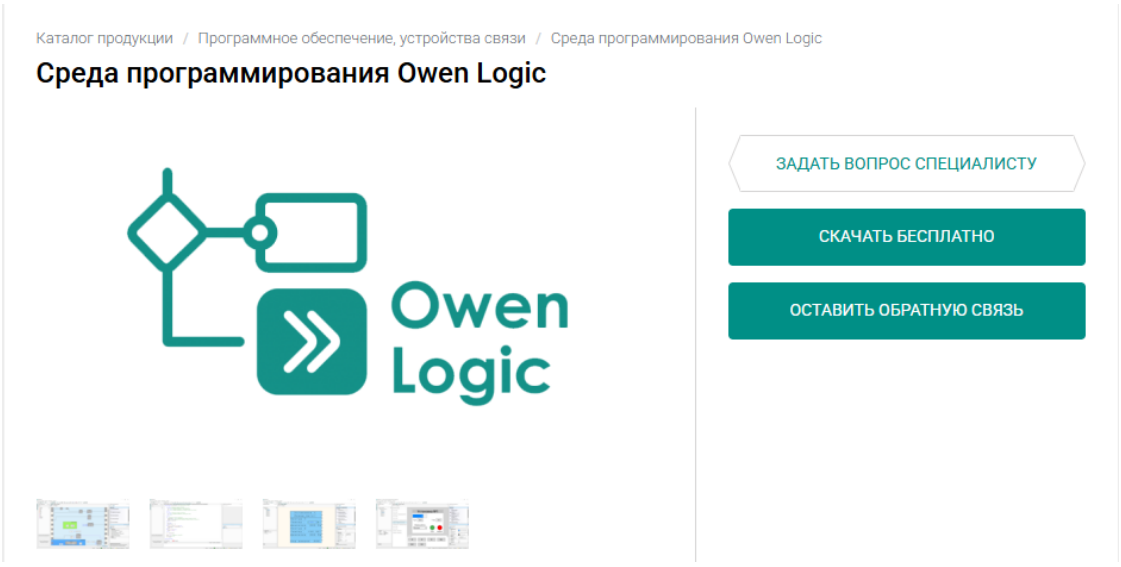


Рисунок 8 – Область сайта с загрузкой программы

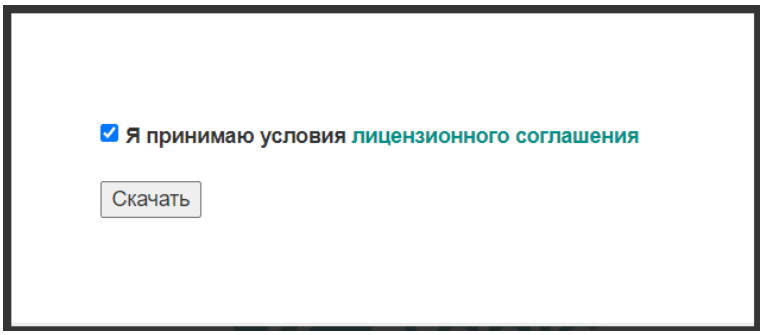


Рисунок 9 – Окно с условиями лицензионного соглашения

При нажатии на кнопку "Скачать" инициируется загрузка архивного файла, содержащего исполняемый файл для установки OWEN Logic. Визуальное представление процесса загрузки представлено на рисунке 10.

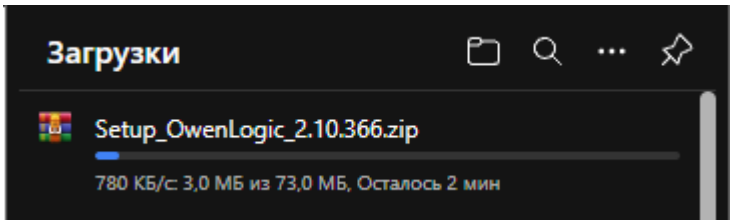


Рисунок 10 – Загрузка архива

Окончив загрузку, архив следует распаковать в отдельную папку. Запустив файл Setup_OwenLogic_2.10.366.exe от имени администратора, нас встречает окно установки, изображенное на рисунке 11.

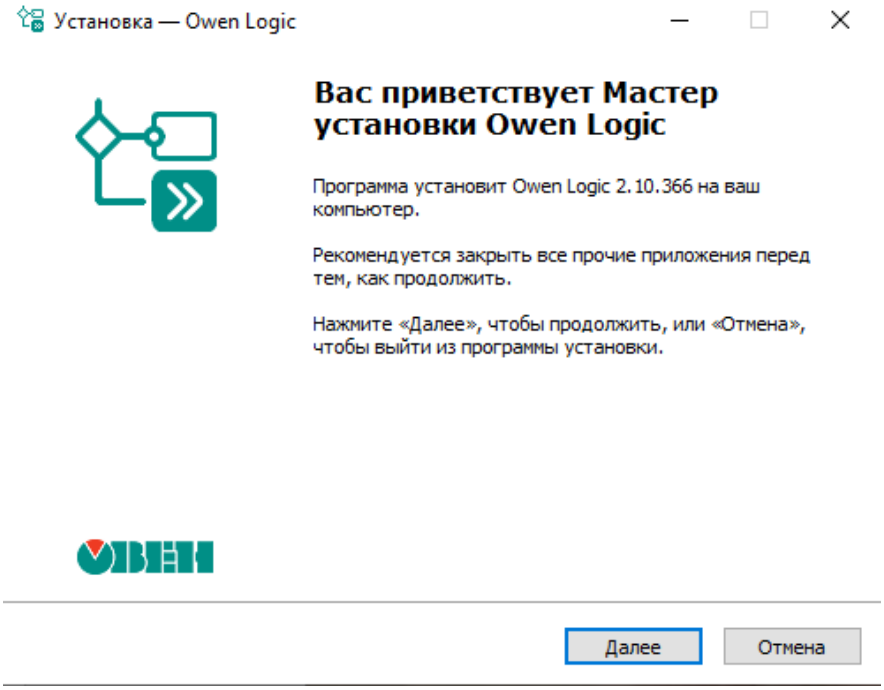


Рисунок 11 – Окно установки OWEN Logic

Затем следует окно с лицензионным соглашением на рисунке 12, которое следует прочитать и принять.

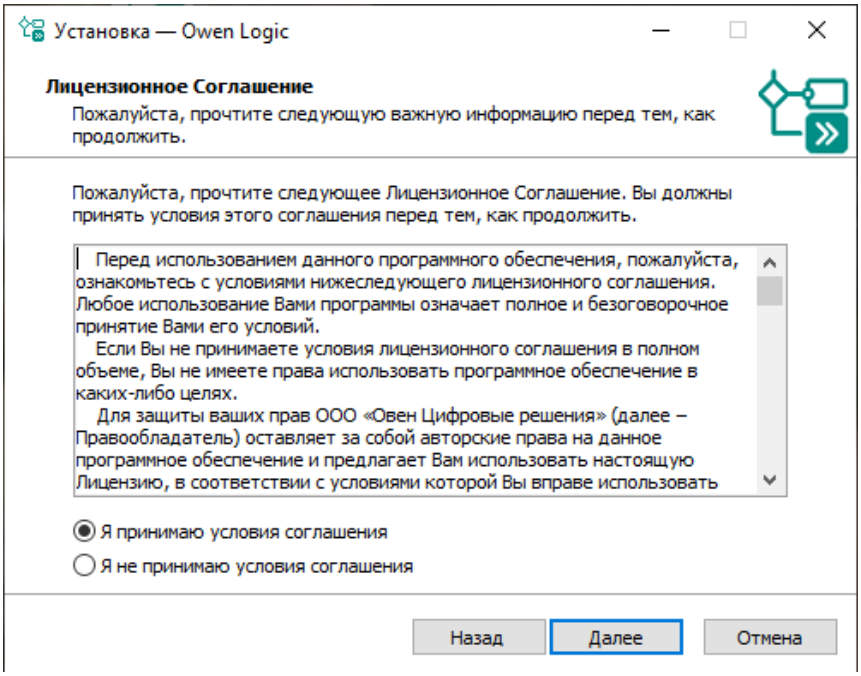


Рисунок 12 – Окно с лицензионным соглашением

Далее следует определить место установки программы с учетом ее размера в 204,9 МБ. Демонстрация на рисунке 13.

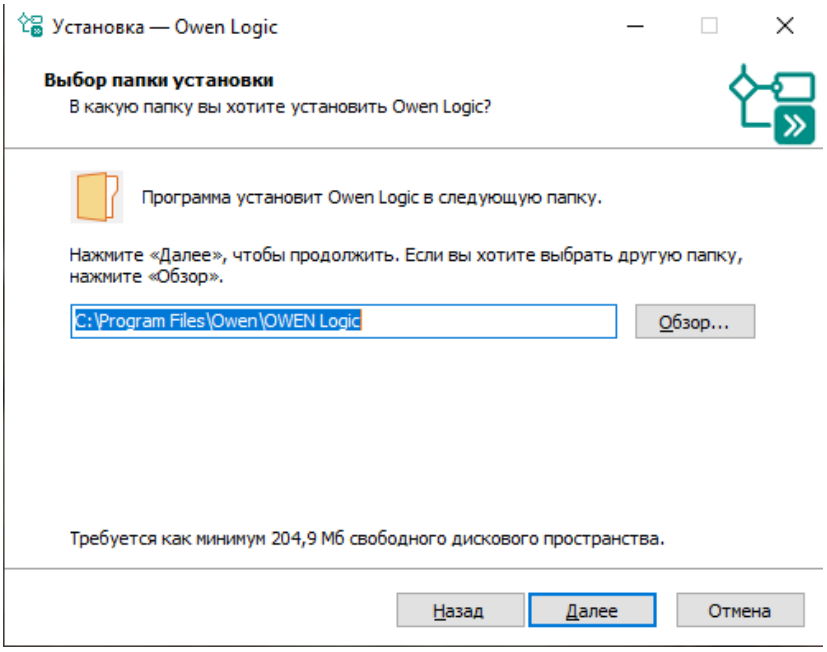


Рисунок 13 – Окно с выбором папки установки

Выбираем папку в меню «Пуск». Демонстрация на рисунке 14.

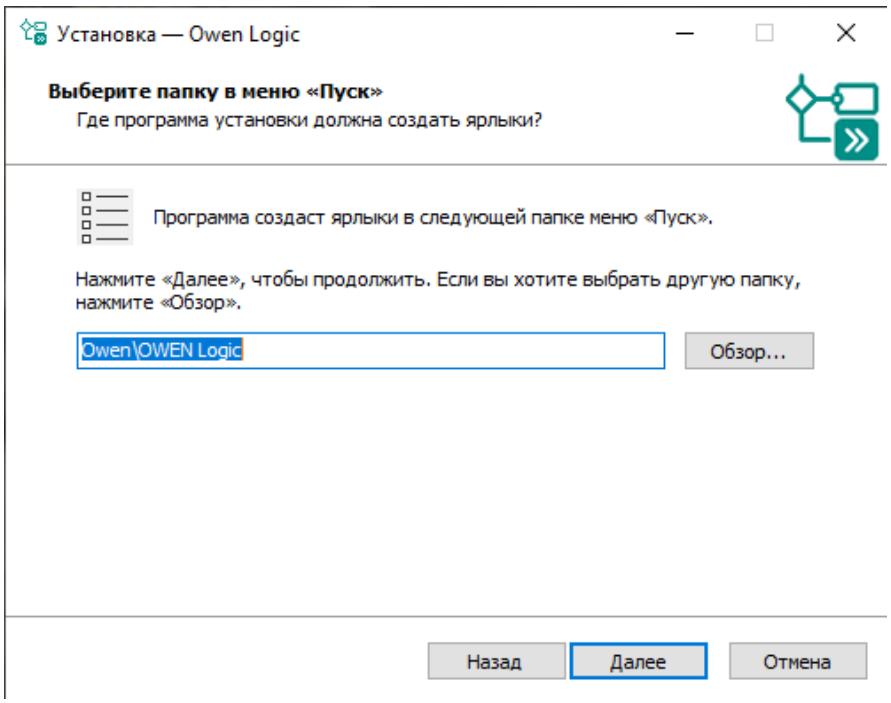


Рисунок 14 – Окно с выбором папки в меню «Пуск»

Затем окно установки программы предлагает выбрать дополнительные задачи, параметры были выбраны на рисунке 15.

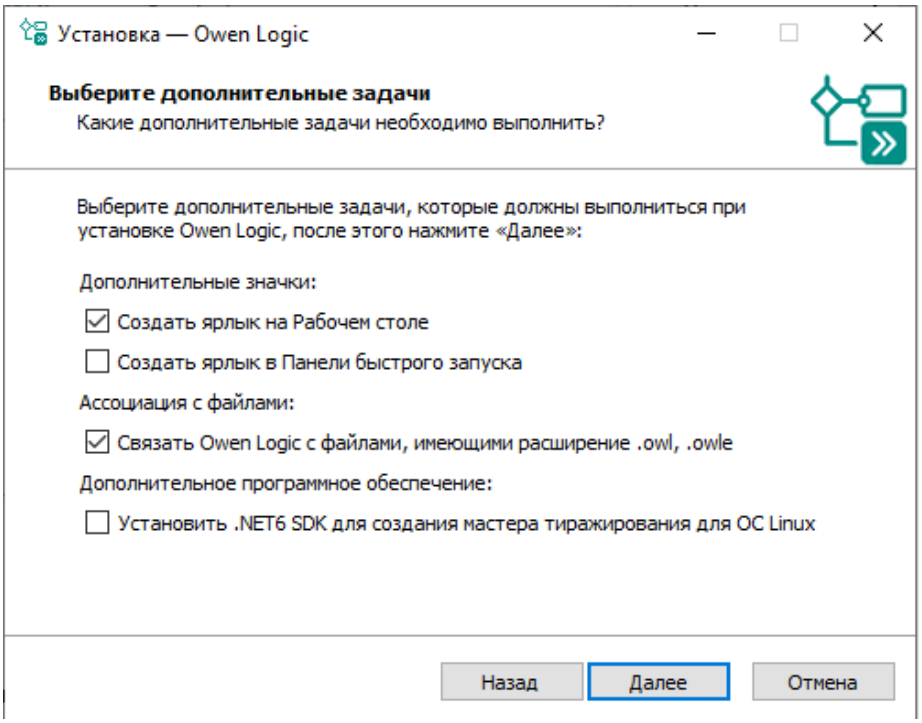


Рисунок 15 – Окно выбором дополнительных задач

После выбора всех параметров установки программы встречается последнее окно, продемонстрированное на рисунке 16.

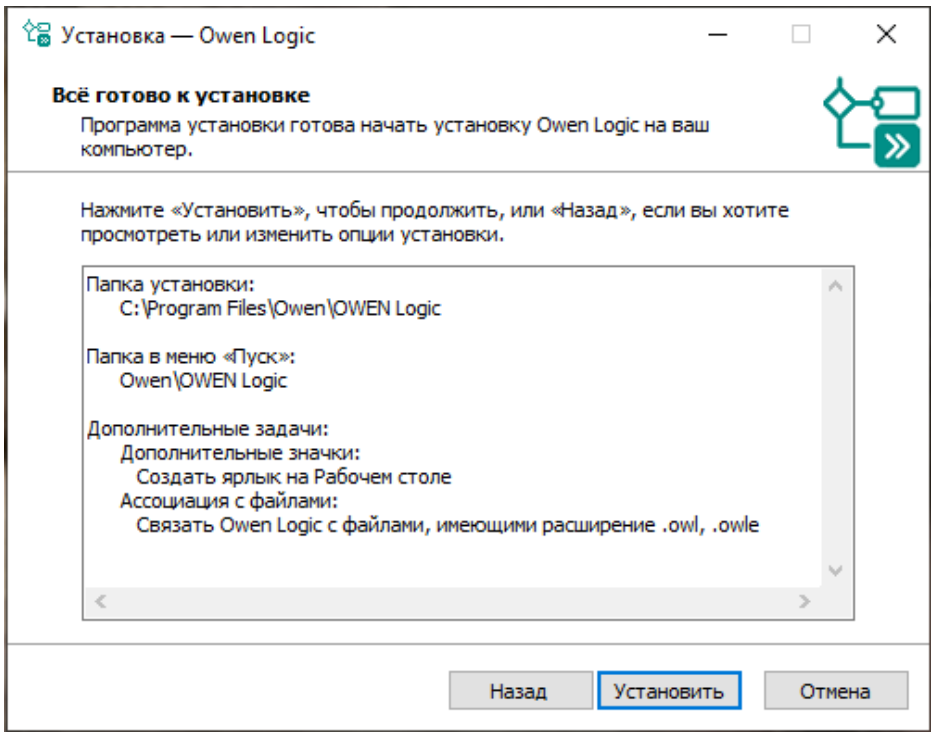


Рисунок 16 – Окно с готовностью к установке

Успешно установив программу OWEN Logic, можно переходить к описанию ее интерфейса.

При первом запуске пользователя встречает пустое поле, изображенное на рисунке 17. Для начала работы нужно создать новый проект через вкладку «Файл» в левом верхнем углу и выбрать пункт «Новый проект», как показано на рисунке 18.

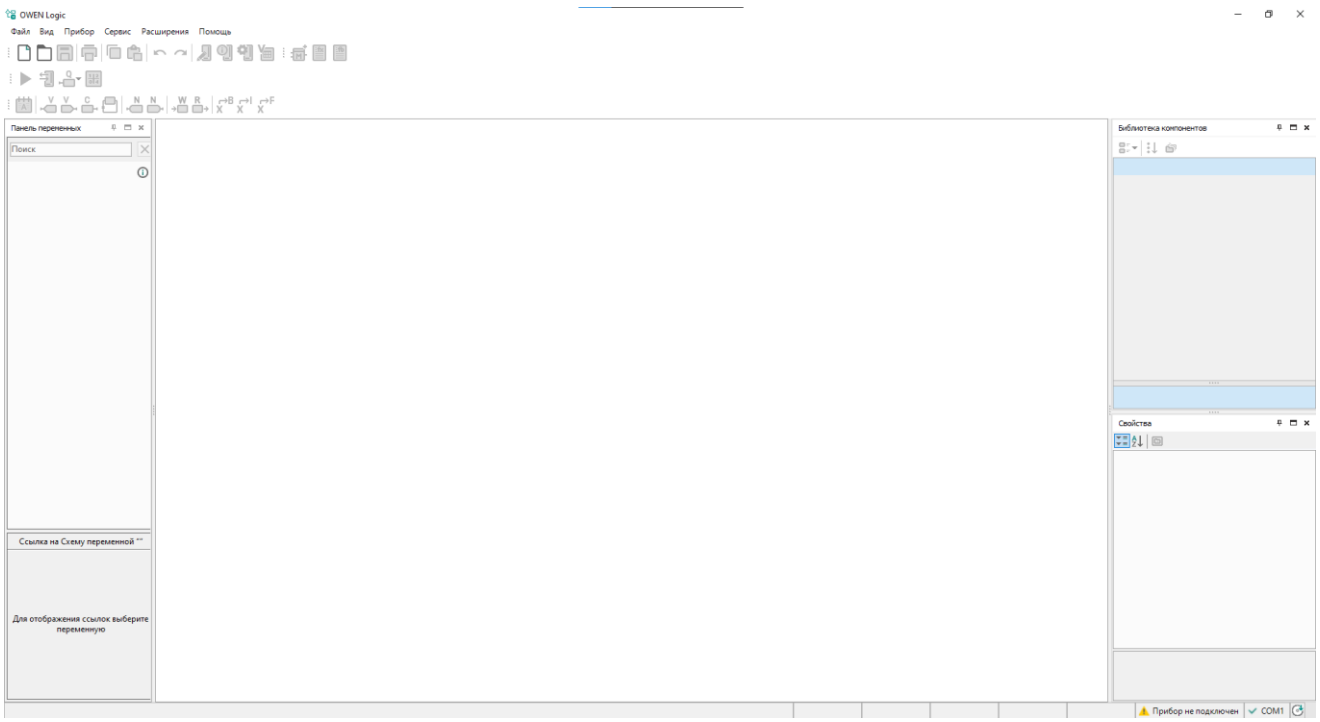


Рисунок 17 – Первое включение программы Owen Logic

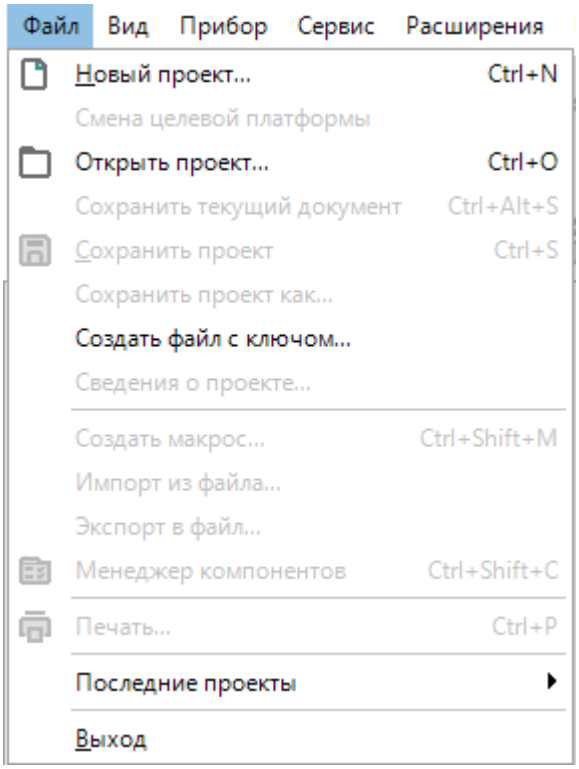


Рисунок 18 – Вкладка «Файл» на панели

Создав проект, появляется рабочая область, как показано на рисунке 19.

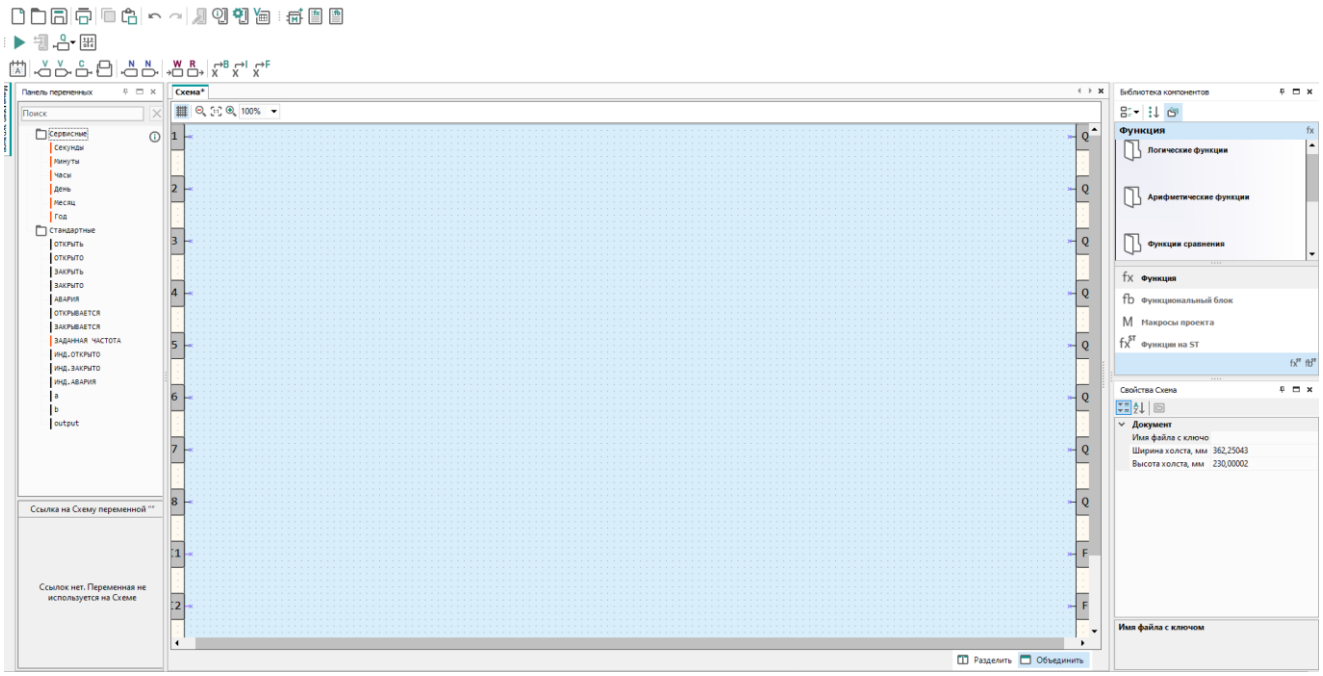


Рисунок 19 – Рабочая область

Справа от рабочей области имеется вкладка «Библиотека компонентов», в которой хранится набор разнообразных функциональных блоков языка FBD, демонстрация на рисунке 20.

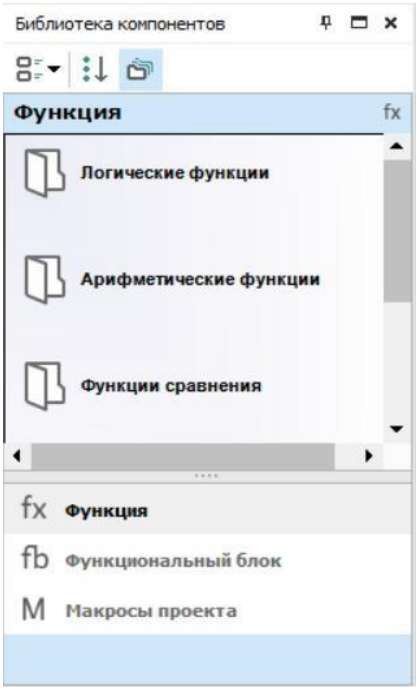


Рисунок 20 – «Библиотека компонентов»

Вверху имеется панель различных инструментов, реализующих разнообразный функционал, показано на рисунке 21.

Рисунок 21 – Панель инструментов

Слева от рабочей области находится «Панель переменных». С ее помощью можно добавлять различные переменные в свой проект, как показано на рисунке 22.

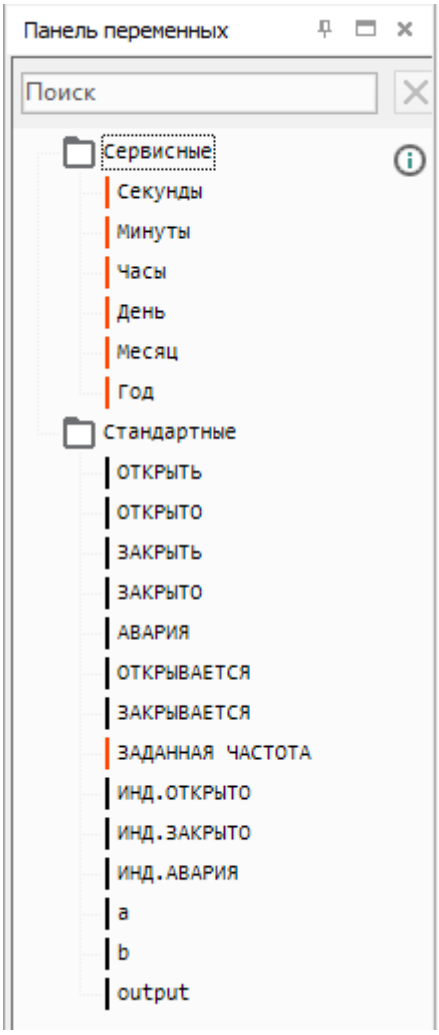


Рисунок 22 – «Панель переменных»

Также в среде разработки OWEN Logic имеется справочник, содержащий основную информацию об эксплуатации программы, демонстрация на рисунке 23.

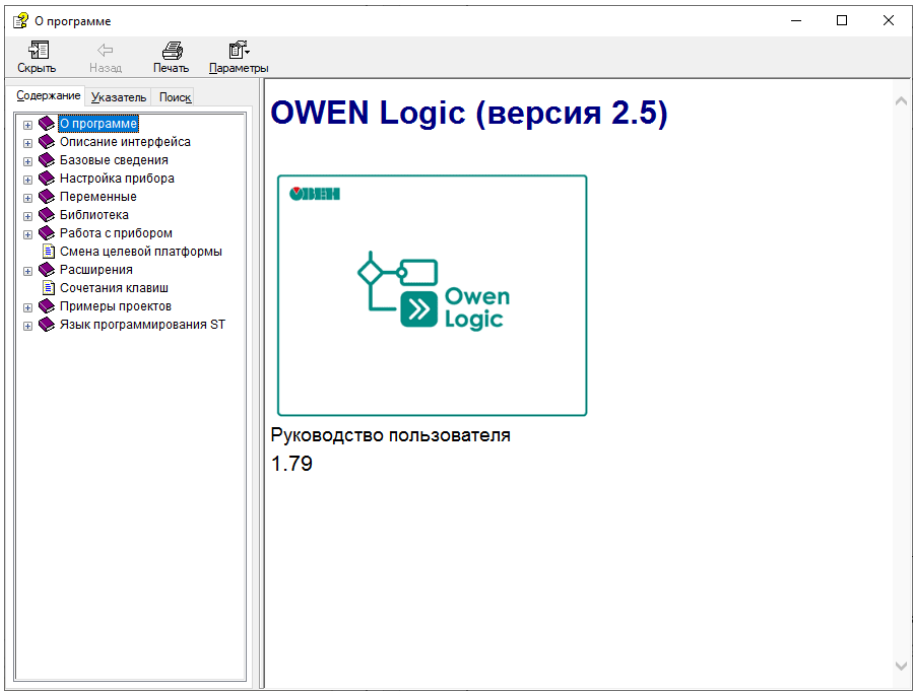


Рисунок 23 – Справочник OWEN Logic

Закончив с установкой и настройкой среды OWEN Logic, можно приступать к части графического интерфейса для панели оператора. Для создания графического интерфейса была установлена программа Конфигуратор СП200.

Программа «Конфигуратор» – это прикладное программное обеспечение, предназначенное для программирования (создания конфигурации) панели оператора.

Логика работы панели СП270 определяется потребителем в процессе конфигурирования на ПК с использованием программного обеспечения «Конфигуратор СП200».

Конфигуратор позволяет формировать и сохранять пользовательские экраны, отображаемые на экране панели в процессе эксплуатации. Совокупность экранов образует проект, который можно загрузить в панель или

сохранить в виде файла на жестком диске компьютера. После загрузки проекта панель сразу начнет опрашивать регистры ПЛК или других приборов и отображать их значения на дисплее [3].

Процесс установки программы Конфигуратор СП200 показан на рисунках ниже. На рисунке 24 показан мастер установки Конфигуратора СП200.

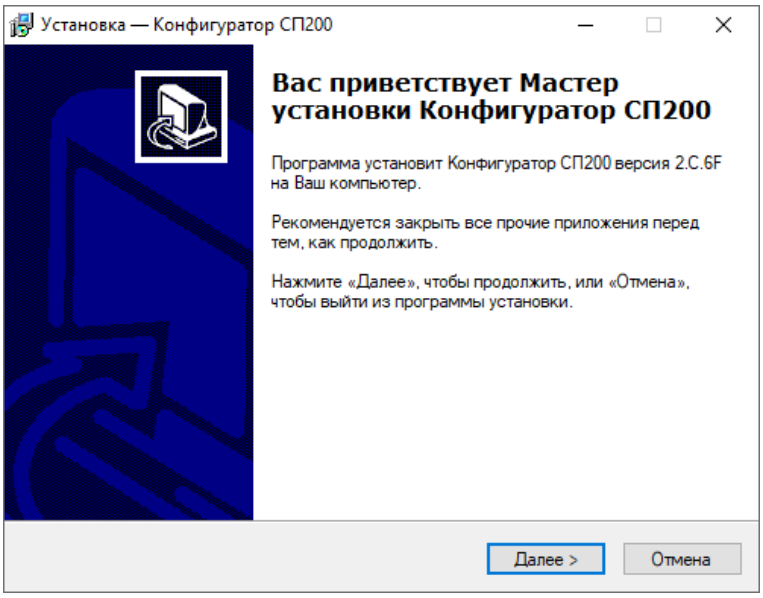


Рисунок 24 – Мастер установки Конфигуратора СП200

На рисунке демонстрируется 25 идет следующий этап, на котором требуется выбрать расположение программы установки Конфигуратора СП200 на дисковом пространстве ПК. Для того, чтобы не выбирать путь по умолчанию, можно нажать на кнопку «Обзор», в открывшемся окне указать нужное расположение. При выборе дискового пространства стоит обращать внимание на минимальный размер программы.

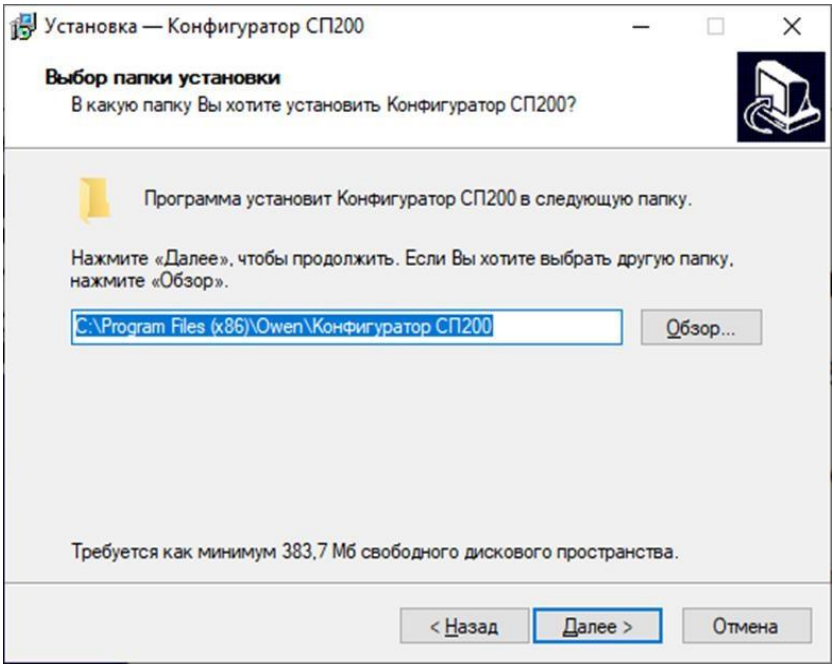


Рисунок 25 – Окно с выбором папки установки

В следующем окне, если требуется, можно создать значок на рабочем столе. Для этого достаточно поставить флажок рядом с соответствующим пунктом, как показано на рисунке 26.

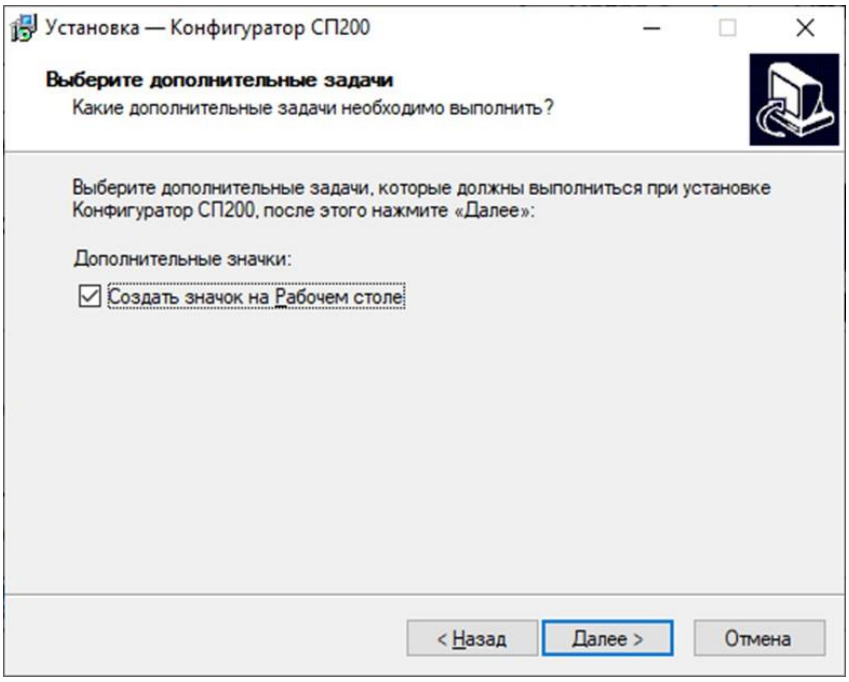


Рисунок 26 – Дополнительные настройки установки Конфигуратора СП200

После выбора всех параметров установки программы встречается последнее окно, продемонстрированное на рисунке 27.

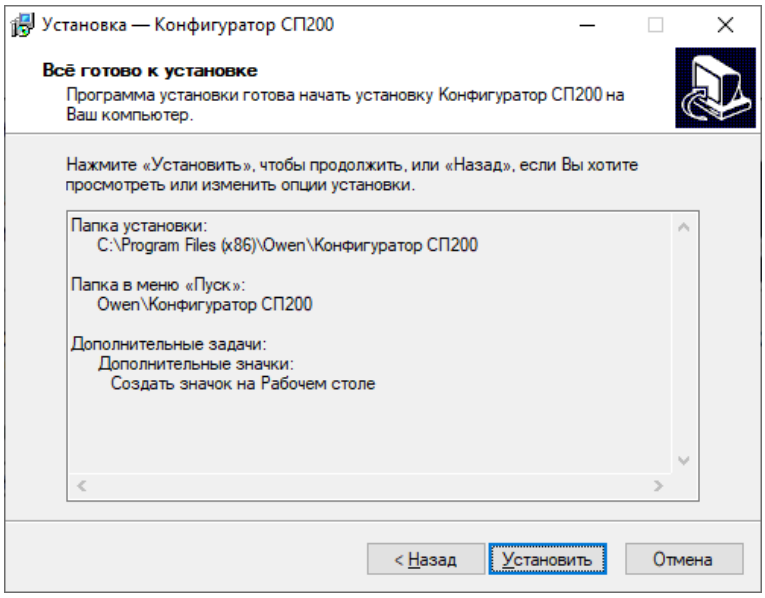


Рисунок 27 – Окно с готовностью к установке

На рисунке 28 изображено окно, которое появляется после установки Конфигуратора СП200. Если вы хотите сразу же запустить программу после установки, убедитесь, что галочка напротив соответствующей опции установлена.

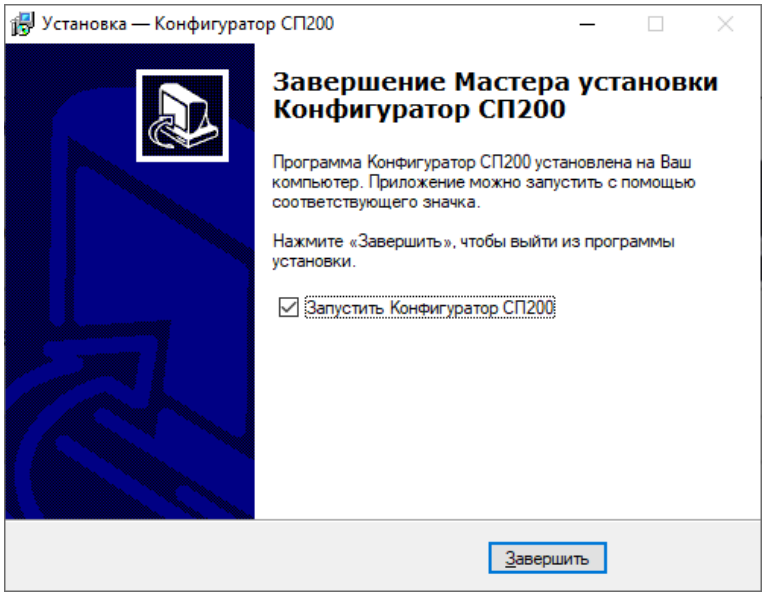


Рисунок 28 – Окно с завершением установки

На рисунке 29 показано активное окно, открывающееся после запуска и установки Конфигуратора СП200. Для взаимодействия с данным ПО необходимо создать новый проект.

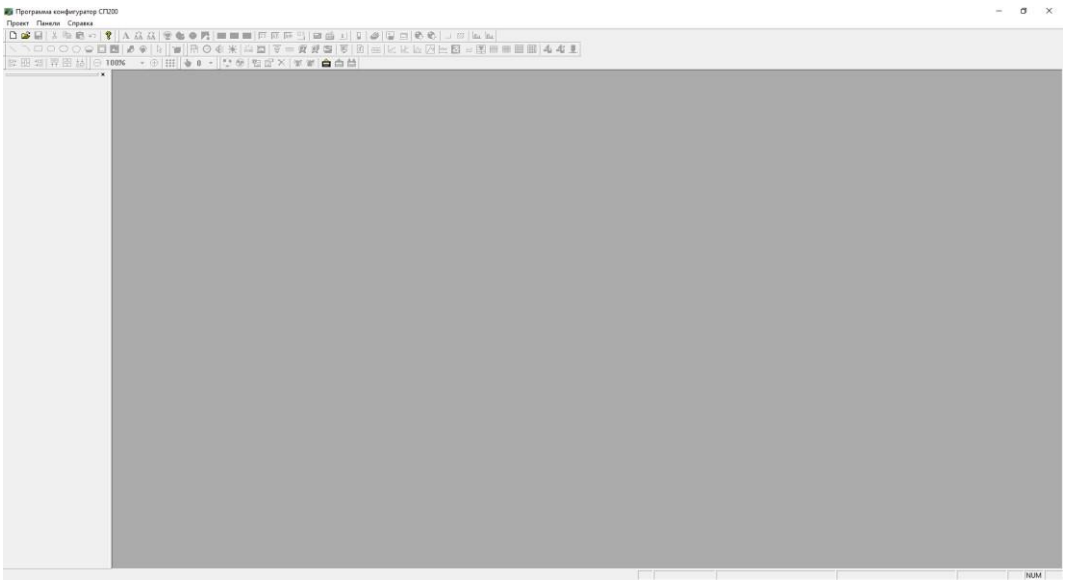


Рисунок 29 – Начальное окно программы Конфигуратор СП200

На рисунке 30 представлена панель инструментов Конфигуратора СП200, находящая в самом верху программы. Подобно среде OWEN Logic, эта панель предоставляет широкие возможности настройки, делая ее более удобной в использовании.

Рисунок 30 – Панель инструментов Конфигуратора СП200

После создания проекта, в открывшемся окне справа имеется древо элементов для удобства навигации по проекту. Вид древа для нового проекта изображен на рисунке 31.

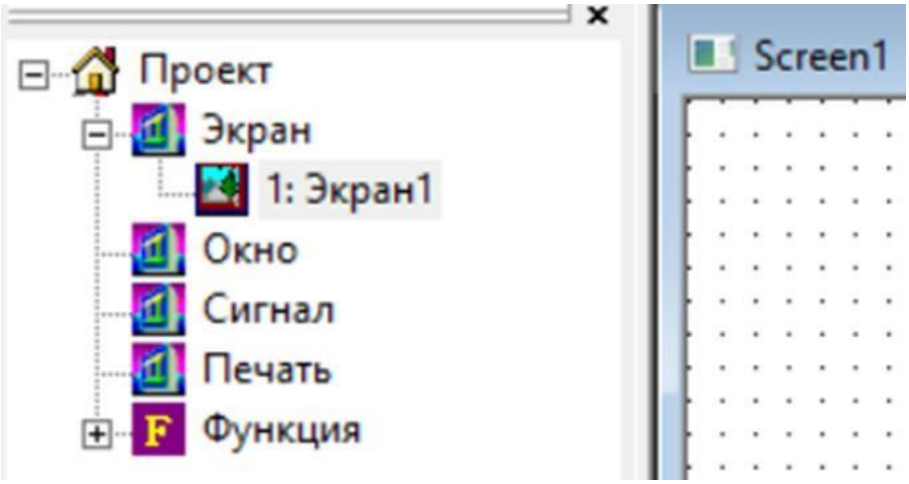


Рисунок 31 – Элементы проекта

1.2 Проектирование и разработка

Проект состоит из управляющей программы, загруженной в логический контроллер и графического интерфейса, загруженного в панель оператора. Подключив к сети панель оператора и ПЛК, пользователь может начинать работу по управлению движением привода продольного перемещения суппорта токарного станка.

Взаимодействуя с графическим интерфейсом, пользователь проекта может контролировать как скорость крутящего момента двигателя привода, так и время задержки на концевых выключателях. Также имеется возможность по плановой или аварийной остановке движения суппорта.

Перемещение суппорта на заданной рабочей скорости начинается по кратковременному нажатию кнопки «ПУСК/СТОП», остановка – по срабатыванию концевого выключателя. Затем привод стоит в течение заданного промежутка времени и быстро возвращает инструмент (реверс на заданной скорости холостого хода), остановка – по срабатыванию второго концевого выключателя (начального). Прервать цикл работы возможно в любой момент времени кратковременным нажатием кнопки «ПУСК/СТОП».

1.2.1 Конфигурирование ПЛК ОВЕН ПР200

ПЛК ОВЕН ПР200 — это программируемый логический контроллер (ПЛК), разработанный российской компанией ОВЕН. Он предназначен для

автоматизации технологических процессов в промышленности, ЖКХ, энергетике и других сферах. Имеет 32 – битный процессор с поддержкой реального времени, оперативную память в размере 64 кб, флэш-память (ПЗУ) 512 КБ (для программы пользователя). Контроллер ПР200 обладает гибкой системой ввода/вывода, которая включает: дискретные входы – до 24 (в зависимости от конфигурации), дискретные выходы – до 16 (реле, транзисторные), аналоговые входы – до 8 (0-10 В, 0-20 мА, термопары, RTD), аналоговые выходы – до 4 (0-10 В, 4-20 мА). Поддерживает языки стандарта IEC 61131-3, в числе которых нужный нам язык FBD (Function Block Diagram) и программируется в среде Owen Logic. ПЛК ПР200 поддерживает несколько протоколов связи, среди которых нужный нам Modbus RTU (RS-485) – для связи с датчиками, частотниками, другими ПЛК. На рисунке 32 представлен внешний вид прибора.

Рисунок 32 – ПЛК ОВЕН ПР200

Чтобы запрограммировать ПЛК ОВЕН ПР200 нужно запустить среду OWEN Logic и нажать кнопку «Новый проект» на панели инструментов. После появления окна «Выбор модели» выбирается прибор ПР200-24.2(4).Х (входы – 12, выходы – 12) соответствующий требованиям проекта. На рисунке 33 представлено окно «Выбор модели».

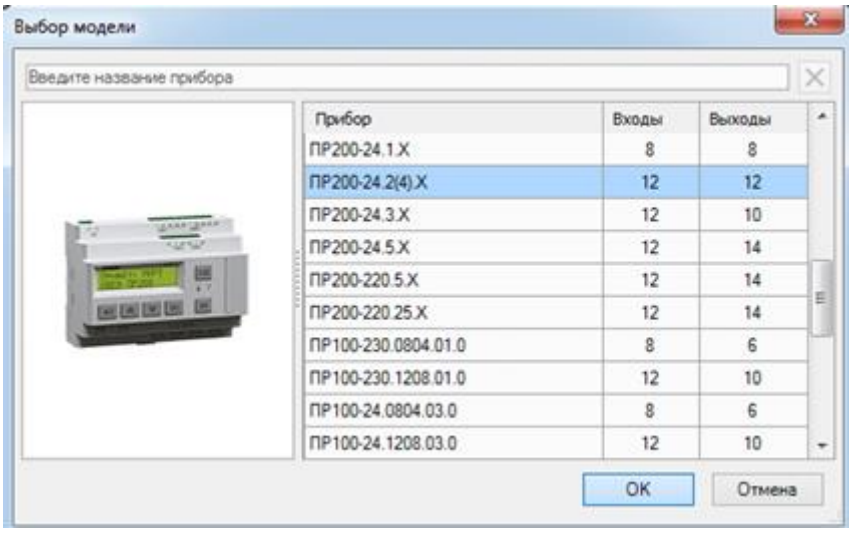


Рисунок 33 – Окно выбора модели при создании нового проекта

После выделения курсором нужной модели устройства выбор подтверждается нажатием кнопки «ОК». Затем появится окно с рабочей областью (холстом) среды программирования, изображенное на рисунке 34.

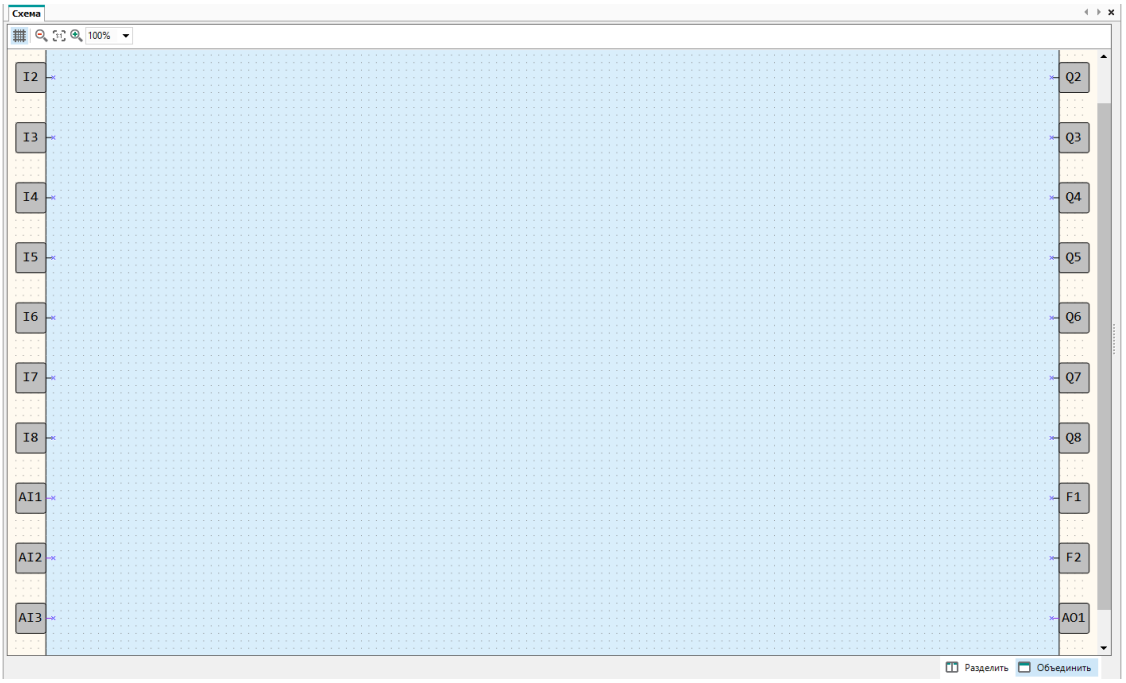


Рисунок 34 – Холст для программирования в среде OWEN Logic

По центру рабочей области расположен холст (поле), на котором будет создаваться алгоритм управляющей программы.

Выбрав устройство проекта, нужно перейти по вкладке «Прибор» и открыть окно «Настройки прибора». В настройках прибора переходим на

вкладку «Интерфейсы» и добавляем два интерфейса RS485 master и slave. В качестве ведущего (master) устройства при обмене должна выступать панель оператора, а ведомым (slave) устройством – контроллер. Выбираем необходимые параметры конфигурации, как показано на рисунках 35 и 36.

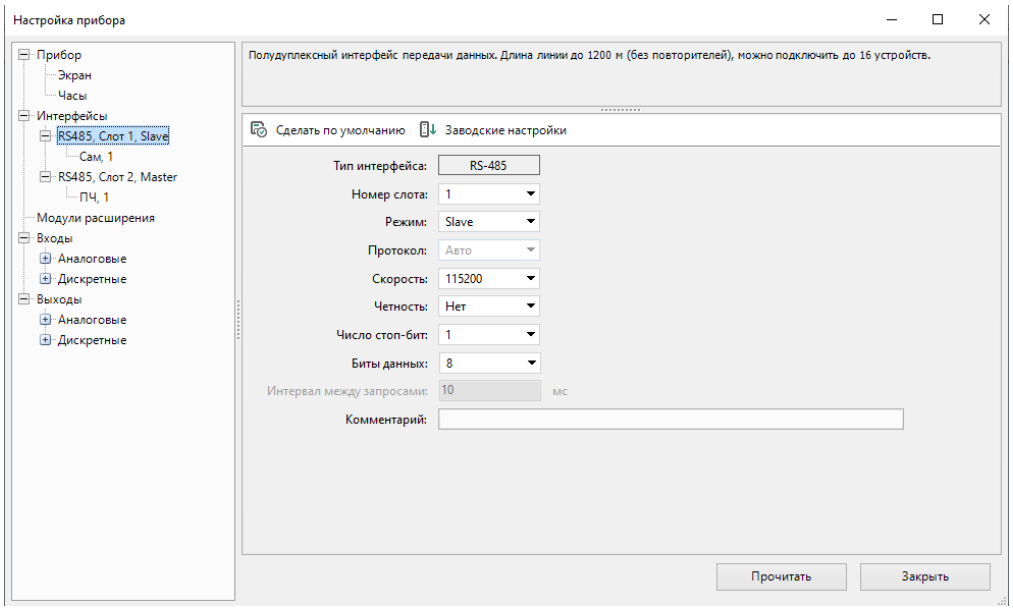


Рисунок 35 – Конфигурация Slave-устройства

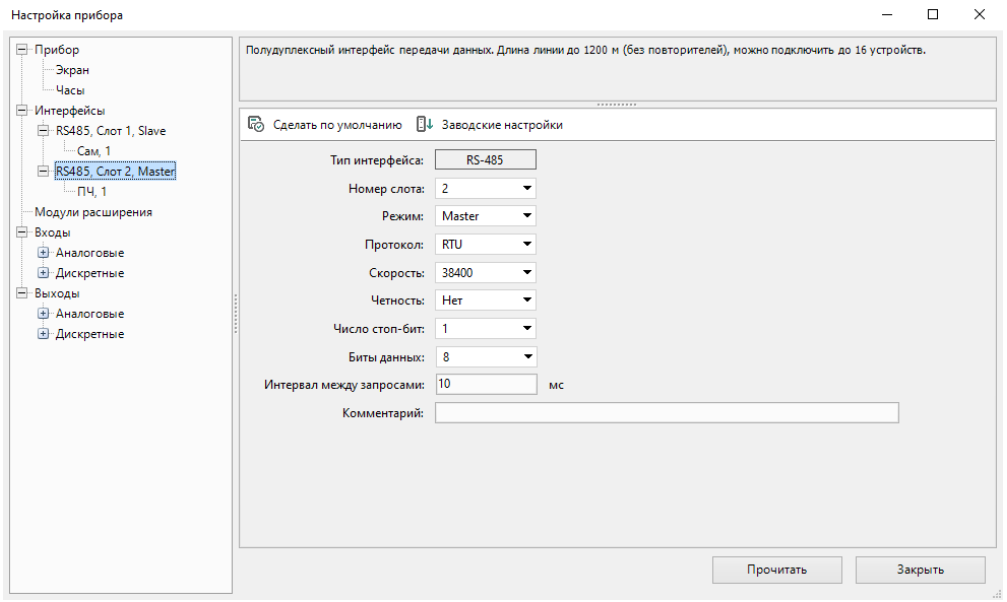


Рисунок 36 – Конфигурация Master-устройства

После выбора конфигурации задаем сетевые переменные, которые будем использовать в процессе разработки проекта, как показано на рисунках 37 и 38.

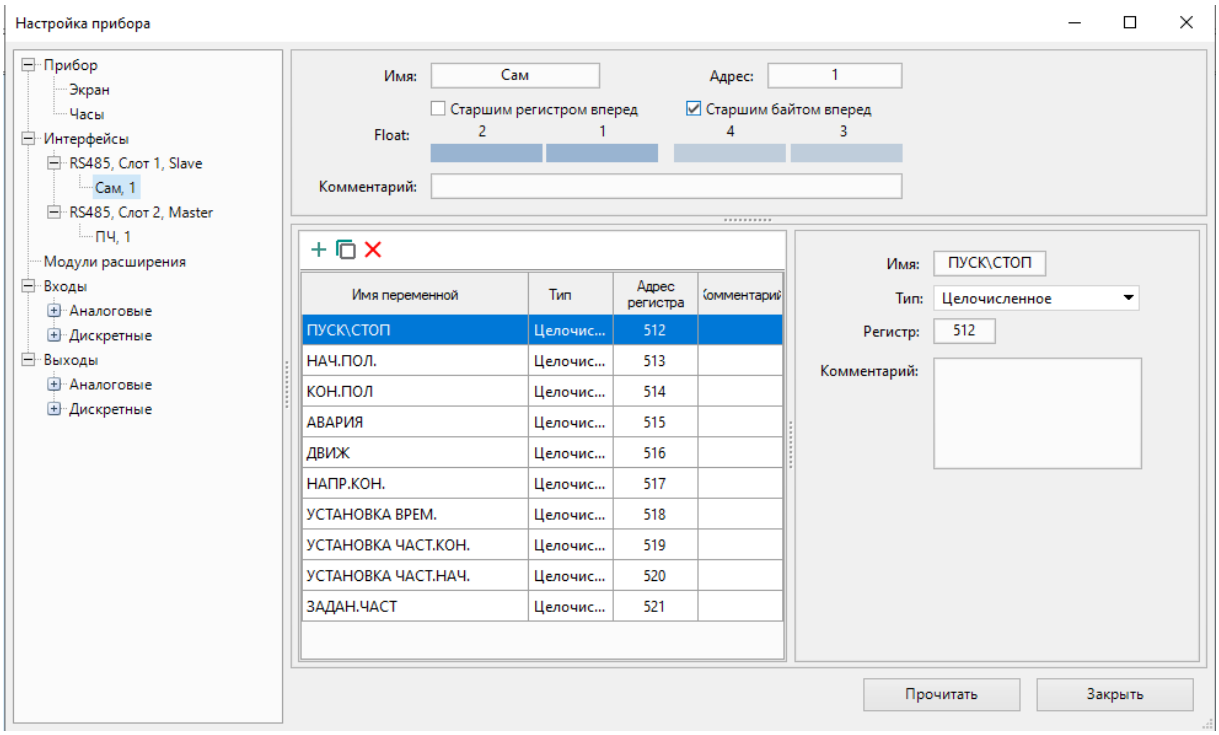


Рисунок 37 – Сетевые переменные для slave – устройства

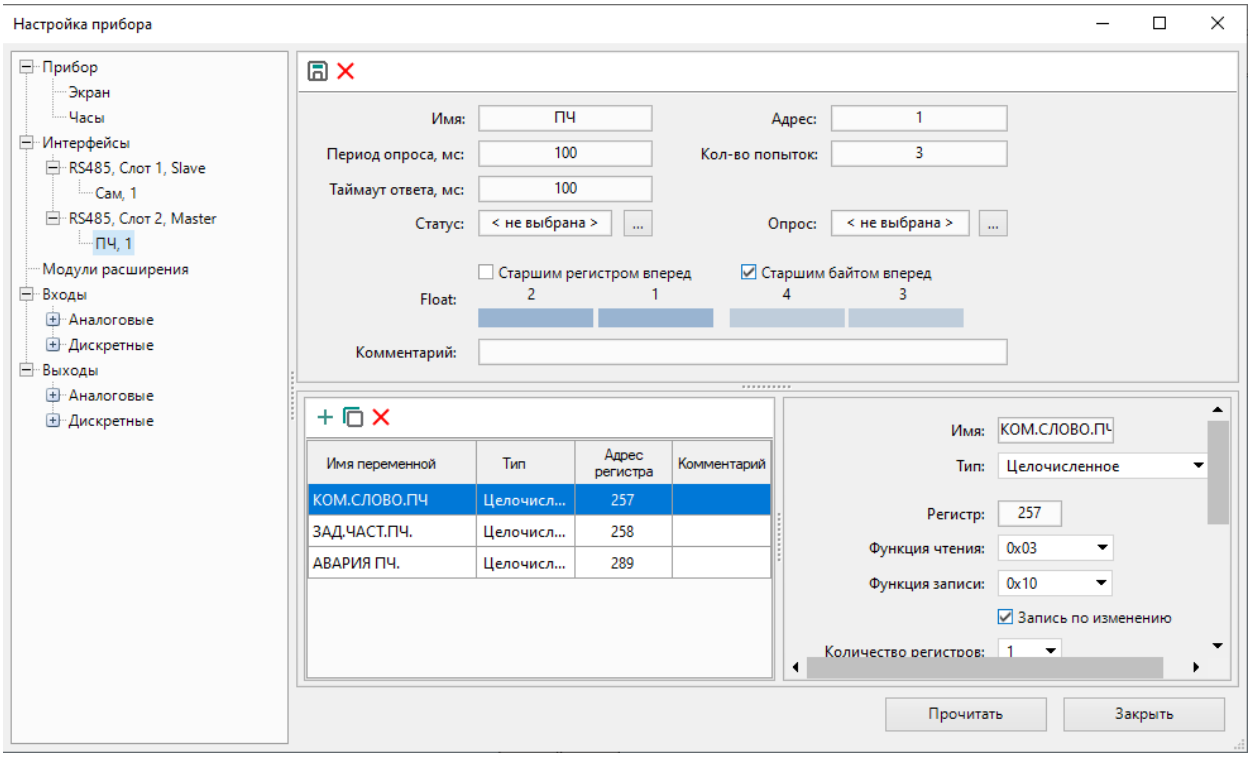


Рисунок 38 – Сетевые переменные для master – устройства

Таблицы 2 и 3 демонстрируют сетевые переменные проекта с адресами регистров, которые указаны в настройках прибора для slave и master устройств. Переменные master – устройства (панели оператора) нужны для связи с преобразователем частоты. Переменные slave – устройства нужны для построения логики работы контроллера.

Таблица 2 – Сетевые переменные проекта для slave – устройства

Имя переменной	Тип	Адрес регистра
ПУСК\СТОП	Целочисленное	512
НАЧ.ПОЛ.	Целочисленное	513
КОН.ПОЛ.	Целочисленное	514
АВАРИЯ	Целочисленное	515
ДВИЖ	Целочисленное	516
НАПР.КОН.	Целочисленное	517
УСТАНОВКА.ВРЕМ	Целочисленное	518
УСТАНОВКА.ЧАСТ.КОН.	Целочисленное	519
УСТАНОВКА.ЧАСТ.НАЧ.	Целочисленное	520
ЗАДАН.ЧАСТ.	Целочисленное	521

Таблица 3 – Сетевые переменные проекта для master – устройства

Имя переменной	Тип	Адрес регистра
КОМ.СЛОВО.ПЧ.	Целочисленное	257
ЗАД.ЧАСТ.ПЧ.	Целочисленное	258
АВАРИЯ ПЧ.	Целочисленное	289

Конфигурирование сетевых переменных, а не локальных, необходимо для организации связи и взаимодействия между панелью оператора, как устройство master и ПЛК, как устройство slave, по протоколу Modbus RTU через интерфейс RS-485. Демонстрация работы связки на рисунке 39

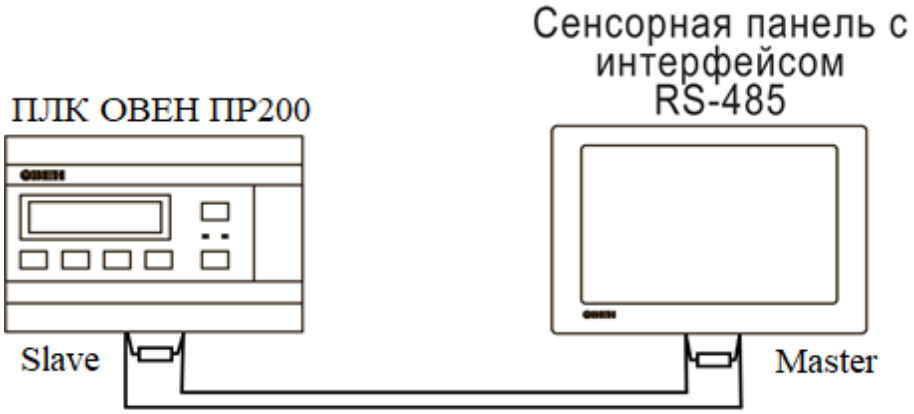


Рисунок 39 – Схема сетевого взаимодействия между мастер-панелью и ПЛК

Адреса сетевых переменных в среде OWEN Logic выбирались с расчетом на организацию впоследствии взаимосвязи между ними и кнопками на панели оператора в программе Конфигуратор ПР200. При нажатии кнопок на панели оператора будет происходить вызов определенной переменной по адресу регистра, привязанному к кнопке и самой переменной.

1.2.2 Разработка управляющей программы в среде OWEN Logic

Завершив работу с конфигурированием ПЛК ОВЕН ПР200 можно приступать к разработке логики управляющей программы. Учтя все требования к разработке проекта, указанные в документе выше, была составлена логическая схема работы управляющей программы, представленная на рисунке 40.

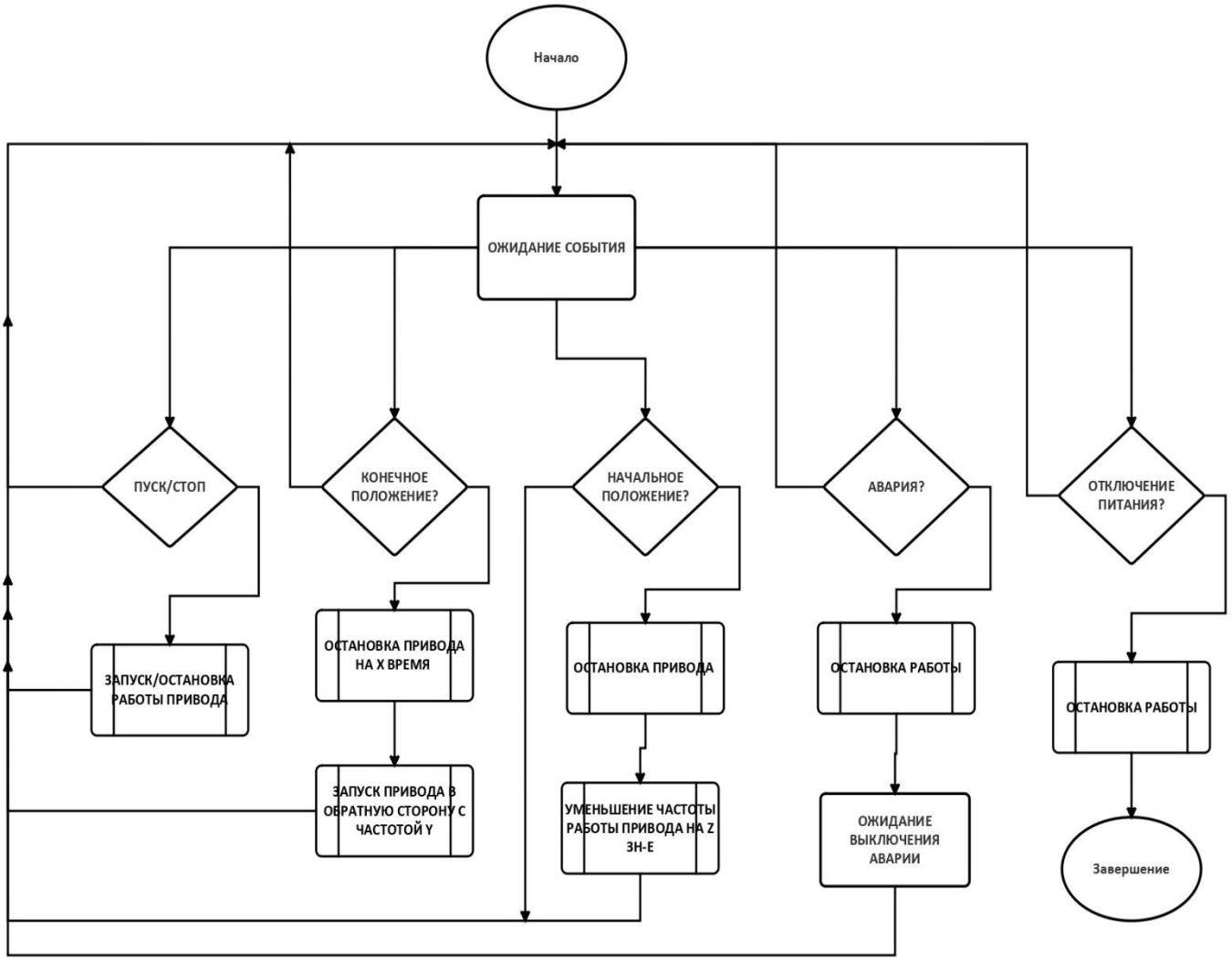


Рисунок 40 – Логическая схема работы управляющей программы

Для упрощения разработки алгоритма всего проекта в среде OWEN Logic, разбили его на несколько частей. Первая часть включает в себя алгоритм логики приведения в движение привода суппорта токарного станка и его остановки. Были созданы и привязаны: кнопка ПУСК/СТОП, подключенная к функциональному блоку RTRIG, тумблер АВАРИЯ, датчики НАЧ.ПОЛ и КОН.ПОЛ, выходная переменная ДВИЖЕНИЕ.

Функция RTRIG (детектор переднего фронта) заключается в генерации единичного импульса на выходе Q при каждом переходе дискретного входного сигнала I из состояния "ложь" в состояние "истина" (нарастающий фронт), рисунки 41 и 42 [2].

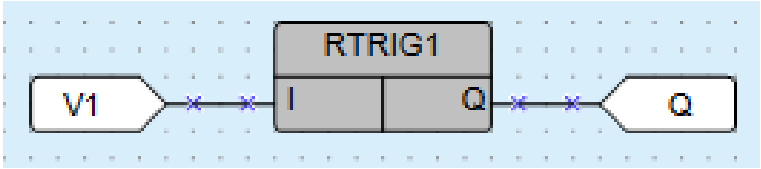


Рисунок 41 – Пример использования ФБ RTRIG

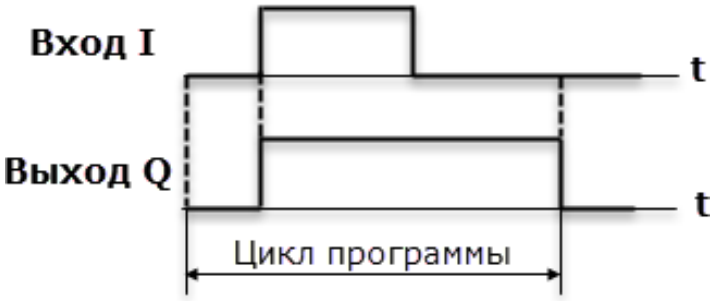


Рисунок 42 – Диаграмма работы ФБ RTRIG

Логика работы движения такова: по нажатию кнопки ПУСК/СТОП происходит реакция ФБ RTRIG, генерируется единичный импульс по переднему фронту, который проходит через функциональные блоки OR с датчиками, доходя до функционального блока RS – триггер с приоритетом выключения. Затем на выходе RS – триггера Q появится логическая «1» по фронту сигнала на входе S от сигнала кнопки ПУСК/СТОП. В итоге получается логическая единица в значении переменной ДВИЖЕНИЕ, которая и будет сигналом того, что привод суппорта токарного станка пришел в движение.

Также для привязки датчика КОН.ПОЛ, по срабатыванию которого привод суппорта останавливается в течение заданного промежутка времени и быстро возвращается до остановки в начальном положении, были использованы функциональные блоки Импульс включения заданной длительности (TP) и Таймер с задержкой включения (TON), которые позволили реализовать описанное выше условие. Для реализации остановки по повторному нажатию кнопки ПУСК/СТОП была выставлена линия задержки, идущая от выхода RS – триггера ко входу функционального блока AND, с подключенным ко второму входу выход кнопки ПУСК/СТОП.

Отдельно добавили целочисленную переменную для записи значения времени для датчика КОН.ПОЛ.

Общий вид схемы алгоритма движения изображен на рисунке 43.

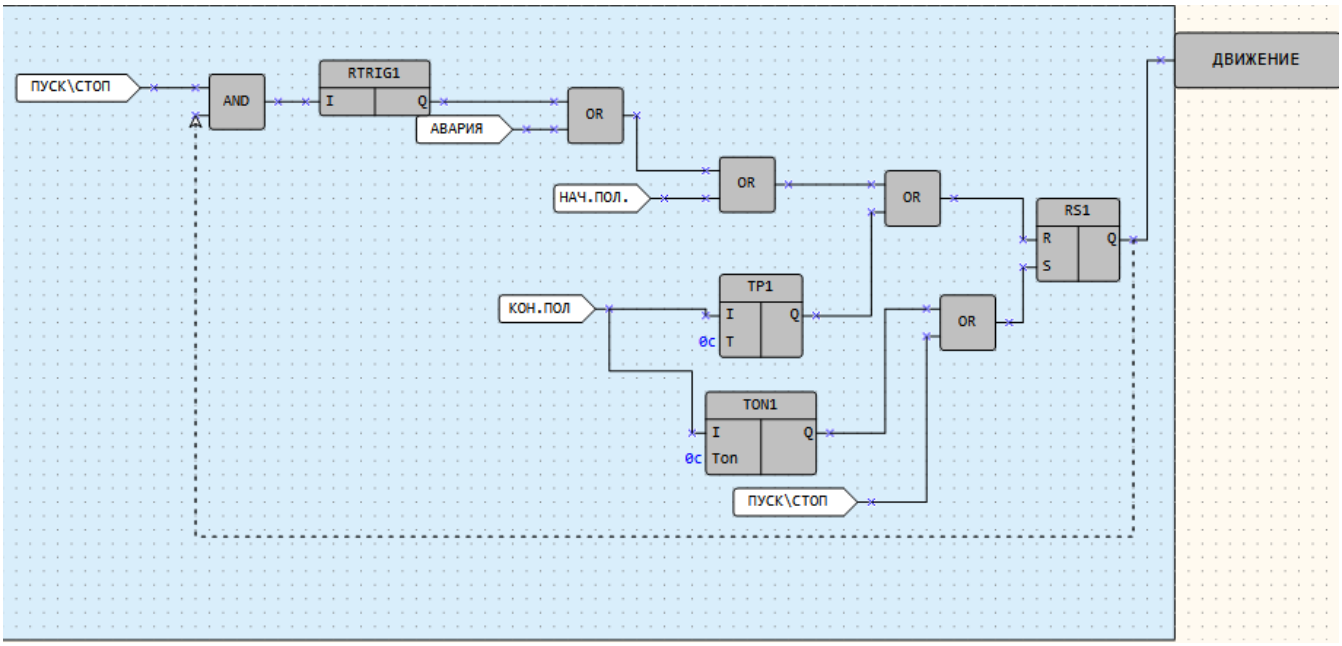


Рисунок 43 – Схема макроса «ПУСК/СТОП»

Блок AND реализует логическую операцию "И". На выходе блока формируется сигнал логической "1" только при одновременном наличии сигналов логической "1" на всех его входах. В противном случае, на выходе будет логический "0". Иллюстрация работы блока представлена на рисунке 44 [2].

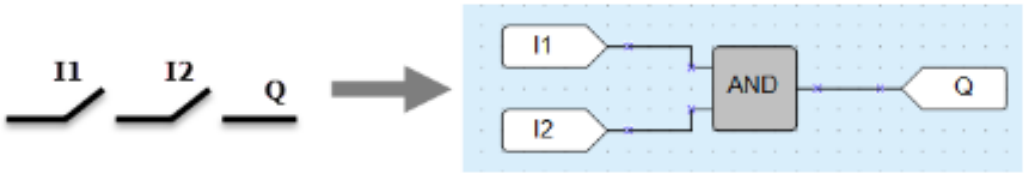


Рисунок 44 – Пример схемы работы ФБ AND

Элемент OR реализует логическую операцию "ИЛИ". Если на любой из входов OR подается сигнал высокого уровня (логическая "1", замкнутый контакт), то выход элемента также переходит в состояние высокого уровня (логическая "1", выход включен), как п оказано на рисунке 45.

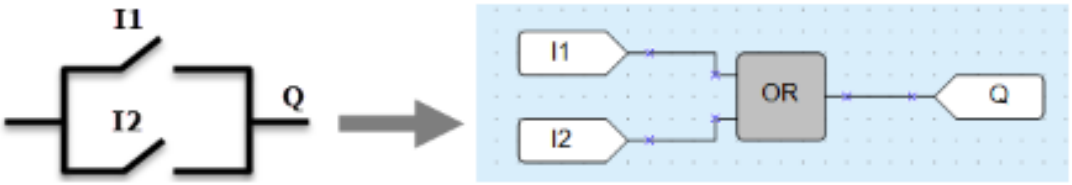


Рисунок 45 – Пример схемы работы ФБ OR

RS-триггер с приоритетом выключения используется для переключения с фиксацией состояния во время поступления коротких импульсов на соответствующий вход. На выходе Q появится логическая «1» по фронту сигнала на входе S, пример и диаграмма работы на рисунках 46 и 47.

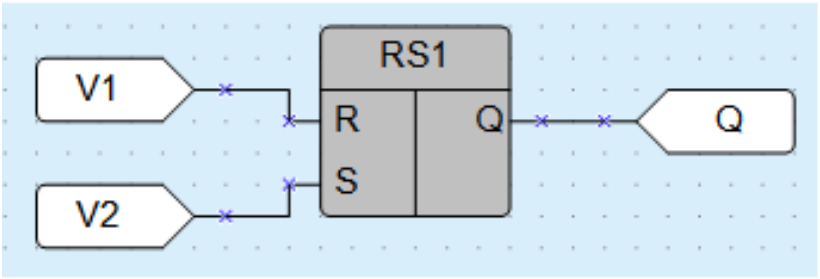


Рисунок 46 – Пример схемы работы ФБ RS-триггер

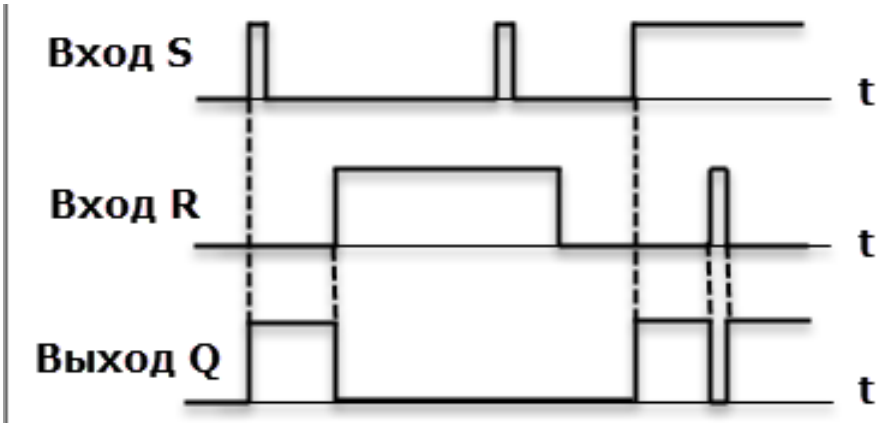


Рисунок 47 – Диаграмма работы ФБ RS-триггер

В случае одновременного поступления сигналов на оба входа приоритетным является сигнал входа R.

Функциональный блок TON (Timer On Delay) реализует задержку включения. Выход Q переходит в состояние "Истина" (логическая "1") только после того, как вход I находится в состоянии "Истина" в течение времени,

равного или превышающего предустановленное значение Ton. Переход в состояние "Ложь" (логический "0") на входе I немедленно деактивирует выход Q. Подробности реализации и временные характеристики представлены на рисунках 48 и 49.

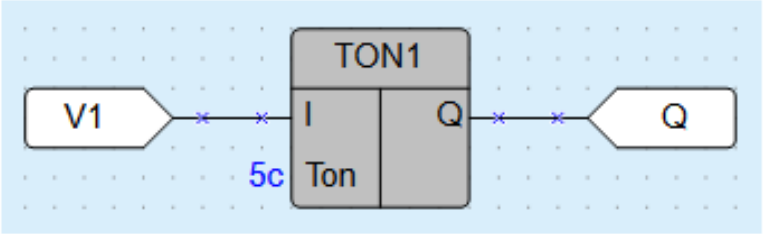


Рисунок 48 – Пример схемы работы ФБ Таймер с задержкой включения (TON)

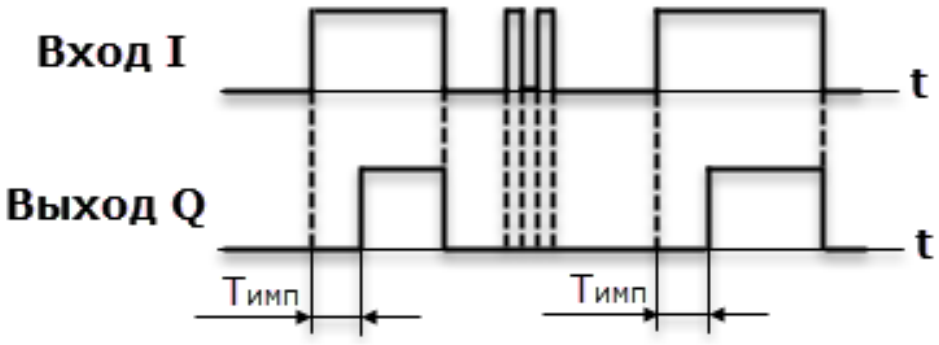


Рисунок 49 – Диаграмма работы ФБ Таймер с задержкой включения (TON)

Блок формирует выходной импульс заданной длительности (TP) в ответ на входной импульс. Появление на входе I фронта сигнала логической "1" активирует выход Q, устанавливая его в состояние логической "1". В течение времени $T_{имп}$, определяющего длительность выходного импульса, блок игнорирует любые изменения входного сигнала I. По истечении времени $T_{имп}$ выход Q возвращается в состояние логического "0", завершая формирование импульса. Принцип работы и временные диаграммы представлены на рисунках 50 и 51 [2].

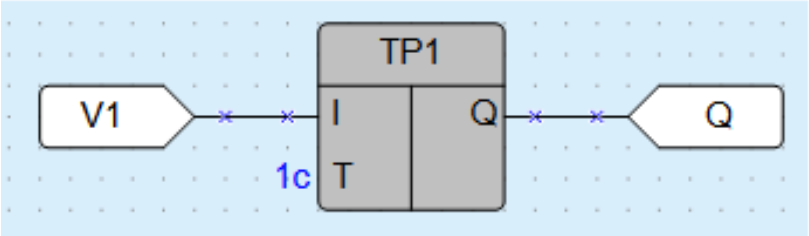


Рисунок 50 – Пример схемы работы ФБ Импульс включения заданной длительности (TP)

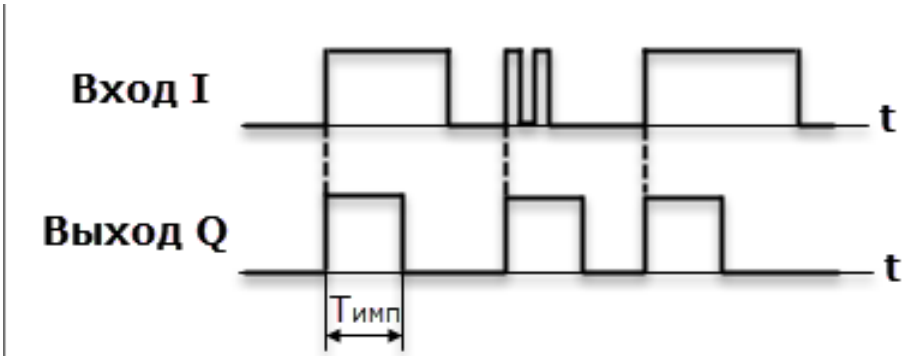


Рисунок 51 – Диаграмма работы ФБ Импульс включения заданной длительности (TP)

Закончив разбор работы алгоритма макроса «ПУСК/СТОП» можно приступать к следующему этапу. В этом этапе будет описана логика работы макроса «ЧАСТОТА».

Макрос – это пользовательский алгоритм для использования в проекте. Он не только позволяет адаптивно использовать рабочее пространство в среде OWEN Logic, упорядочивая его, но и импортировать его в другие проекты.

По заданию, контроллер управляет преобразователем частоты, а тот в свою очередь приводом продольного перемещения суппорта токарного станка, который совершает возвратно-поступательное движение, также присутствует разница в скорости работы при движении от одного концевого выключателя к другому. Исходя из вышеописанных требований, используем две целочисленные переменные «УСТАНОВКА ЧАСТ.НАЧ.» и «УСТАНОВКА ЧАСТ.КОН.» с уже имеющимися.

Суть алгоритма такова: на функциональный блок SELL подаются два значения частоты – начальная и конечная, переключать и контролировать эти значения будет сигнал, идущий от концевых выключателей (датчиков). Соответственно, при каждом срабатывании одного из датчиков, сигнал поступает на функциональный блок RS – триггер с приоритетом выключения вследствие чего, частота работы привода продольного перемещения суппорта будет меняться. Конечное значение частоты будет записываться в переменную «ЗАДАН.ЧАСТ».

Для реализации задержки работы привода при срабатывании датчика используется уже знакомый функциональный блок таймер с задержкой включения (TON).

Схема работы алгоритма макроса «ЧАСТОТА» изображена на рисунке 52.

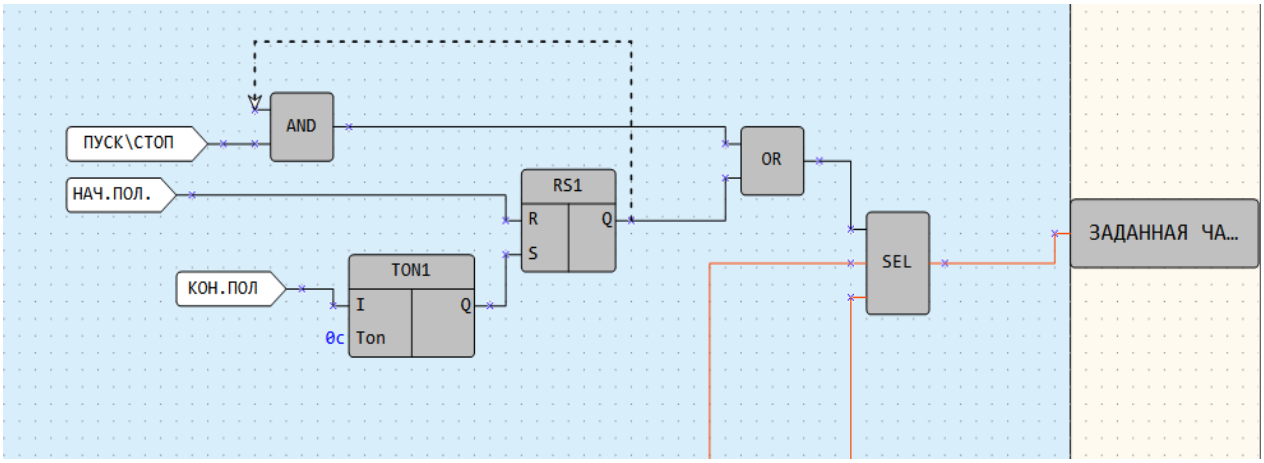


Рисунок 52 – Схема работы макроса «ЧАСТОТА»

Функция Выбор (SEL, fSEL) в OWEN Logic представляет собой тернарную условную операцию сравнения. Если значение V1 равно логическому «0», то выходным сигналом этого блока становится входной сигнал V2. В случае, когда V1 равно логической «1», на выходе блока будет сигнал V3. Пример работы функционального блока SEL можно увидеть на рисунке 53 [2].

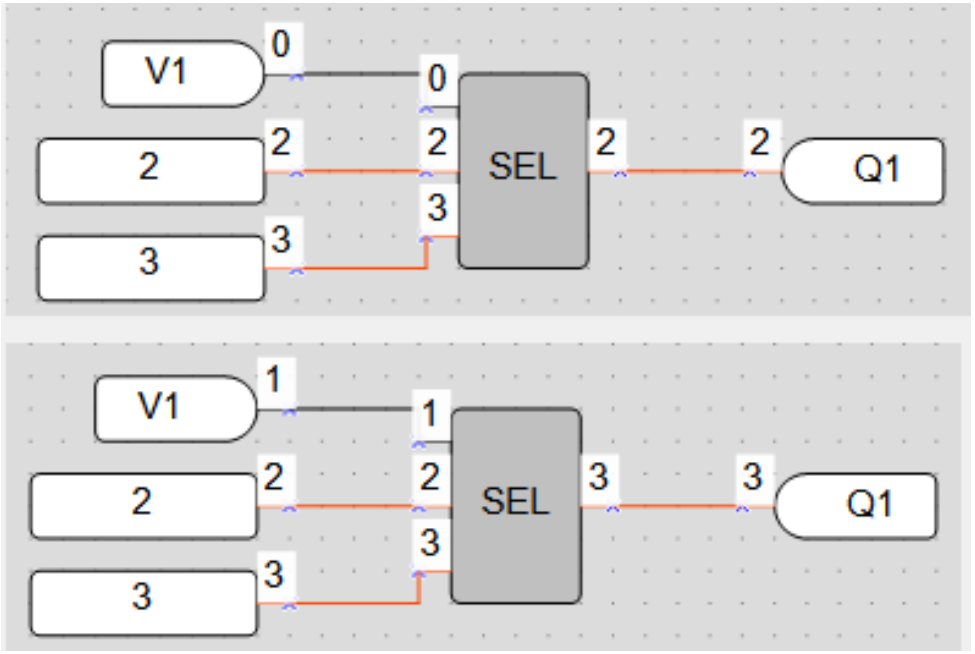


Рисунок 53 – Схема работы ФБ SELL

После окончания работы с макросом «ЧАСТОТА», можно приступать к регулировке направления движения привода суппорта токарного станка. По условию, при достижении одного из концевых выключателей, привод должен менять направление своего движения в противоположную сторону. Для того, чтобы реализовать это условие, добавим еще одну переменную «НАПР.КОН.», которая будет иметь значение либо 0 либо 1, в зависимости от направления.

Суть алгоритма изменения направления: если значения частот переменных «УСТАНОВКА ЧАСТ.КОН.» и «ЗАДАН.ЧАСТ» равны, то значение переменной «НАПР.КОН.» будет равно 1. Таким образом, получается, что направление из конца в начало будет обозначать 1, а из начала в конец – 0. Реализовано данное условие было через функциональный блок EQ. Схема работы алгоритма изображена на рисунке 54.

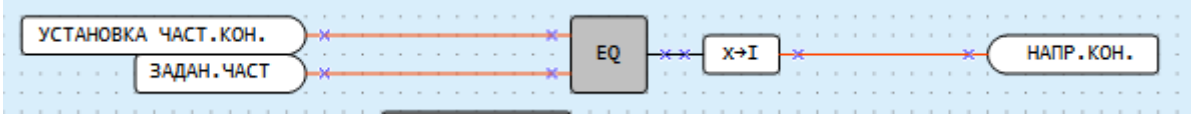


Рисунок 54 – Схема работы алгоритма смены направления

Суть работы функционального блока EQ в том, чтобы сравнивать два значения и формировать на выходе единицу, если равенство верное. На рисунке 55 изображен пример схемы.

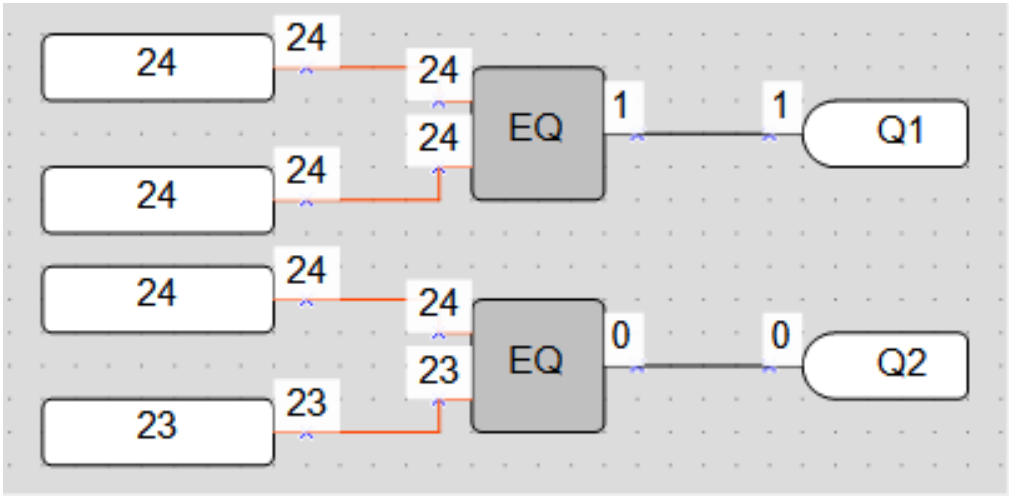


Рисунок 55 – Пример схемы работы ФБ EQ

После окончания работы с направлением остается последняя часть – преобразователь частоты. В выборе дополнительных средств разработки мы указывали его, как средство для управления электродвигателем. С помощью преобразователя частоты будет осуществляться контроль над скоростью вращения двигателя, его запуском и остановкой, его направлением и защитой в аварийном режиме.

Взаимодействие по протоколу Modbus RTU основано на чтении и записи данных в регистры, которые представляют собой области памяти, содержащие определенную информацию. Каждый регистр идентифицируется уникальным адресом, представленным как в десятичной, так и в шестнадцатеричной форме. Шестнадцатеричное представление адреса обычно обозначается суффиксом "h", указывающим на использование шестнадцатеричной системы счисления [1].

После окончания работы с направлением остается последняя часть – преобразователь частоты. В выборе дополнительных средств разработки мы указывали его, как средство для управления электродвигателем. С помощью преобразователя частоты будет осуществляться контроль над скоростью

вращения двигателя, его запуском и остановкой, его направлением и защитой в аварийном режиме.

Перед началом разработки алгоритма следует ознакомиться с основными регистрами преобразователя частоты.

Регистр "командное слово" является ключевым для управления приводом. Он предоставляет интерфейс для выполнения таких операций, как запуск, остановка, изменение направления вращения и сброс аварийных состояний.

"Командное слово" состоит из битового поля, где каждый бит представляет собой отдельную команду. Протокол Modbus RTU, используемый на интерфейсе RS-485, требует передачи данных в виде целых чисел. Поэтому, отдельные биты командного слова кодируются в целочисленное значение посредством применения стандартной битовой маски. В результате, двоичное представление команд преобразуется, например, в число 126 для передачи по интерфейсу.

На рисунке 56 отображены биты, отвечающие за формирование сигналов управления ПЧ.

Биты формирования сигналов формирования управления преобразователя частоты

№ регистра	Описание битов			
0101h	Сигналы управления	Бит	0	1
		0	Стоп	Пуск
		1	Вперед	Назад
		3	Нет сброса	Сброс ошибки

Рисунок 56 – Описание битов формирования сигналов управления ПЧ

В системе управления, команда "Пуск назад" для электродвигателя представлена числом 3, а команда "Сброс ошибки" - числом 8. Преобразование двоичных кодов в целые числа и обратно осуществляется с использованием встроенного калькулятора Windows [1].

Частота ПЧ конфигурируется через регистр с адресом 0102h. Этот регистр принимает значения в диапазоне [0, 5000], где 5000 представляет

собой максимальную частоту 50 Гц. Значение, записанное в регистр, линейно отображается на частоту ПЧ в пределах указанного диапазона.

Регистр 0121h, предназначенный для индикации аварийных ситуаций, сигнализирует о возникновении ошибки в электроприводе при наличии в нем ненулевого значения.

Значение, хранящееся в регистре 0124h, представляет собой выходную частоту, которая управляет частотой тока, питающего электродвигатель.

Пятый регистр с номером 0127h – это сам выходной ток, т.е. ток электродвигателя.

Шестой регистр с номером 0129h – выходной момент.

Четыре последних регистра используются для отслеживания ключевых показателей преобразователя частоты (ПЧ).

Поскольку переменные частотного преобразователя представлены в целочисленном формате, для получения более точных значений используется масштабирование путем деления.

После того как определены все регистры ПЧ, можно приступить к разработке алгоритма в OWEN Logic. В окне «Настройка прибора» в списке слева во вкладке «Интерфейсы» мы уже добавляли нужное подключение и настроили его аналогично настройкам подключения преобразователя частоты. В нашем случае данным элементом будет являться ПЧ ВЕСПЕР Е2-8300 с адресом 1. На рисунке 38, изображенном выше, мы указывали в настройках элемента его имя и адрес в сети. Здесь же необходимо было настроить список опрашиваемых параметров. Для этого добавили сетевые переменные, которые будут храниться в регистрах опрашиваемого устройства.

В рамках настройки устройства необходимо произвести замену всех экземпляров переменной "АВАРИЯ" на "АВАРИЯ.ПЧ". Ранее сигнал "АВАРИЯ" генерировался нажатием соответствующей кнопки на панели оператора, сигнализируя об аварийном состоянии электропривода. Теперь же, в текущем проекте, этот сигнал формируется внутренней системой диагностики частотного преобразователя (ПЧ), что и обуславливает

необходимость замены переменной.

Так как компоненты программы управления могут соединяться только через линии связи, передающие данные одного типа (булевские, целые или с плавающей точкой), для соединения входов и выходов разных типов потребовались блоки преобразования. Результат добавления блока «Преобразователь в булевское значение» представлен ниже на рисунке 57 [1].

Рисунок 57 – Пример использования блоков преобразования

"Преобразователь в булевское значение" преобразует входной сигнал в логическое значение. Если входное значение равно 1 или больше, на выходе будет "истина" (1). В противном случае на выходе будет "ложь" (0) [2].

В результате проделанных действий программа управления должна принять вид, представленный ниже на рисунке 58.

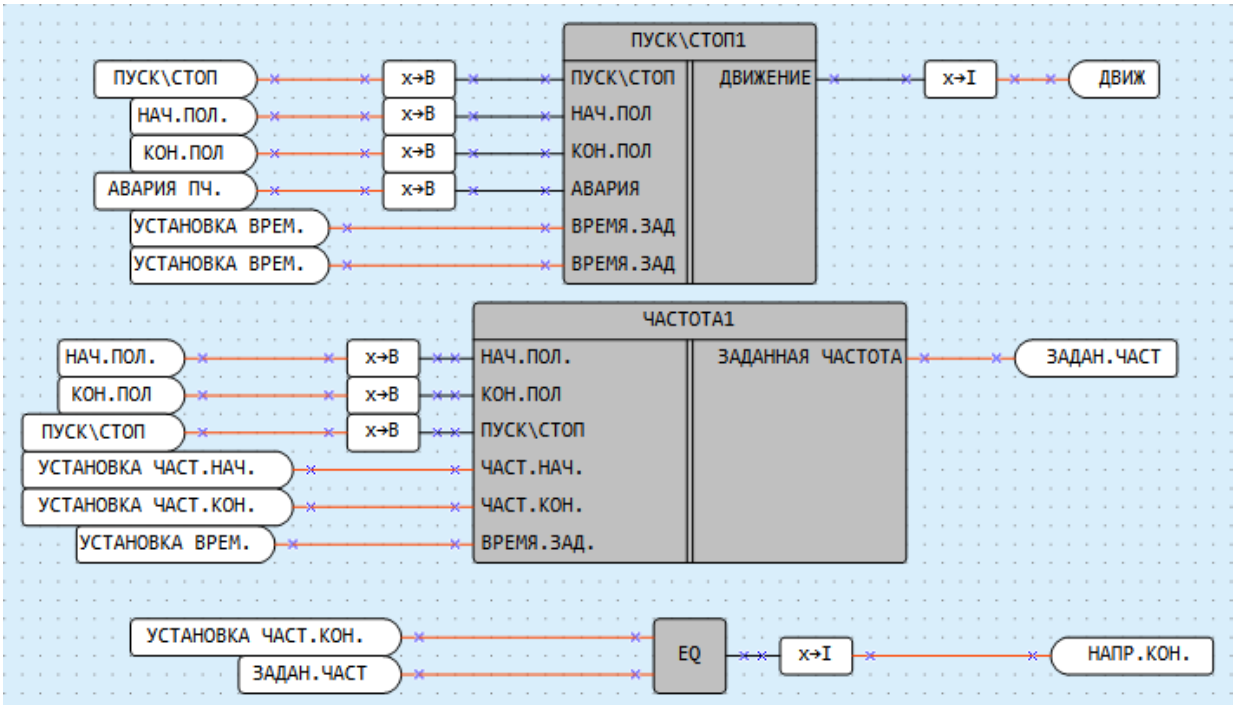


Рисунок 58 – Общий вид программы управления

Чтобы обеспечить передачу данных между контроллером, панелью оператора и преобразователем частоты, и в частности, чтобы оператор мог вводить новые значения сетевых переменных с панели и эти значения передавались в управляющую программу контроллера, в алгоритм управления был добавлен фрагмент, показанный на рисунке 59.

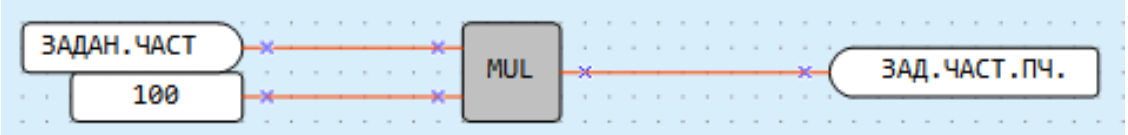


Рисунок 59 – Фрагмент алгоритма управления

Для выполнения числового преобразования над целочисленными значениями мы использовали блок MUL. Результатом операции функции на выходе является произведение входных значений.

При выполнении арифметических операций в функции, если результат превышает максимальное 32-битное значение (4294967295), происходит переполнение, и старшие биты, выходящие за пределы 32-битной разрядности, отбрасываются. На рисунке 60 показано, как использовать функциональные блоки для вычисления произведения 24 2 2 * 7 = 672 [2].

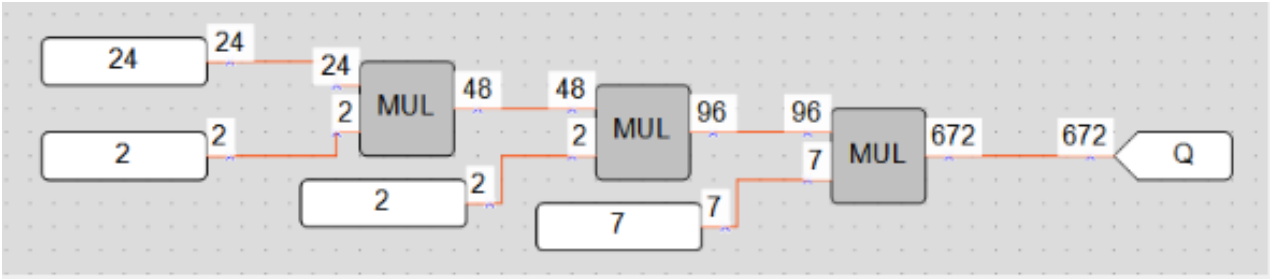


Рисунок 60 – Работа ФБ MUL

Для работы с командным словом в управляющую программу следует ввести функциональный блок в виде готового макроса из онлайн-базы «Менеджер компонентов» OWEN Logic, собирающий дискретные сигналы в битовую маску, изображенный на рисунке 61.

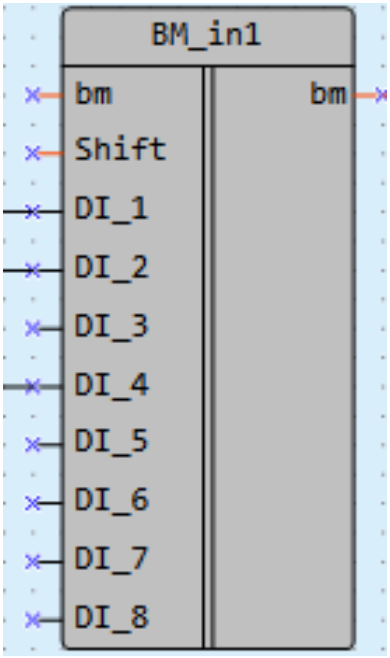


Рисунок 61 – Функциональный блок – макрос «Битовая маска»

Теперь осталось подключить переменные к правильным входам и выходам, учитывая описание битов формирования сигналов управления преобразователя частот. Получившееся подключение изображено на рисунке 62.

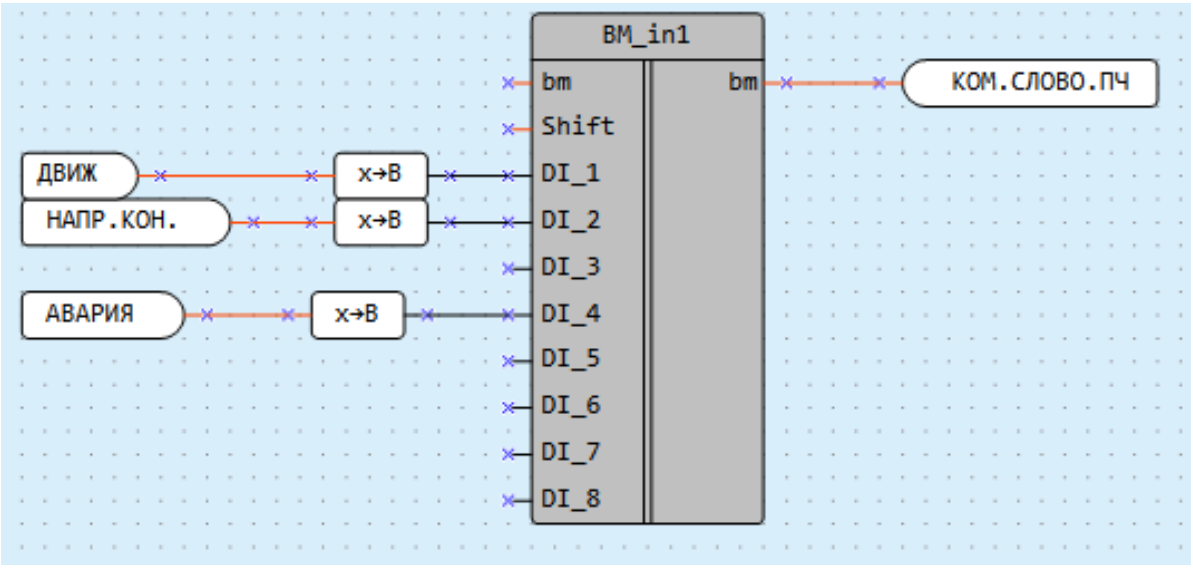


Рисунок 62 – Фрагмент алгоритма проекта с битовой маской.

После выполнения последнего этапа добавления битовой маски разработка управляющей программы заканчивается. Общая схема всего алгоритма проекта изображена на рисунке 63.

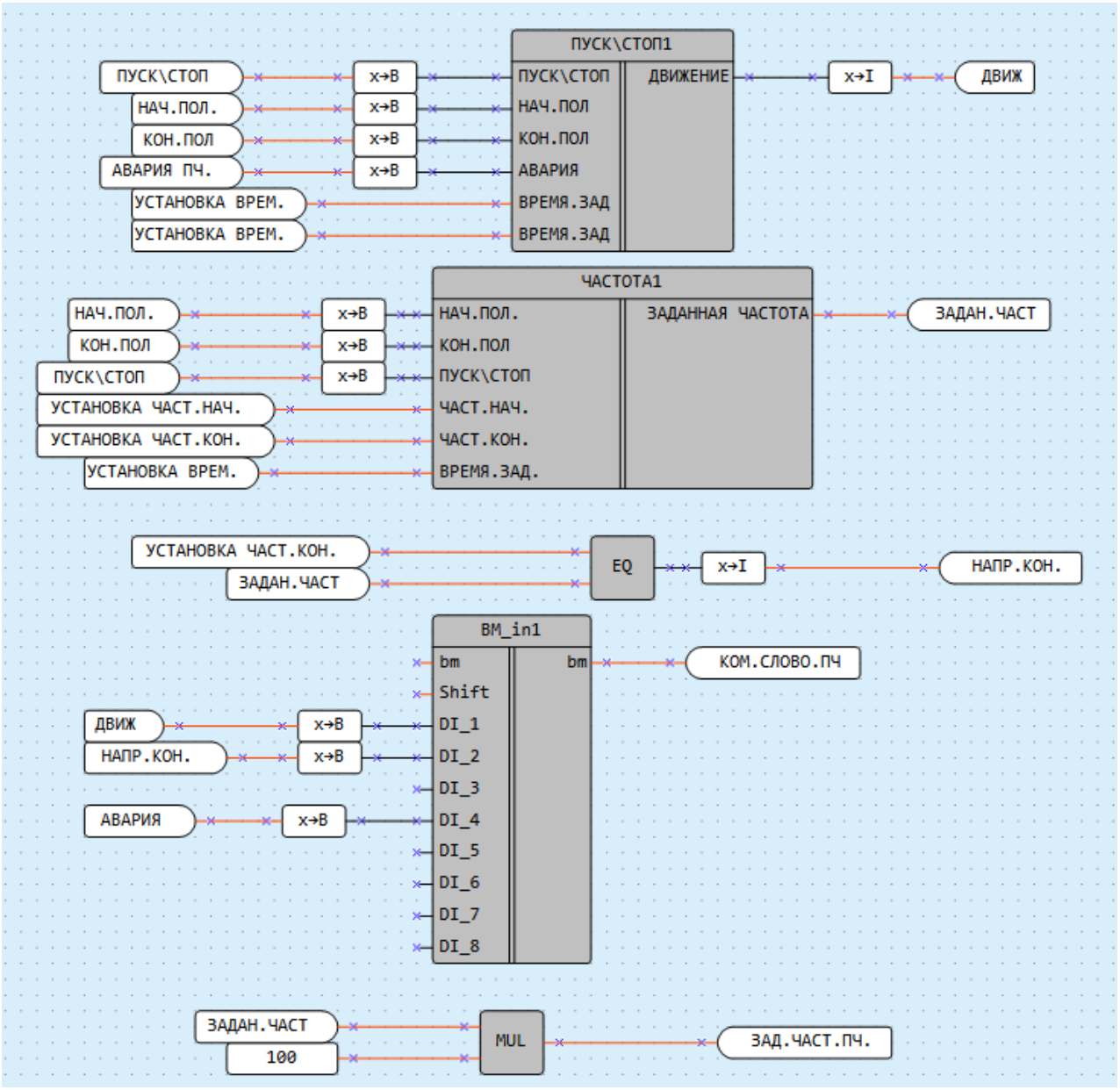


Рисунок 63 – Общая схема всего проекта

Итоговую управляющую программы необходимо загрузить в контроллер. Подключить ПЛК ОВЕН ПР200 к персональному компьютеру возможно с помощью кабеля mini USB, как показано на рисунке 64

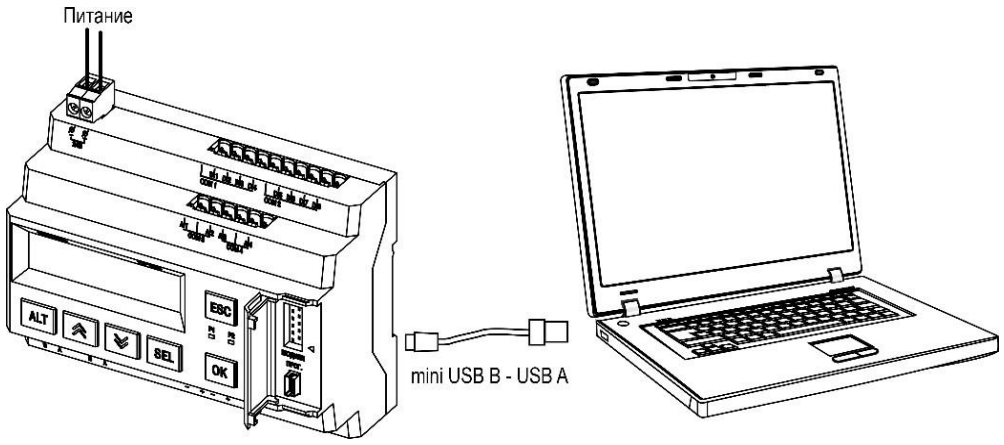


Рисунок 64 – Подключение ПЛК и ПК

1.2.3 Конфигурирование преобразователя частоты ВЕСПЕР Е2-8300

Закончив разработку управляющей программы, можно приступать к описанию конфигурирования преобразователя частоты ВЕСПЕР Е2 – 8330.

При подаче питающего напряжения на ПЧ, на дисплее пульта управления в течение 3-5 секунд отображается значение напряжения питания. По истечении этого времени, дисплей переключается на отображение заданной частоты, установленной на 5 Гц. В течение всего процесса инициализации (3-5 секунд) индикаторы дисплея и светодиод "ВРАЩЕНИЕ ВПЕРЕД" находятся в режиме мигания. Функциональное назначение кнопок управления ПЧ представлено на рисунке 65 [1].

Назначение кнопок пульта управления

Кнопка	Режим управления	Режим программирования
	Увеличение заданной частоты	Увеличение номера или значения параметра
	Уменьшение заданной частоты	Уменьшение номера или значения параметра
СБРОС	Переход к следующему разряду, сброс аварийного состояния	Переход к следующему разряду
ВВОД	Не используется	Чтение и запись значений параметров
ПРОГ.	Переход в режим программирования	Выход из режима программирования
ПУСК / СТОП	Пуск и остановка двигателя	Пуск и остановка двигателя
ВПЕРЕД / НАЗАД	Выбор направления вращения	Не используется
ЧАСТОТА	Задание частоты	Не используется

Рисунок 65 – Назначение кнопок ПЧ ВЕСПЕР Е2-8300

Режим управления устанавливается после подачи питания на преобразователь. В этом режиме доступны следующие функции:

– пуск/остановка двигателя при установленном значении параметра 1-01 = 0000 (источник команды ПУСК/СТОП – пульт управления); пуск двигателя срабатывает нажатием кнопки «ПУСК/СТОП», остановка – повторным нажатием этой же кнопки;

– при параметре 1-06, равном 0000, управление частотой осуществляется посредством кнопок пульта; кнопка "▲" выполняет инкремент частоты, кнопка "▼" - декремент, а кнопка "◄/СБРОС" обеспечивает навигацию по разрядам частоты для изменения. В случае, когда параметр 1-06 имеет значение 0001, источником задания частоты является потенциометр "ЧАСТОТА";

– Индикация ошибки преобразователя сбрасывается нажатием кнопки «◄/СБРОС».

Нажатие кнопки "ПРОГ" переводит устройство в режим программирования, в котором доступны следующие возможности:

– выбор номера параметра кнопками «▲» или «▼», переход к следующей цифре – кнопкой «◄/СБРОС»;

– просмотр значения параметра нажатием кнопки «ВВОД»;

– изменение значения параметра кнопками «▲» или «▼», переход к следующей цифре – кнопкой «◄/СБРОС»;

– сохранение значения измененного параметра нажатием кнопки «ВВОД»;

– выход из режима программирования нажатием кнопки «ПРОГ.» [1].

Состояние индикаторов пульта управления:

– светодиод «УПР. ДИСТАНЦИОННО» загорается при значениях параметра 1-00 = 0001 (управление от внешних клемм), 0002 (от последовательной линии связи) или 0003 (от встроенного PLC);

– светодиод «РЕГ. ДИСТАНЦИОННО» загорается при значениях параметра 1-06 = 0001 (задание частоты с потенциометра пульта управления), 0002 (от внешнего потенциометра), 0003 (сигналами БОЛЬШЕ/МЕНЬШЕ) или 0004 (от последовательной линии связи);

– светодиод «ВПЕРЕД» – индикация вращения вперед (мигает при остановке, горит при вращении);

– светодиод «НАЗАД» – индикация вращения назад (мигает при остановке, горит при вращении);

– светодиод «ПРОГ.» загорается в режиме программирования, на цифровом дисплее отображается номер или значение параметра;

– светодиод «Гц» загорается в режиме отображения частоты.

В состоянии покоя двигателя на цифровом дисплее отображается заданная частота в мигающем режиме. При работающем двигателе дисплей показывает фактическую выходную частоту, отображаемую в статичном режиме (без мигания).

– светодиод «В» светится в режиме отображения напряжения;

– светодиод «А» светится в режиме отображения выходного тока.

Рисунок 66 иллюстрирует управление электродвигателем, включая запуск, остановку, регулировку частоты и изменение направления вращения. Текущие параметры отображаются на рисунке 67, а схема чтения и записи параметров представлена на рисунке 68 [1].

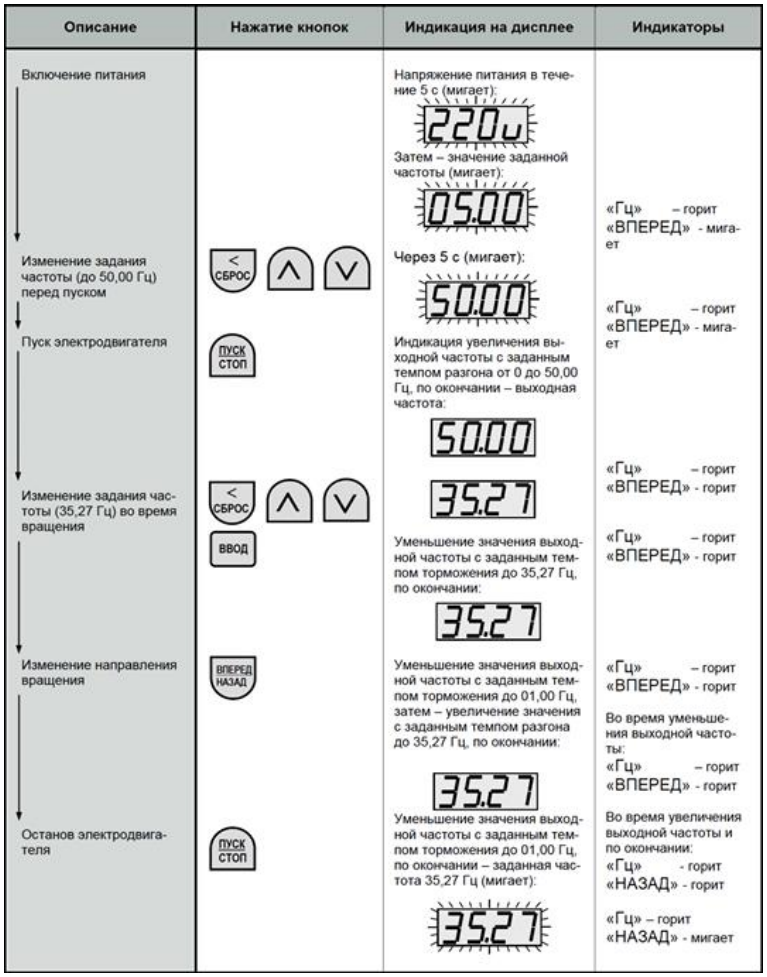


Рисунок 66 – Схема пуска и остановки электродвигателя, изменение задания частоты, направления вращения

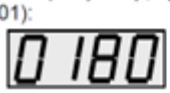
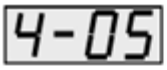
Описание	Нажатие кнопок	Индикация на дисплее	Индикаторы
Включение питания ***Примечание**. Для индикации текущих параметров необходимо запрограммировать константы группы 4* *4-00 = 0001 – ток двигателя;* *4-01 = 0001 - напряжение на двигателе;* *4-02 = напряжение звена постоянного тока;* *4-03 = 0001 – состояние встроенного PLC;* *4-05 = 0001…0004 – скорость вращения в об/мин.*		Напряжение питания в течение 5 с (мигает): Затем – значение заданной частоты (мигает): 	«Гц» – горит «ВПЕРЕД» - мигает
Пуск электродвигателя	ПУСК СТОП	Индикация увеличения скорости вращения с заданным темпом разгона от 0 до 180 об/мин (например, при 4-05 = 0001): 	«Гц» – горит «ВПЕРЕД» - горит «Гц» – горит «ВПЕРЕД» - горит
Индикация номера последней просмотренной константы, например, 4-05	ПРОГ		«ПРОГ» - горит «ВПЕРЕД» - горит
Индикация напряжения двигателя	ПРОГ		«В» – горит «ВПЕРЕД» - горит
Индикация напряжения звена постоянного тока	ПРОГ		«В» – горит «ВПЕРЕД» - горит
Индикация тока двигателя	ПРОГ		«А» – горит «ВПЕРЕД» - горит
Индикация скорости вращения двигателя	ПРОГ		«Гц» – горит «ВПЕРЕД» - горит
Останов двигателя	ПУСК СТОП	Уменьшение значения скорости с заданным темпом торможения до 0 об/мин, по окончании – задание частоты 5,00 Гц (мигает): 	Во время уменьшения выходной частоты: «Гц» – горит «ВПЕРЕД» - горит «Гц» – горит «ВПЕРЕД» - мигает

Рисунок 67 – Схема пуска и остановки электродвигателя, индикация текущих параметров

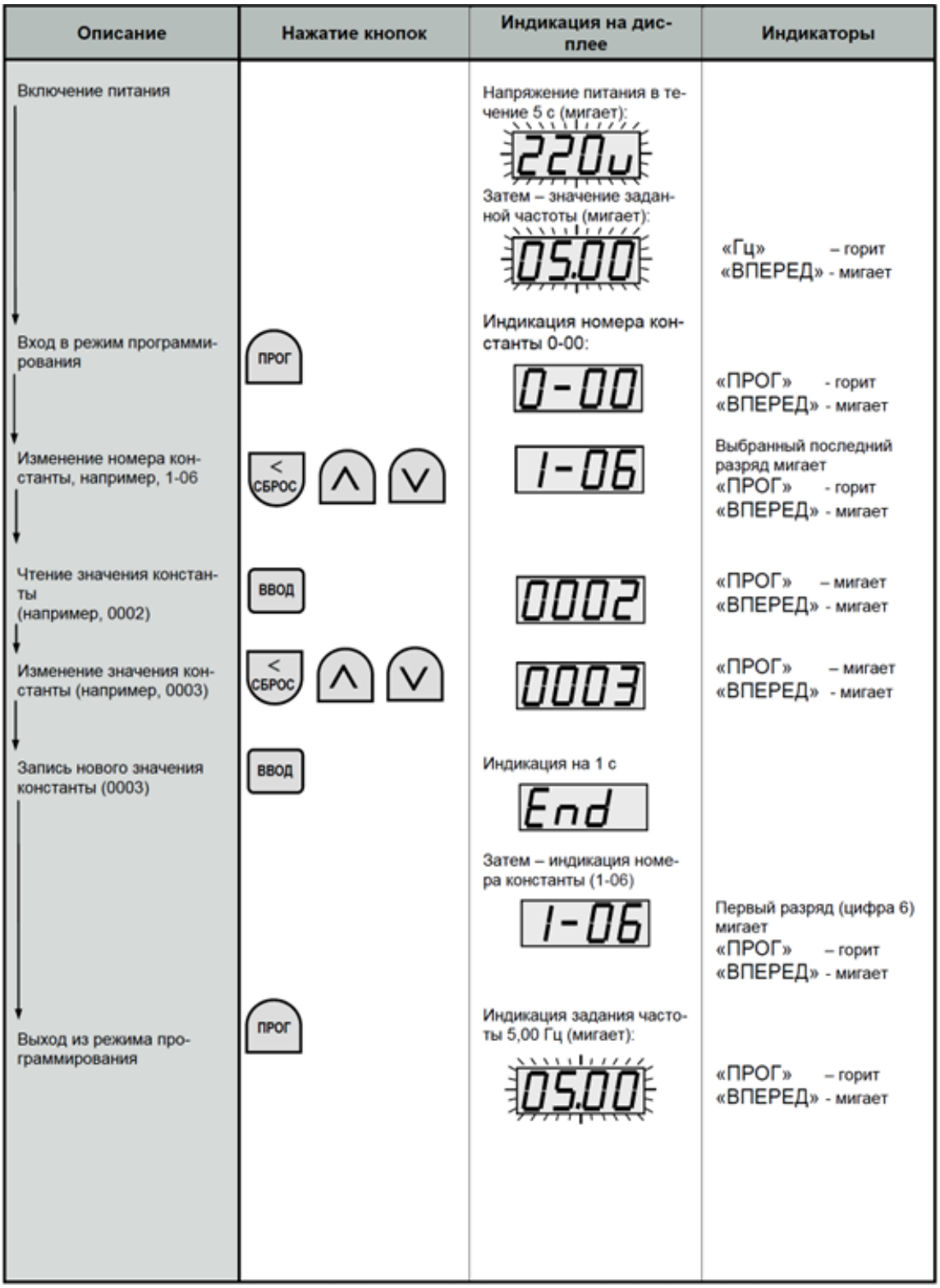


Рисунок 68 – Схема чтения и записи значений параметров

Чтобы упростить подбор оптимальных настроек для преобразователя, на рисунках 69 и 70 приведен список программируемых параметров, который поможет быстро сориентироваться в их выборе.

Перечень программируемых параметров

№ группы параметров	Наименование группы	Краткое описание функций
1	2	3
0-	Режимы управления	Выбор режимов управления. Индикация текущих параметров
1-	Режимы пуска/остановки и задания частоты	Способы пуска/остановки и задания частоты
2-	Режимы ручного/автоматического перезапуска	Режимы перезапуска при кратковременной к потере питания и аварийных остановках
3-	Параметры функционирования	Уровень доступа к константам. Время разгона и торможения. S-кривые разгона/торможения. Верхний и нижний пределы рабочей частоты. Выбор запрещенных частот. Несущая частота ШИМ. Параметры торможения постоянным током. Колебательный режим
4-	Режимы работы дисплея	Включение индикации выходного тока, напряжения, скорости и других параметров
5-	Многофункциональные входы	Программирование входных клемм управления для выполнения различных функций
6-	Установка фиксированных скоростей и скорости шагового режима с пульта управления	Задание до 8 фиксированных скоростей и скорости шагового режима
7-	Настройка аналоговых входных сигналов	Выбор коэффициентов усиления, смещения и наклона характеристики аналоговых входов AIN и AI2(S6)
8-	Настройка многофункциональных выходных реле и выходных сигналов	Программирование аналогового выхода для контроля основных параметров. Программирование дискретных выходов для выполнения различных функций
9-	Функции защиты двигателя и преобразователя	Настройка характеристик: защиты двигателя от перегрузки; предотвращения срыва; защиты от перегрузки по моменту

Рисунок 69 – Перечень программируемых параметров

1	2	3
10-	Настройка характеристик U/f	Выбор одной из 18 фиксированных характеристик U/f и характеристики, задаваемой пользователем
11-	Настройка параметров ПИД-регулятора	Выбор режима работы ПИД-регулятора, настройка параметров ПИД-регулятора для регулирования технологических параметров (например, давления, температуры) при работе преобразователя с датчиком обратной связи
12-	Настройка ПИД-регулятора при потере обратной связи	Настройка параметров определения потери обратной связи ПИД-регулятора. Способы управления приводом при потере обратной связи ПИД-регулятора
13-	Параметры управления по последовательной линии связи	Выбор параметров при управлении преобразователем от PLC или ПК
14-	Параметры автонастройки	Параметры двигателя, которые записываются в память преобразователя в процессе автонастройки
15-	Индикация параметров преобразователя и восстановление заводских уставок	Индикация модели преобразователя и версии программного обеспечения. Список последних 3-х аварийных сообщений. Индикация времени наработки преобразователя. Восстановление заводских уставок

Рисунок 70 – Перечень программируемых параметров

В случае, если информации из рисунков 69 и 70 недостаточно для работы с преобразователем, необходимо обратиться к руководству по эксплуатации устройства.

Для обеспечения управления частотным преобразователем (ПЧ) от контроллера необходимо организовать последовательное соединение по протоколу RS-485 с использованием экранированной витой пары. Конфигурация параметров последовательного порта ПЧ должна соответствовать значениям, представленным на рисунке 71.

Параметры преобразователя частоты, используемые при работе с последовательным портом

№ параметра	Наименование	Значение и описание	Устанавливаемое значение
1	2	3	4
1-00	Выбор источника команды ПУСК/СТОП	0000: Пульт управления 0001: Внешние клеммы 0002: Последовательная линия связи – используется при управлении ПЧ от компьютера или внешнего контроллера 0003: Встроенный PLC – используется, если во встроенный PLC загружена программа и требуется управление работой ПЧ под ее управлением	0002
1-06	Выбор источника задания частоты	0000: Кнопки на пульте управления 0001: Потенциометр на пульте управления 0002: Внешний аналоговый сигнал или внешний потенциометр 0003: Задание частоты сигналами, поступающими через многофункциональные входы 0004: Последовательная линия связи – используется при задании частоты ПЧ от компьютера или внешнего контроллера 0005: Импульсное задание частоты с клеммы S5	0004
13-0	Адрес устройства	1 – 254: для каждого из ПЧ, подключенных к одному управляющему компьютеру (контроллеру) задается индивидуальный (не повторяющийся) адрес	0001
13-1	Скорость обмена (бит/с)	0000: 4800 0001: 9600 0002: 19200 0003: 38400	0003
13-2	Количество стоповых бит	0000: 1 0001: 2	0000
13-3	Контроль четности	0000: Отсутствует 0001: Четно 0002: Нечетно	0000
13-4	Формат данных	0000: 8 бит 0001: 7 бит	0000
13-5	Способ остановки при потере связи	0000: Торможение за время 1 (параметр 3-03) 0001: Инерционное торможение 0002: Торможение за время 2 (параметр 3-07) 0003: Продолжение работы	0000
13-6	Время определения потери связи (с)	0,0…25,5	0,0

Рисунок 71 – Параметры преобразователя частоты

При установке параметра 15-6 = 1110 будут восстановлены заводские значения констант для двигателя с частотой питающего напряжения в 50 Гц [1].

1.2.4 Конфигурирование панели оператора ОВЕН СП270

Последним этапом разработки и проектирования является создание графического интерфейса в программе Конфигуратор СП200.

Проанализировав предполагаемый сценарий использования панели оператора, был создан новый проект в среде Конфигуратор СП200. При создании проекта был указан протокол связи Modbus RTU (Панель мастер) для порта PLC, как показано на рисунке 72.

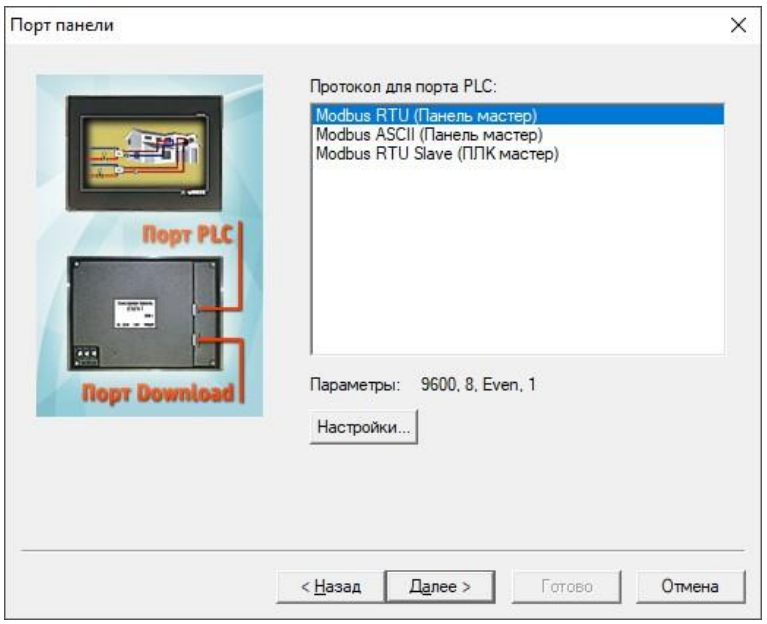


Рисунок 72 – Выбор протокола для порта PLC

Сетевые параметры, введенные в окне "Настройки параметров соединения" на рисунке 73, полностью соответствуют настройкам подключения контроллера.

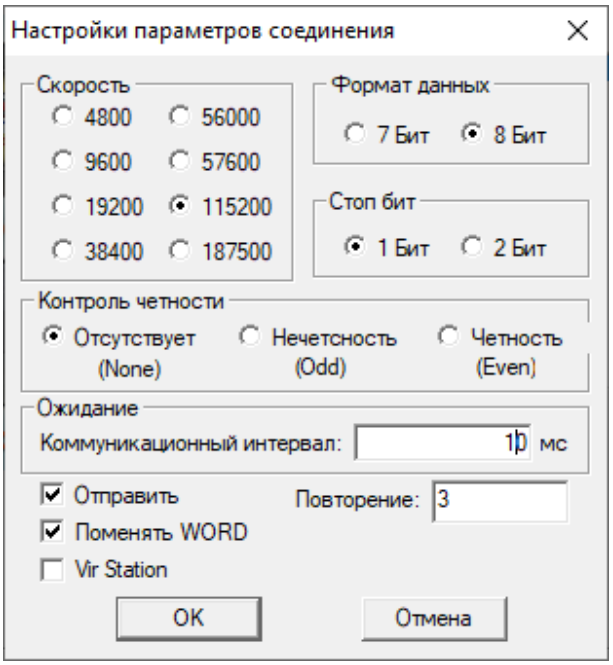


Рисунок 73 – Выбор настроек параметров соединения

Согласно рисунку 74, порт Download не функционирует, и, следовательно, протокол для него не выбирался.

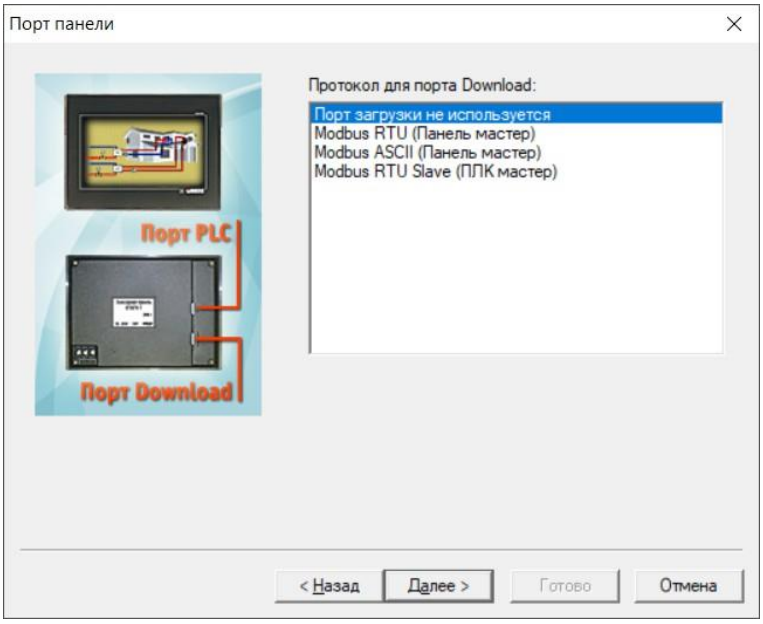


Рисунок 74 – Выбор протокола загрузки

Рисунок 75 показывает главное окно, которое появляется после создания нового проекта.

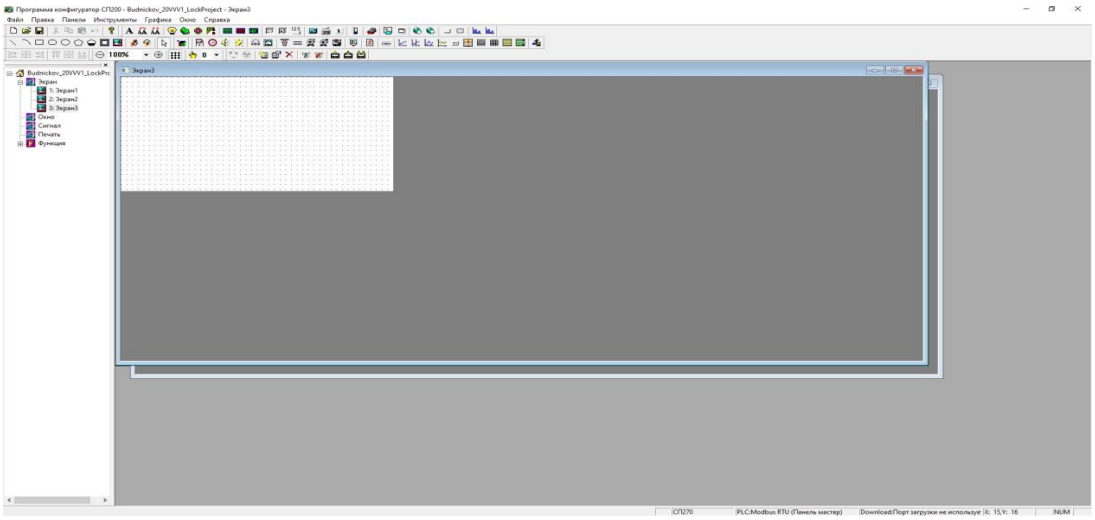


Рисунок 75 – Окно нового проекта в программе Конфигуратор СП200

Снизу приведена таблица 4 с соотношением сетевых переменных, созданных в управляющей программе для ПЛК в OWEN Logic, с элементами интерфейса панели оператора. Все они будут связаны между собой адресами регистров.

Таблица 4 – соотношение сетевых переменных и элементов интерфейса

Имя переменной	Адрес регистра	Тип элемента интерфейса
ПУСК/СТОП	512	Кнопка управления битом
НАЧАЛЬНОЕ ПОЛОЖЕНИЕ	513	Переключатель с подсветкой
КОНЕЧНОЕ ПОЛОЖЕНИЕ	514	Переключатель с подсветкой
СБРОС АВАРИИ	515	Переключатель с подсветкой
ВЫХ.ЧАСТ.	521	Цифровой дисплей
ЧАСТ.КОН	519	Ввод данных
ЧАСТ.НАЧ.	520	Ввод данных
ВРЕМЯ.ЗАД.	518	Ввод данных

Для кнопки «ПУСК/СТОП» был выбран элемент интерфейса «Кнопка управления битом», с выбранным 512 адресом регистра соответствующей

переменной в управляющей программе. Демонстрация на рисунке 76.

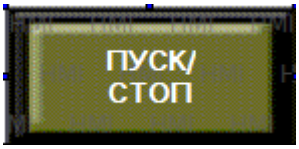


Рисунок 76 – Кнопка «ПУСК/СТОП»

Снизу на рисунке 77 и 78 продемонстрированы внутренние настройки, указанные при создании кнопки «ПУСК/СТОП».

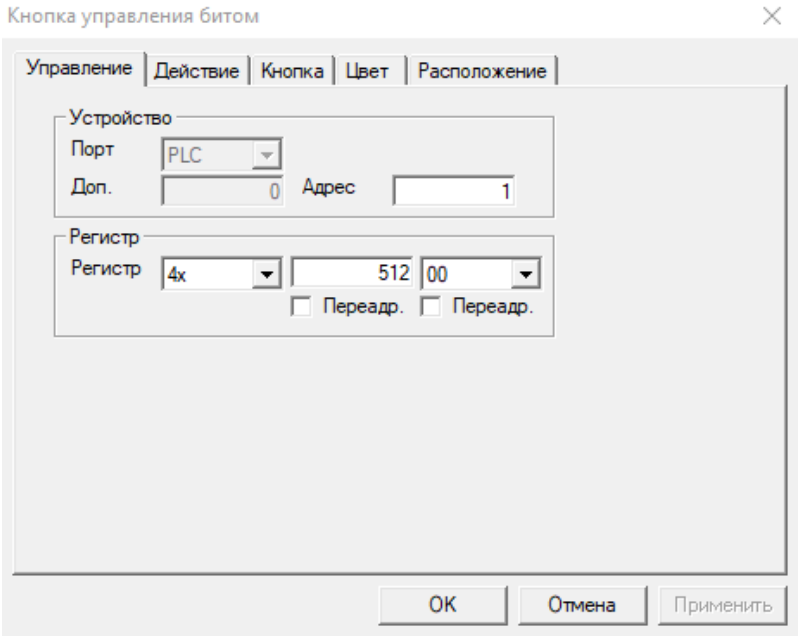


Рисунок 77 – Настройки кнопки «ПУСК/СТОП»

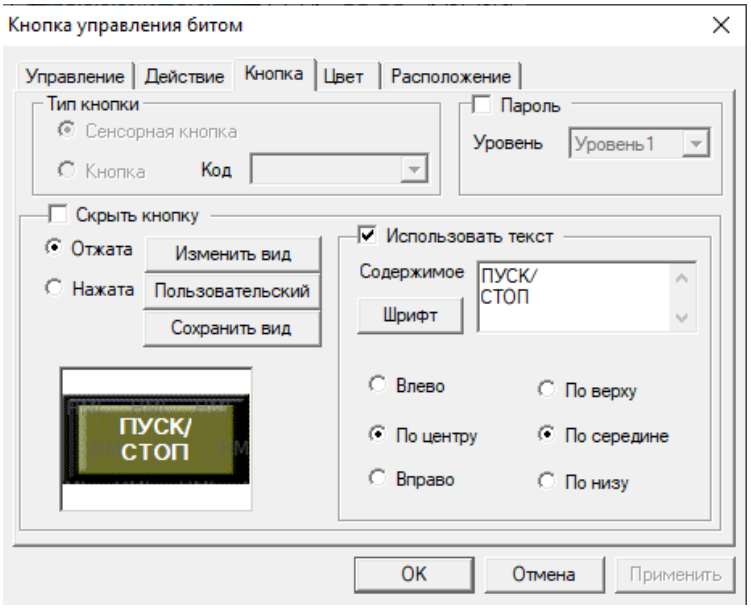


Рисунок 78 – Кнопка «ПУСК/СТОП»

Затем были добавлены «Переключатели с подсветкой», аналогичным образом, как и кнопка «ПУСК/СТОП», связанные с переменными «НАЧАЛЬНОЕ ПОЛОЖЕНИЕ», «КОНЕЧНОЕ ПОЛОЖЕНИЕ», «СБРОС АВАРИИ». Демонстрация настроек и внешнего вида переключателей на рисунках 79, 80 и 81

Рисунок 79 – «Переключатели с подсветкой»

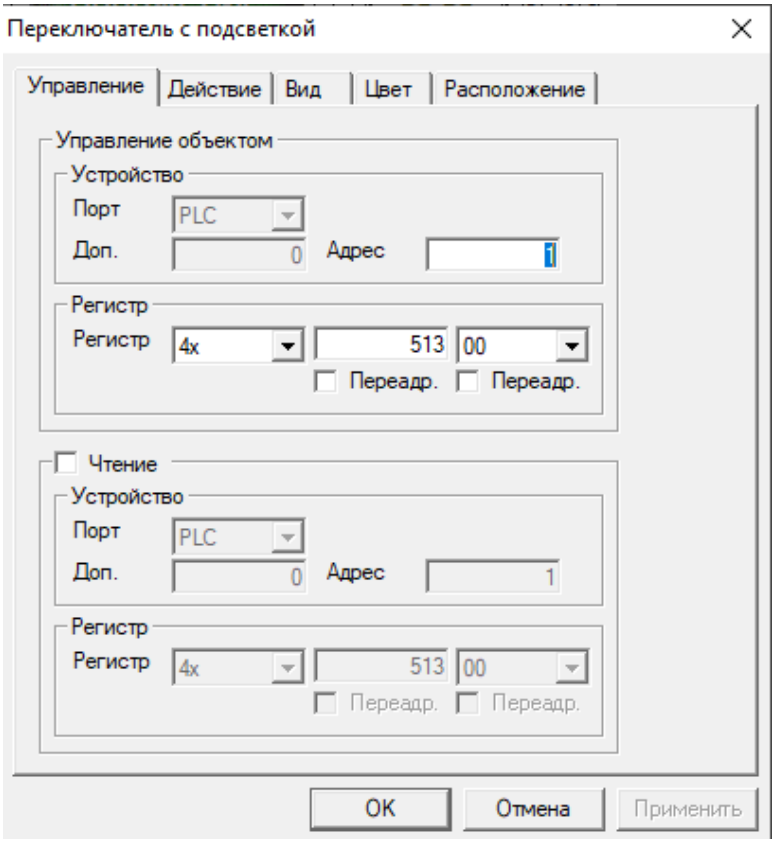


Рисунок 80 – Настройка «Переключателей с подсветкой»

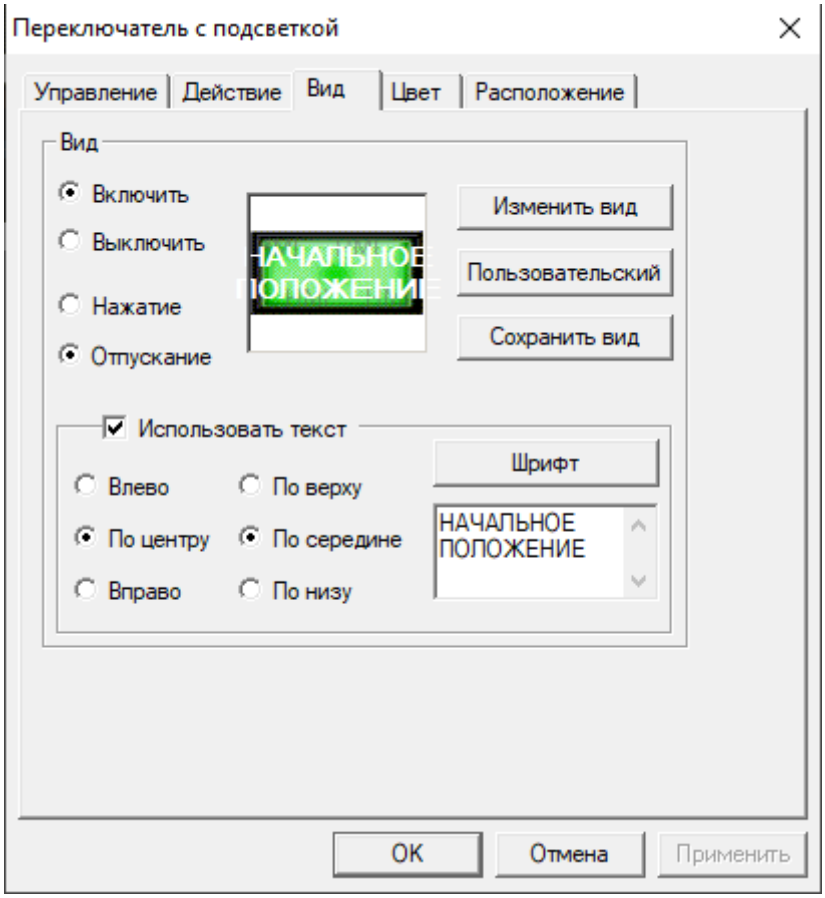


Рисунок 81 – Настройка «Переключателей с подсветкой»

Все настройки «Переключателей с подсветкой» аналогичны друг другу, кроме адресов регистров.

В итоге получается панель, демонстрируемая на рисунке 82.

Рисунок 82 – Часть панели с кнопками и выключателями

Также был добавлен «Цифровой дисплей», связанный с переменной «ВЫХ.ЧАСТ» адресом регистра, чтобы отобразить на панели оператора рабочую частоту в момент времени. Демонстрация настройки и самого дисплея на рисунках 83 и 84.

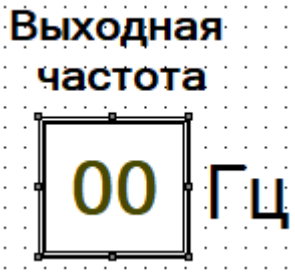


Рисунок 83 – «Цифровой дисплей» с переменной «ВЫХ.ЧАСТ»

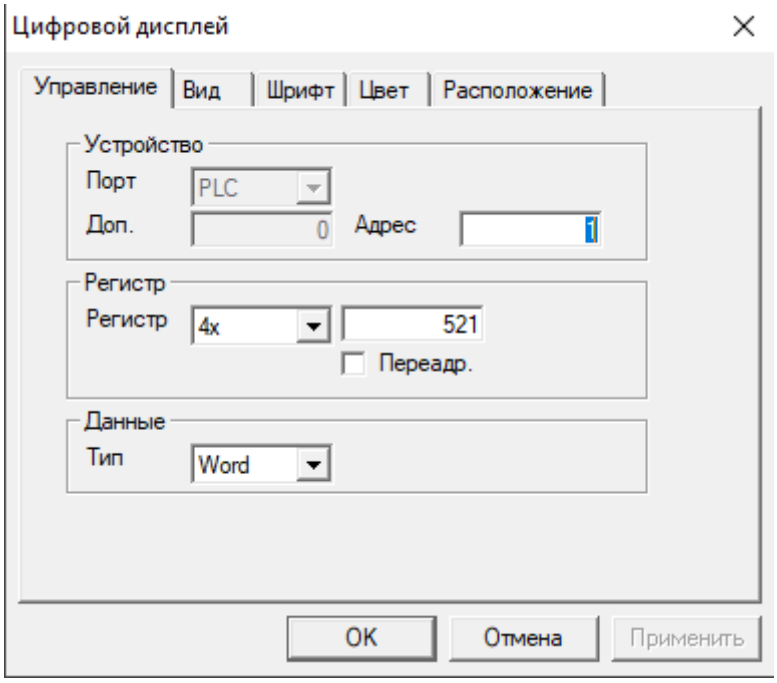


Рисунок 84 – Настройки «Цифрового дисплея» с переменной «ВЫХ.ЧАСТ»

Окончив работу с первым экраном панели, был добавлен еще один, содержащий элементы «Ввод данных» для переменных «КОН.ЧАСТ.», «НАЧ.ЧАСТ.», «ВРЕМЯ.ЗАД.». Данный элемент позволяет вводить определенные значения в связанные с ним при помощи адресов регистров переменные. Демонстрация внешнего вида и настроек на рисунках 85 и 86.

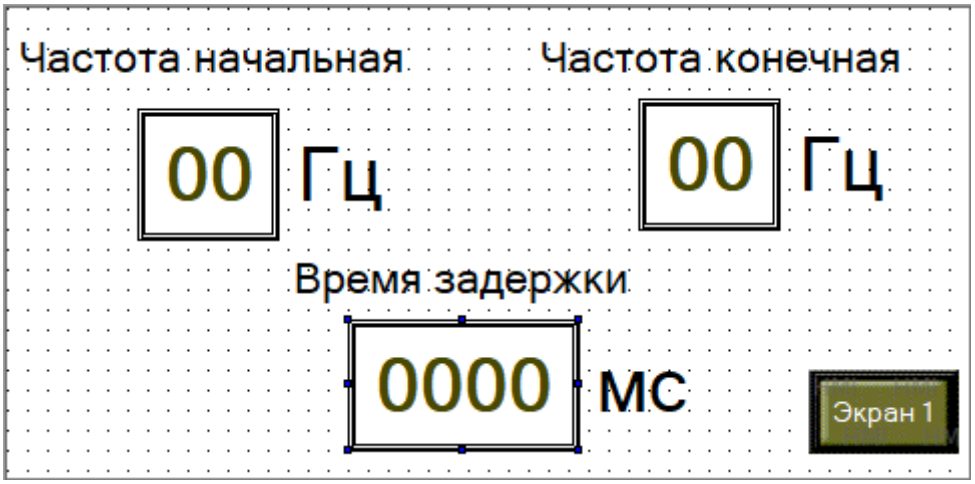


Рисунок 85 – Экран с элементами «Ввод данных» для переменных «КОН.ЧАСТ.», «НАЧ.ЧАСТ.», «ВРЕМЯ.ЗАД.»

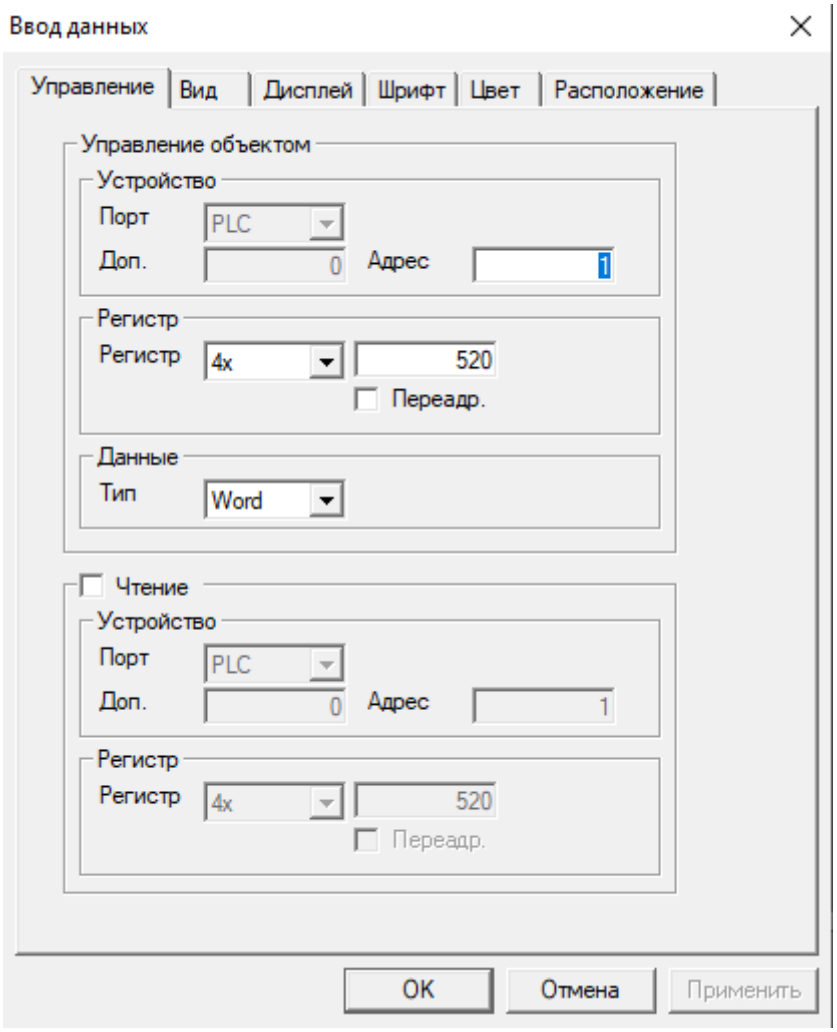


Рисунок 86 – Настройка управления «Ввод данных» для переменных «КОН.ЧАСТ.», «НАЧ.ЧАСТ.», «ВРЕМЯ.ЗАД.»

Настройка управления у всех трех переменных одинаковая, за исключением адресов регистров. Также отличаются настройки во вкладке Вид, как продемонстрировано на рисунках 87, 88.

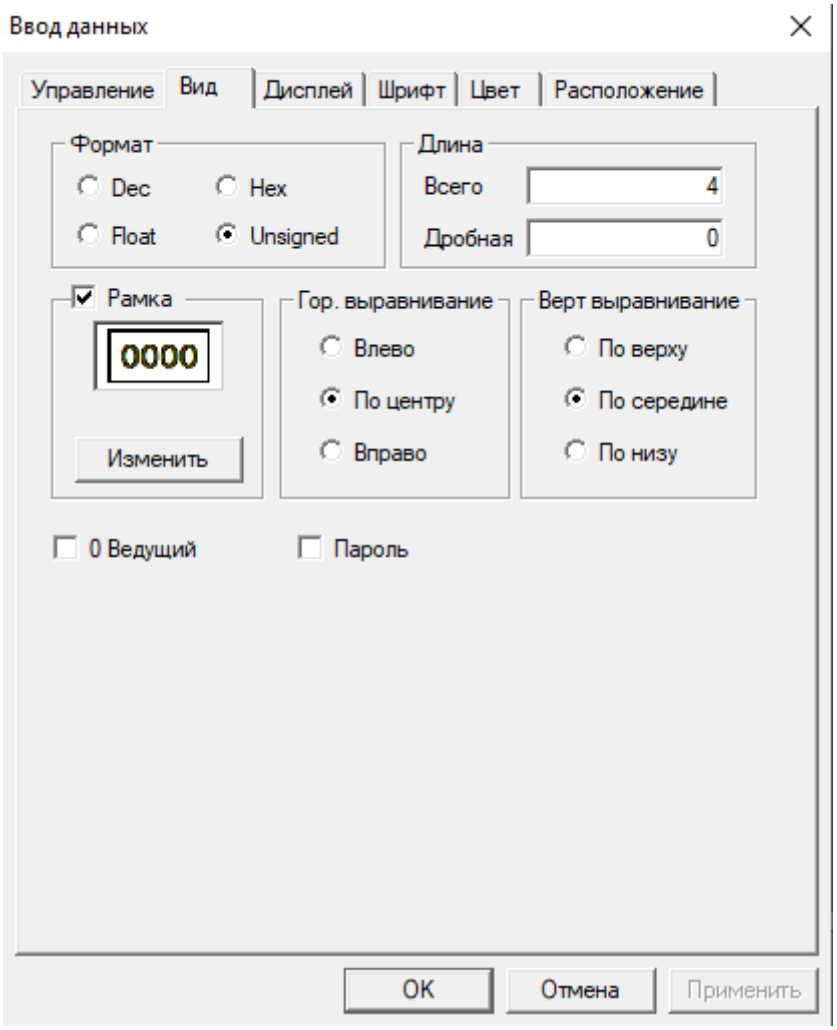


Рисунок 87 – Настройка ввода для переменной «ВРЕМЯ.ЗАД.»

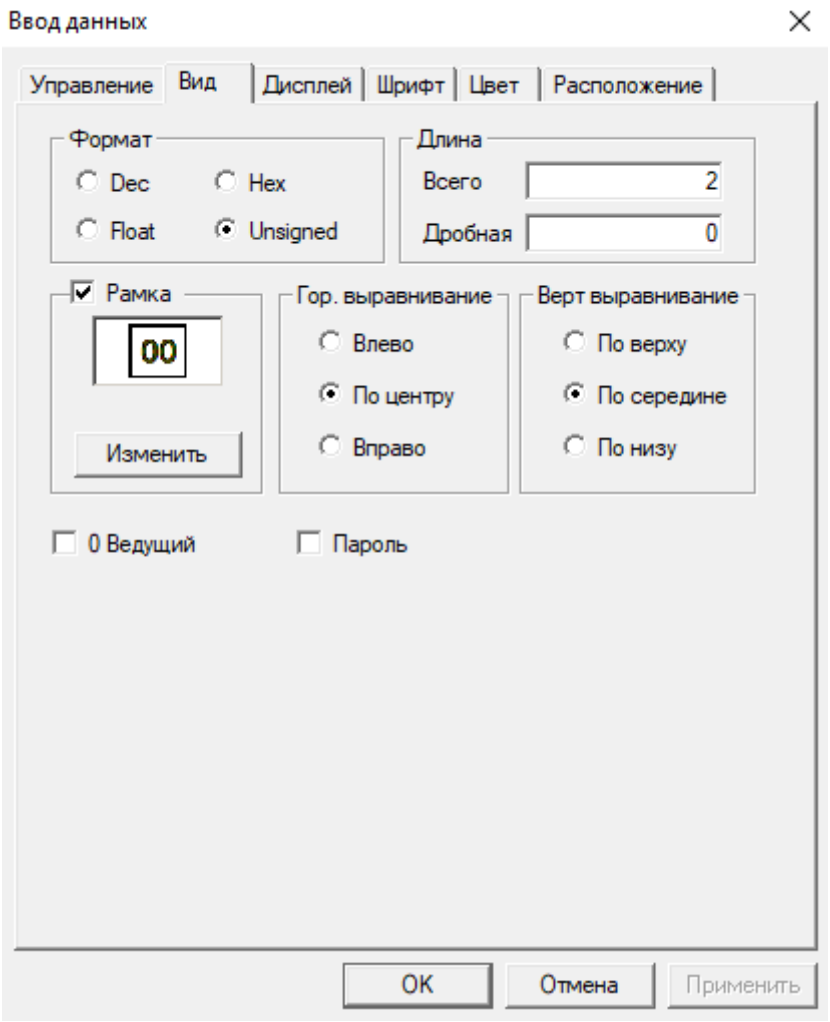


Рисунок 88 – Настройка ввода для переменных КОН.ЧАСТ.» и «НАЧ.ЧАСТ.»

В самом конце получается итоговая панель управления с двумя экранами, между которыми оператор может перемещаться, нажимая соответствующие кнопки «Переход на экран», как показано на рисунке 89.

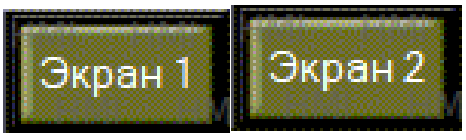


Рисунок 89 – Переходы между экранами

После выполнения всех вышеописанных действий, получалась панель оператора в двух экранах, как показано на рисунках 90 и 91.

Рисунок 90 – Итоговый экран 1 панели оператора

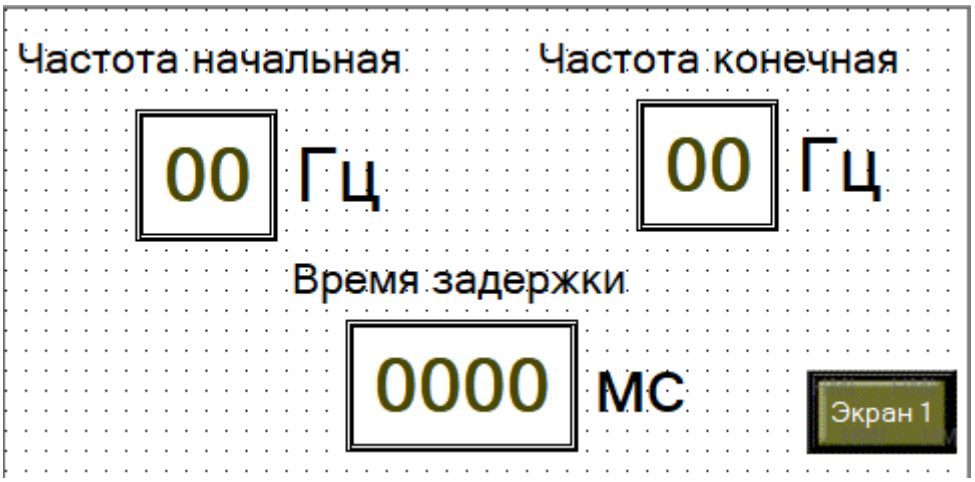


Рисунок 91 – Итоговый экран 2 панели оператора

1.3 Анализ результатов

1.3.1 Тестирование и отладка проекта в среде OWEN Logic

Работа управляющей программы, разработанной в соответствии с заданием, была протестирована в симуляторе OWEN Logic. Результаты тестирования, подтверждающие ее корректность, представлены на рисунках 92-95.

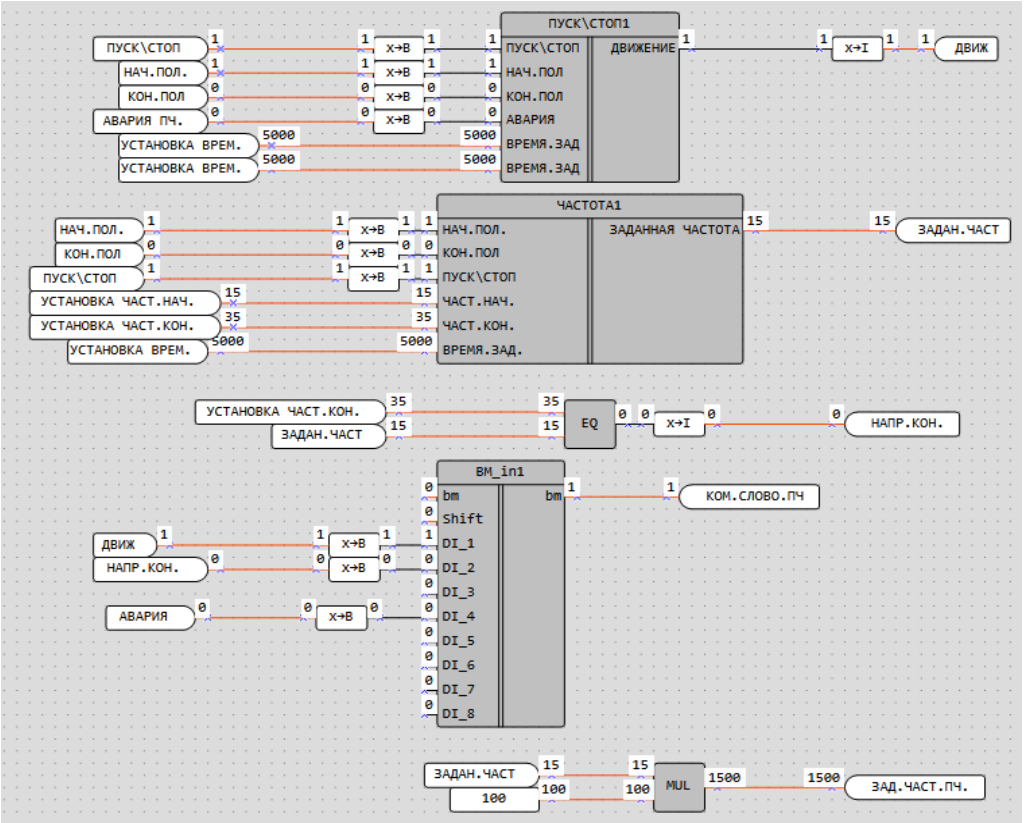


Рисунок 92 – Тестирование в OWEN Logic

Перемещение суппорта на заданной рабочей скорости в 15 герц начинается по кратковременному нажатию кнопки «ПУСК/СТОП» от «НАЧ.ПОЛ.».

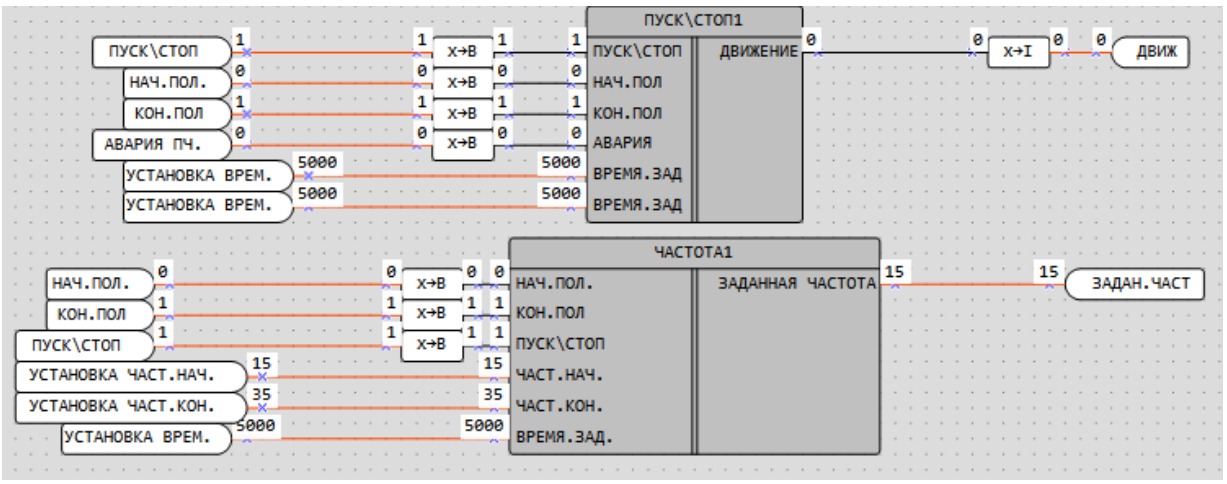


Рисунок 93 – Тестирование в OWEN Logic

Остановка по срабатыванию концевого выключателя на 5 секунд.

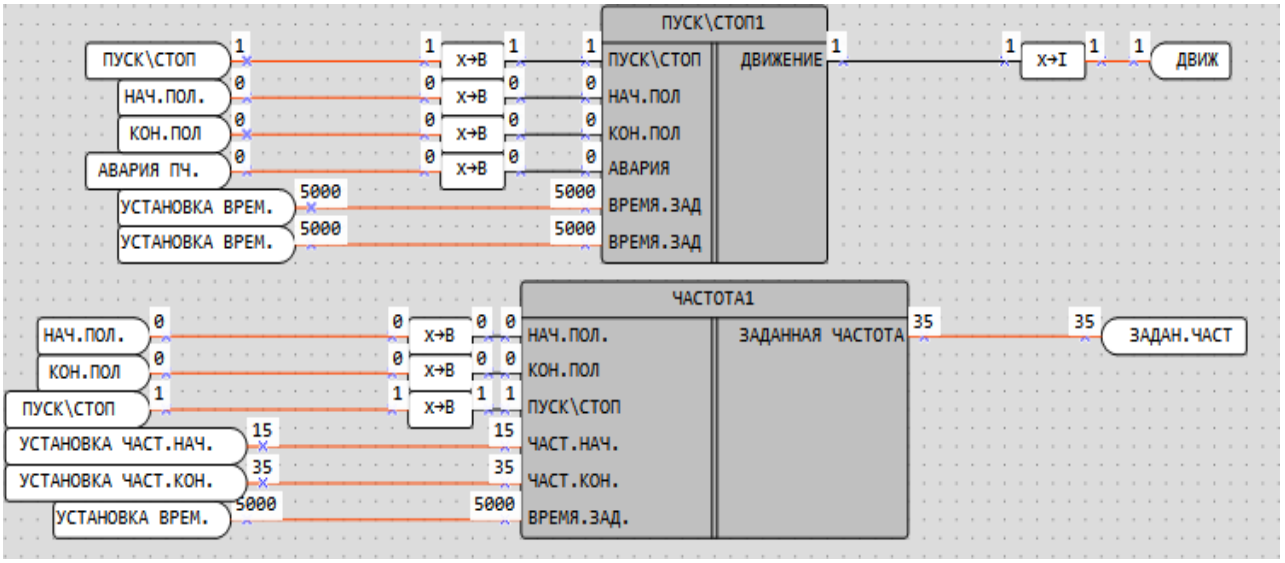


Рисунок 94 – Тестирование в OWEN Logic

Движение на повышенной скорости 35 герц в обратную сторону к «НАЧ.ПОЛ.».

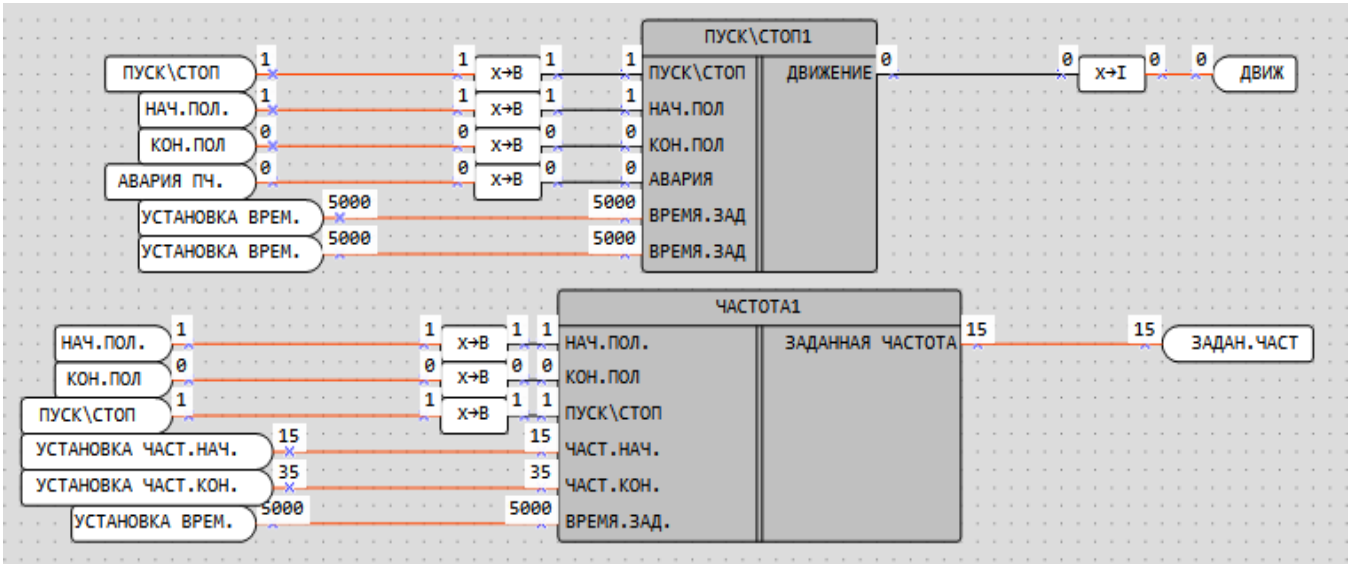


Рисунок 95 – Тестирование в OWEN Logic

Достижение «НАЧ.ПОЛ.», снижение скорости до 15 герц и остановка двигателя.

Также были успешно были протестированы кнопка «АВАРИЯ» и остановка двигателя по нажатию кнопки «ПУСК/СТОП».

1.3.2 Тестирование на физическом оборудовании

Аналогичные действия по тестированию были проведены на физическом оборудовании, которое включает в себя: Панель оператора ОВЕН СП270, ПЛК ОВЕН ПР200, ПЧ ВЕСПЕР Е2-8300. Демонстрация на рисунках 96 – 100.

Рисунок 96 – Начальные значения тестирования

Рисунок 97 – Движение от «НАЧ.ПОЛ.» к «КОН.ПОЛ.»

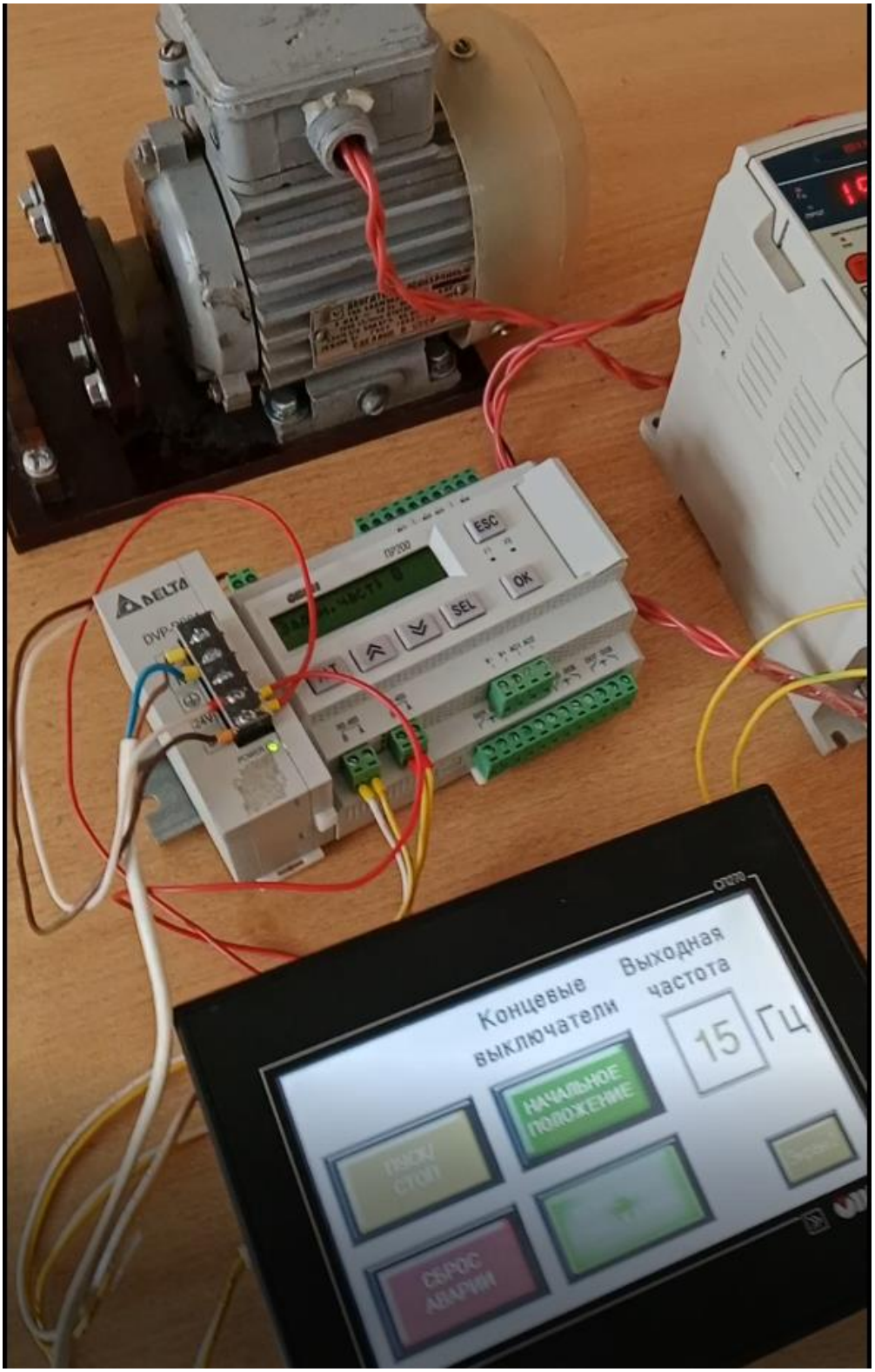


Рисунок 98 – «КОН.ПОЛ.» и остановка двигателя на 5 секунд

Рисунок 99 – Реверс в обратную сторону на повышенной скорости

Рисунок 100 – Достижение «НАЧ.ПОЛ.» и остановка

Результаты тестирования соответствуют требованиями проекта.

По результат выполненных работ по проекту были составлены несколько UML – диаграмм: UML – диаграмма компонентов на рисунке 101, UML – диаграмма последовательностей на рисунке 102, UML – диаграмма развертывания на рисунке 103, UML – диаграмма деятельности на рисунке 104.

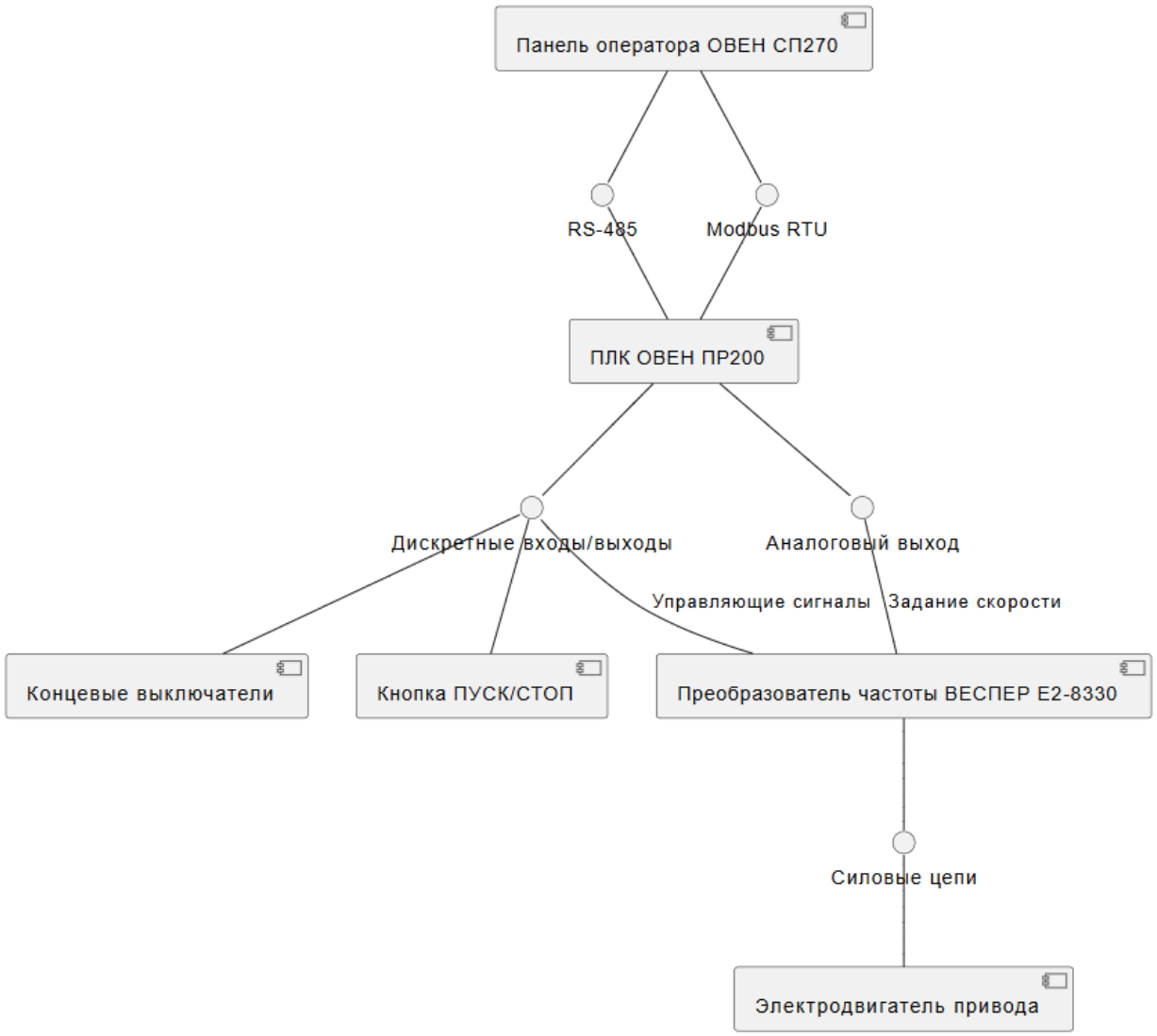


Рисунок 101 – UML – диаграмма компонентов

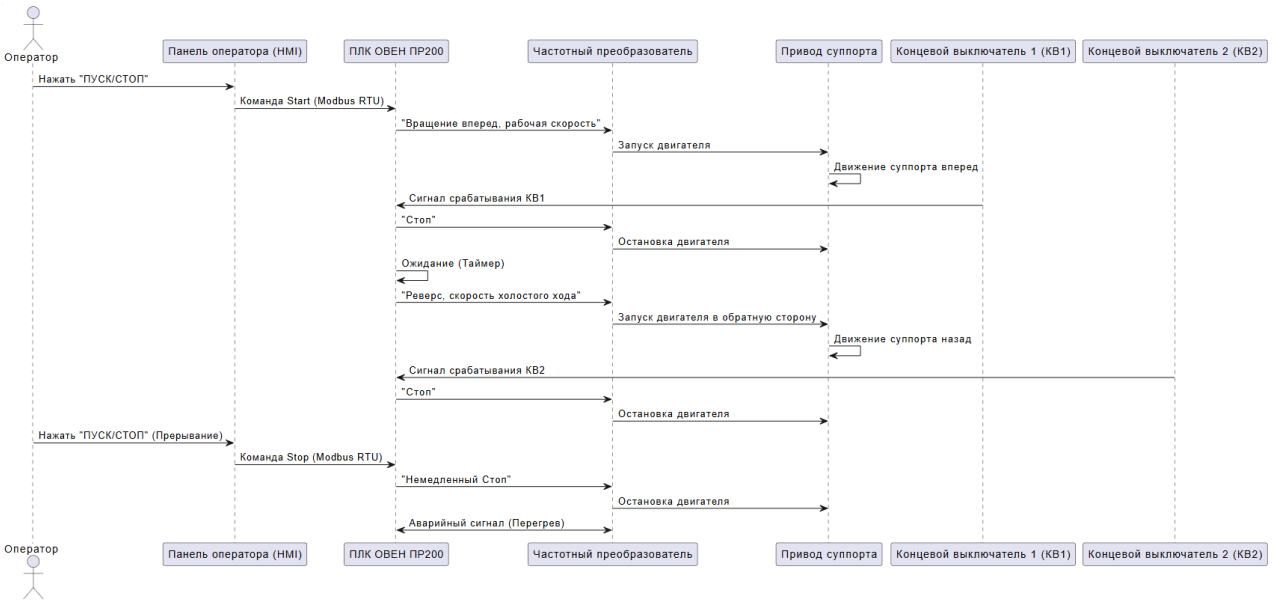


Рисунок 102 – UML – диаграмма последовательностей

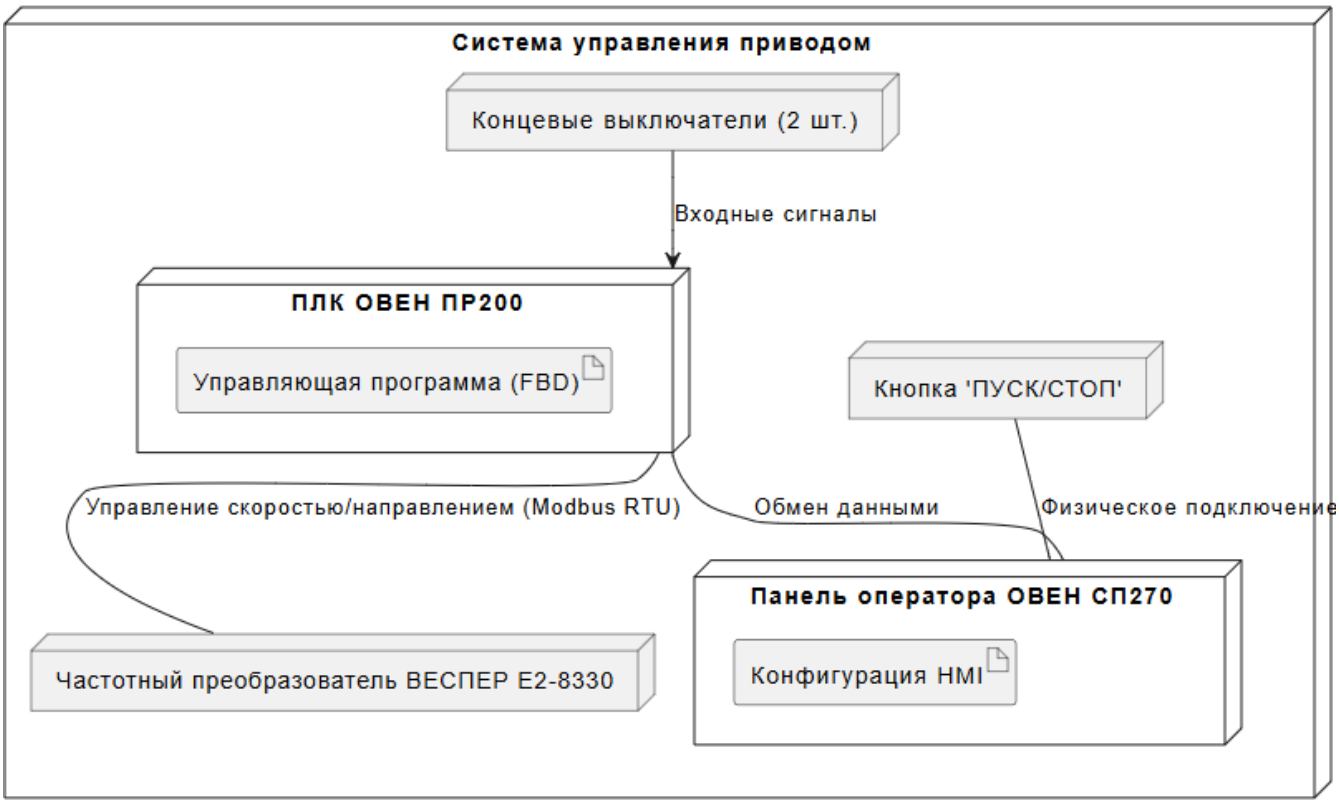


Рисунок 103 – UML – диаграмма развертывания

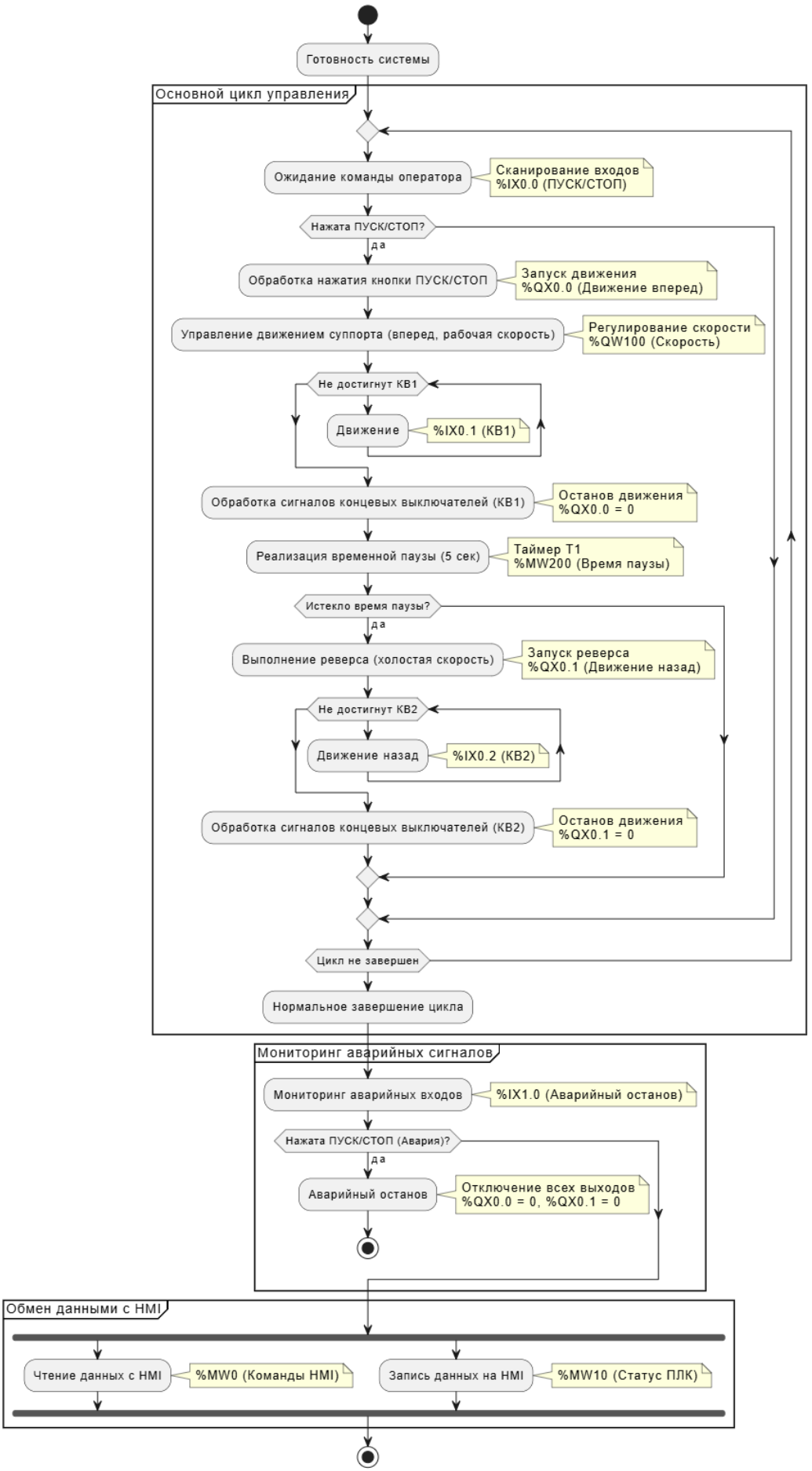


Рисунок 104 – UML – диаграмма деятельности

2 Охрана труда

Настоящий раздел регламентирует требования к организации помещений и рабочих мест, оборудованных компьютерной и офисной техникой, в соответствии с действующей нормативной базой. В частности, рассматриваются вопросы обеспечения эргономичности, надлежащего освещения, электробезопасности и оптимальных параметров микроклимата в целях создания комфортных и безопасных условий труда.

2.1 Требования к организации рабочих мест пользователей ПЭВМ

Работа за компьютером сопряжена с воздействием нескольких неблагоприятных факторов, таких как статическое напряжение мышц, зрительное утомление, электромагнитное излучение и недостаток физической активности. Чтобы снизить их негативное воздействие, следует придерживаться следующих рекомендаций:

– для поддержания правильной осанки необходимо держать спину прямо, предплечья располагать на поверхности стола, а ноги – на полу или специальной подставке, формируя угол в 90 градусов в коленях;

– для комфортной работы и здоровья глаз, установите монитор на расстоянии 50-70 см от лица; верхний край экрана должен располагаться на уровне глаз или чуть ниже, чтобы избежать напряжения шеи [6];

– для поддержания здоровья и продуктивности рекомендуется делать короткие перерывы (5-10 минут) каждые 45-60 минут работы; в это время следует выполнять разминку и упражнения для глаз [5];

– необходимо избегать сгибания или разгибания запястий [7].

2.1.1 Оборудование рабочего места

Рабочее место должно соответствовать следующим требованиям:

– кресло с возможностью индивидуальной настройки: регулируется высота, наклон спинки и глубина сиденья; оснащено подлокотниками и

поясничной поддержкой для комфорта;

– оптимальное расстояние от глаз до монитора составляет 600-700 мм; угол наклона экрана должен быть в пределах 10-15 градусов ниже горизонтальной линии, проходящей через глаза; необходимо исключить попадание прямых солнечных лучей или отражений от других источников света на экран, используя антибликовые покрытия или изменяя положение монитора;

– стол должен иметь регулируемую высоту в диапазоне от 680 до 800 мм; если регулировка не предусмотрена, высота стола должна быть 725 мм; размер столешницы: глубина не менее 800 мм, ширина не менее 1200 мм, чтобы удобно разместить монитор, клавиатуру и рабочие документы; под столешницей должно быть достаточно места для ног: высота не менее 600 мм, глубина 450 мм на уровне колен и 650 мм для вытянутых ног.

Правильная посадка за ПК демонстрируется на рисунке 105

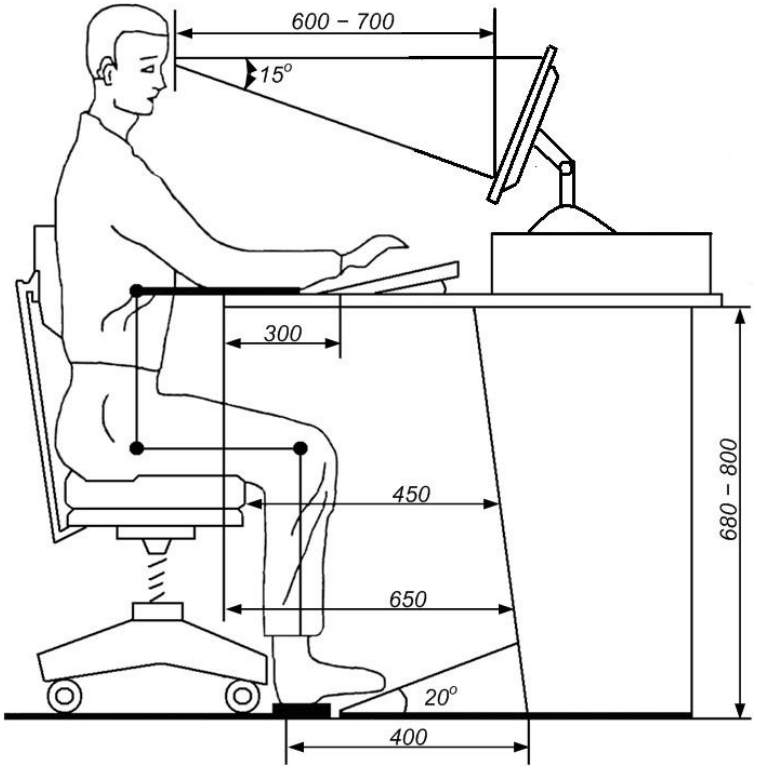


Рисунок 105 – Правильная посадка за ПК

2.1.2 Освещенность рабочего места

Нормы освещенности регламентируются СП 52.13330.2016 "Естественное и искусственное освещение" [8]:

– уровень общего освещения должен составлять не менее 300–500 люкс;

– комбинированное освещение (естественное и искусственное) должно обеспечивать достаточную освещенность – не менее 400 люкс;

– разместите настольную лампу таким образом, чтобы свет от нее не создавал отражений и бликов на экране монитора.

Дополнительные рекомендации включают:

– ориентация светильников должна совпадать с вектором направления взгляда оператора;

– недопустимы резкие колебания яркости в пределах видимости;

2.1.3 Электробезопасность

Требования к электробезопасности регламентируются ГОСТ 12.1.019 – 2017 [10]:

– все электроприборы должны быть подключены к сети электропитания посредством заземленных розеток европейского типа (тип F);

– запрещается эксплуатация электросети с использованием удлинителей, не оборудованных устройствами защиты от перегрузки;

– провода должны иметь надлежащую изоляцию и располагаться таким образом, чтобы не препятствовать свободному перемещению;

2.1.4 Микроклимат и шум

Согласно СанПиН 1.2.3685-21 [11]:

– диапазон температур воздуха составляет 22-24°C в зимний период и 23-25°C в летний период;

– влажность находится в пределах от 40 до 60 процентов;

– уровень шума не должен превышать 50 дБ в офисных помещениях;

Заключение

В рамках данного дипломного проекта была успешно разработана программа управления приводом продольного перемещения суппорта токарного станка на базе ПЛК ОВЕН ПР200 и панели оператора ОВЕН СП270 с использованием частотного преобразователя ВЕСПЕР Е2–8330. Основной целью работы являлось создание автоматизированной системы, обеспечивающей возвратно-поступательное движение суппорта с возможностью управления через интуитивный интерфейс оператора.

Разработка велась в среде программирования OWEN Logic с применением языка функциональных блоков FBD, что позволило реализовать четкую логику управления и обеспечить удобство отладки. Программа предусматривает запуск и остановку привода по сигналу кнопки «ПУСК/СТОП», а также автоматическую остановку при срабатывании концевых выключателей, что гарантирует безопасность эксплуатации. Управление параметрами движения, такими как скорость и направление, осуществляется через графический интерфейс панели оператора, что значительно упрощает взаимодействие пользователя с системой.

В результате проведенной работы была создана надежная и функциональная система, соответствующая всем поставленным требованиям. Разработанное решение демонстрирует эффективность применения современных промышленных контроллеров и частотных преобразователей для автоматизации металлорежущего оборудования. Полученные результаты могут быть использованы для дальнейшего усовершенствования системы, включая интеграцию дополнительных датчиков и расширение функциональных возможностей управления.

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. 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 (дата обращения: 22.05.2025).
3. Панель оператора СП270. Паспорт и руководство по эксплуатации КУВФ.421449.003 РЭ. ООО «Производственное Объединение ОВЕН». [Электронный ресурс] / Режим доступа: https:// свободный (дата обращения: 22.05.2025).
4. Общие сведения о языке FBD. [Электронный ресурс] / Режим доступа: https://doc.ioot.pro/usage-guide/iec_guide/fbd_guide.html (дата обращения: 22.05.2025).
5. СанПиН 1.2.3685-21 "Гигиенические нормативы факторов рабочей среды" [Электронный ресурс] / Режим доступа: https://assistentus.ru/forma/prikaz-o-provedenii-sanitarnogo-dnya/?ysclid=mbe9l7vdyt603855887 (дата обращения: 22.05.2025)
6. ГОСТ Р 50923-96 "Дисплеи. Рабочее место оператора" [Электронный ресурс] / Режим доступа: https://docs.cntd.ru/document/1200025975?ysclid=mbe9oup0d9773443776 (дата обращения: 22.05.2025)
7. ГОСТ 12.2.032-78 "Рабочее место при выполнении работ сидя" [Электронный ресурс] / Режим доступа: https://ivrayon.gosuslugi.ru/netcat_files/558/2277/12.2.032_78_1_.pdf?ysclid=mbe9ryx485642552930 (дата обращения: 22.05.2025)

8. СП 52.13330.2016 "Естественное и искусственное освещение" [Электронныйресурс] / Режим доступа: https://kola.rosavtodor.gov.ru/storage/app/media/kola/uploaded-files/08-16-sp-52133302016-estestvennoe-i-iskusstvennoe-osveshchenie-aktualizirovannaya-redaktsiya-snip-23-05-95-1.pdf?ysclid=mbe9sitc15675054489 (дата обращения: 22.05.2025)

9. ГОСТ 12.1.019-2017 "Электробезопасность" [Электронный ресурс] / Режим доступа: https://docs.cntd.ru/document/1200161238?ysclid=mbe9thst2k636242628 (дата обращения: 22.05.2025)

10. СанПиН 1.2.3685-21 "Гигиенические требования к микроклимату производственных помещений" [Электронный ресурс] / Режим доступа: https://promarket.shop/sanpin-1-2-3685-21-sanitarnye-pravila-i-normy-gigienicheskie-normativy-i-trebovaniya-k-obespecheniyu-bezopasnosti-i-ili-bezvrednosti-dlya-cheloveka-faktorov-sredy-obitaniya-vi-gigienicheskie-normativy-po-ustroystvu-soderzhaniyu-i-rezhimu-raboty-organizats/?ysclid=mbe9tx8k56899327183 (дата обращения: 22.05.2025)

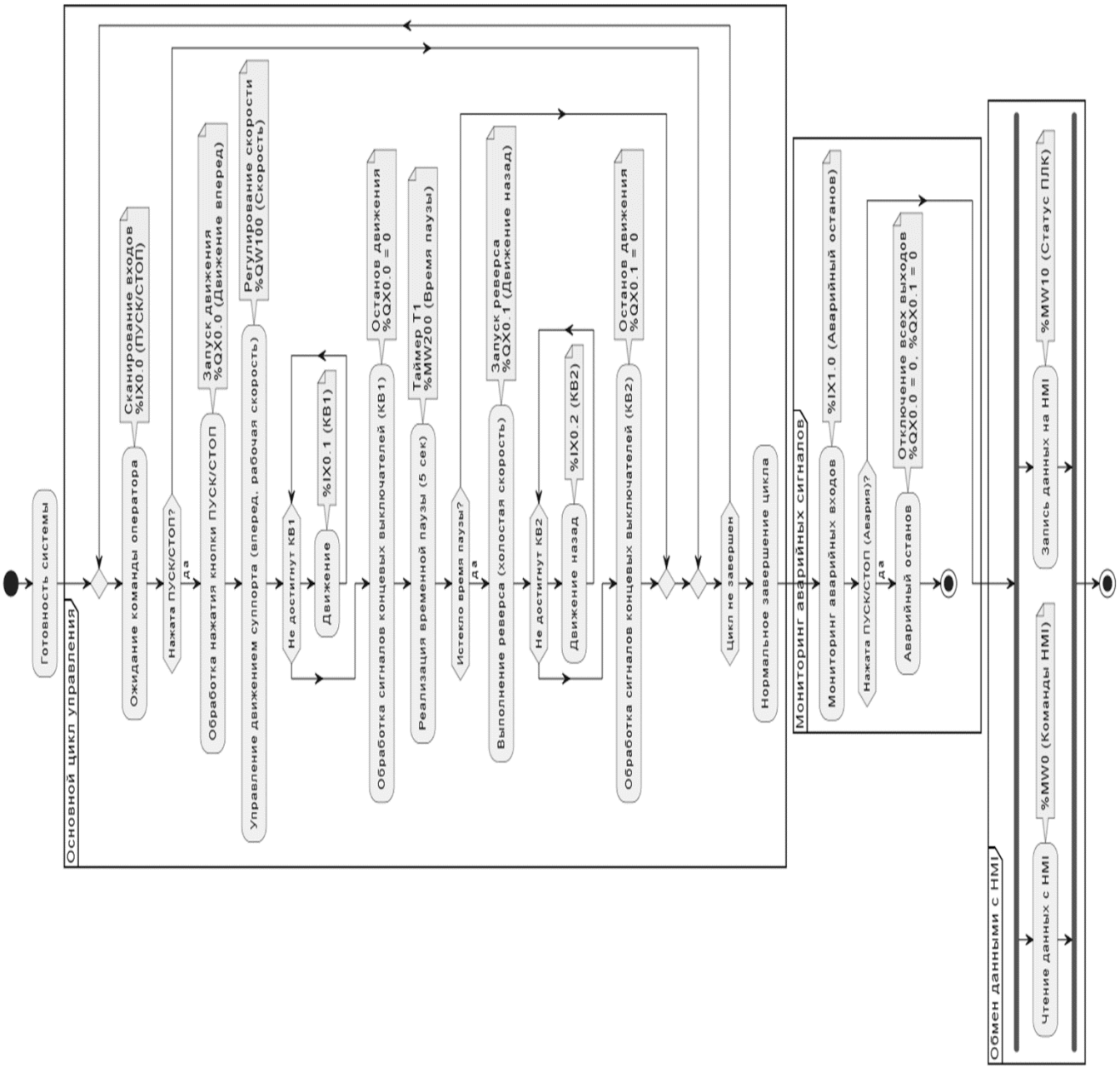
Приложение А

(Обязательное)

Программа управления приводом продольного перемещения суппорта токарного станка

UML-диаграммы

Приложение А.1 – UML – диаграмма деятельности



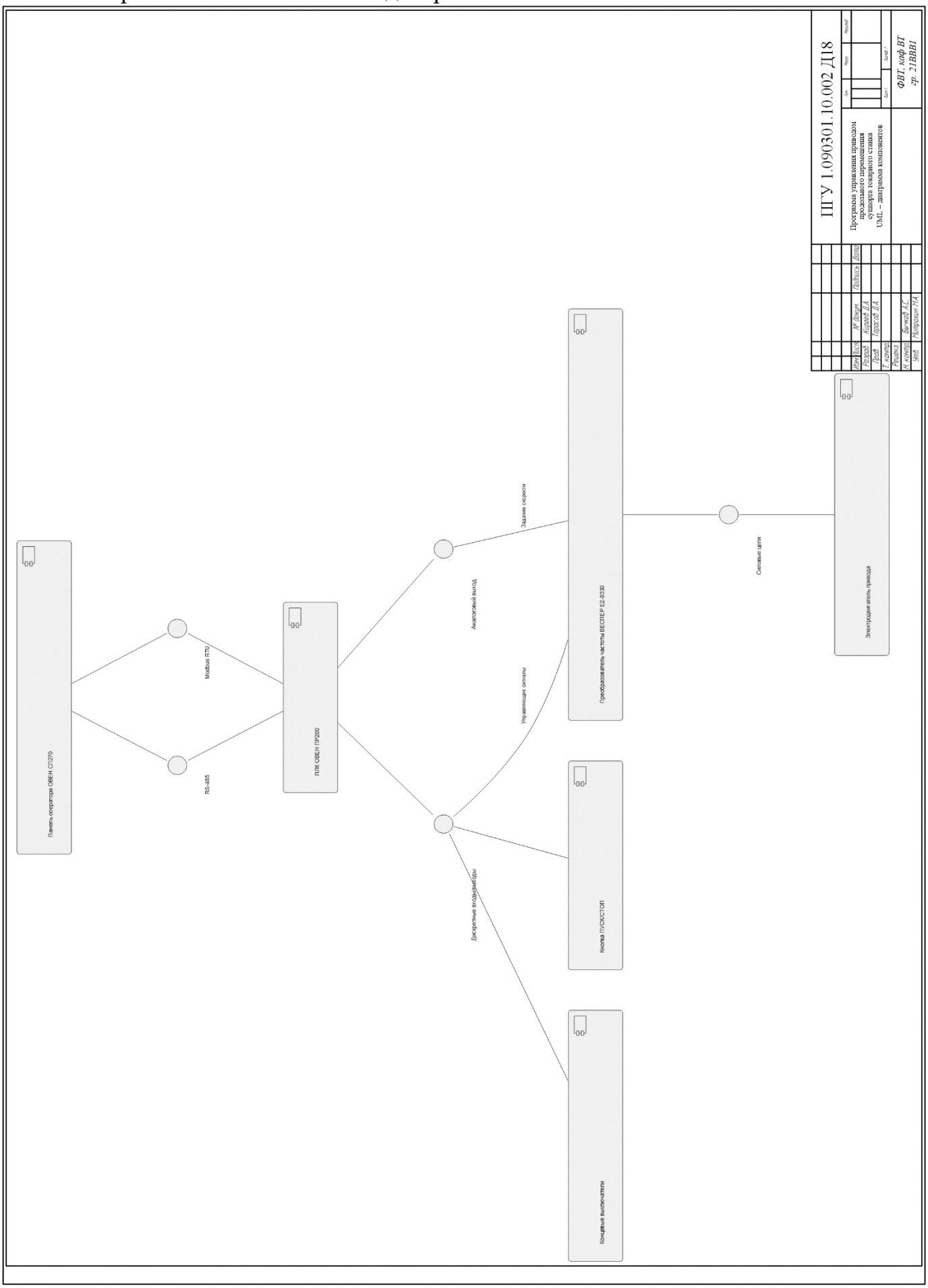
Изм.	Лист	№ докум.	Подпись	Дата	ПГУ 1.090301.10.001 ПЗ	Лист 88

ПГУ 1.090301.10.002 Д15

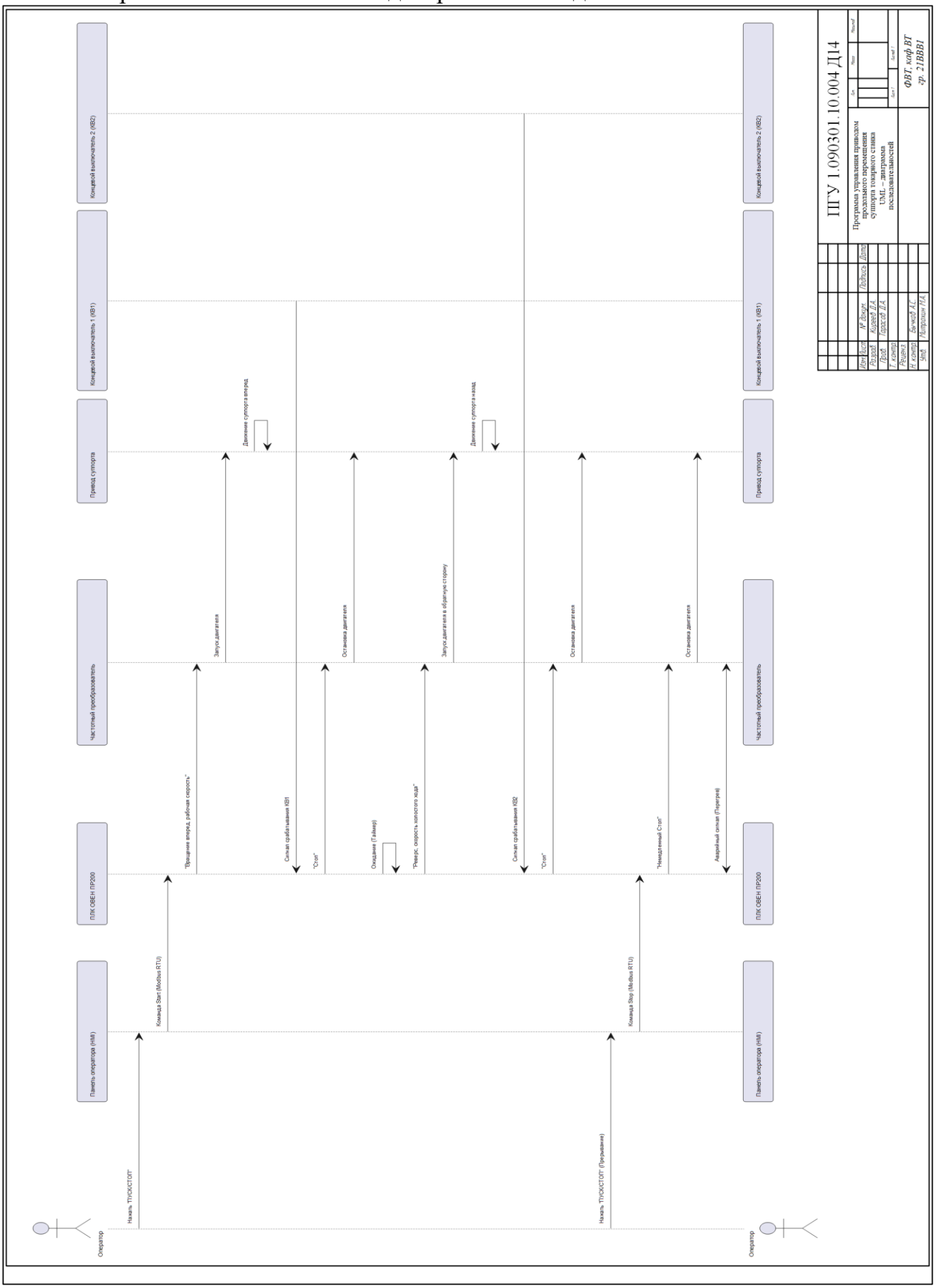
Программа управления приводом продольного перемещения суппорта токарного станка
UML – диаграмма деятельности

ФВТ, каф ВТ
гр. 21ВВВ1

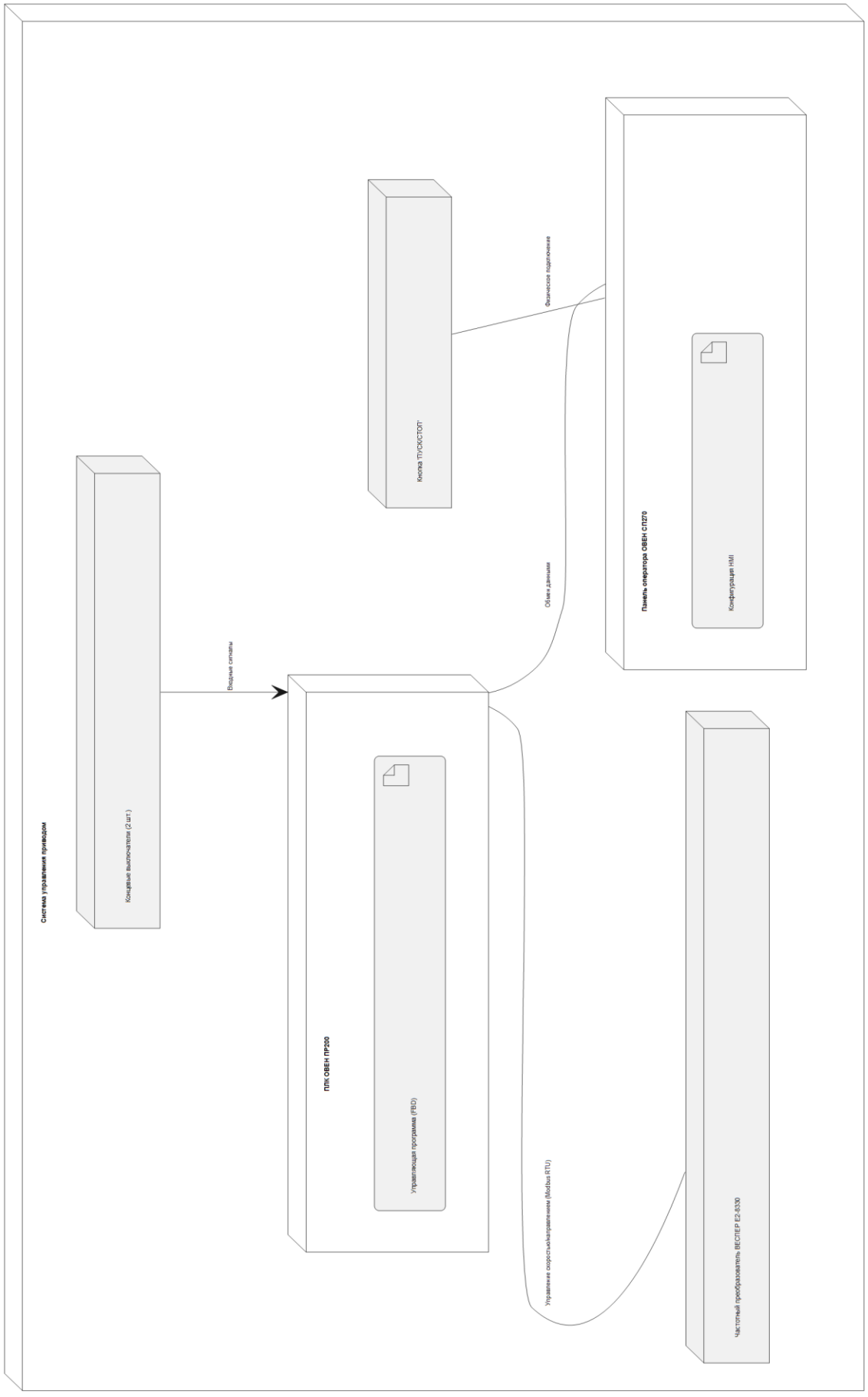
Приложение А.2 – UML – диаграмма компонентов



Приложение А.3 – UML – диаграмма последовательностей



Приложение А.4 – UML – диаграмма развертывания



ПГУ 1.090301.10.005 Д19

Программа управления приводом продольного перемещения суппорта токарного станка
UML – диаграмма развертывания

ФВТ, каф.ВТ
гр. 21ВВВ1

Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090301.10.001 ПЗ

Лист 91

Приложение Б

(Обязательное)

Программа управления приводом продольного перемещения суппорта токарного станка

Схемы функциональных блоков программ

Общая схема всех функциональных блоков проекта

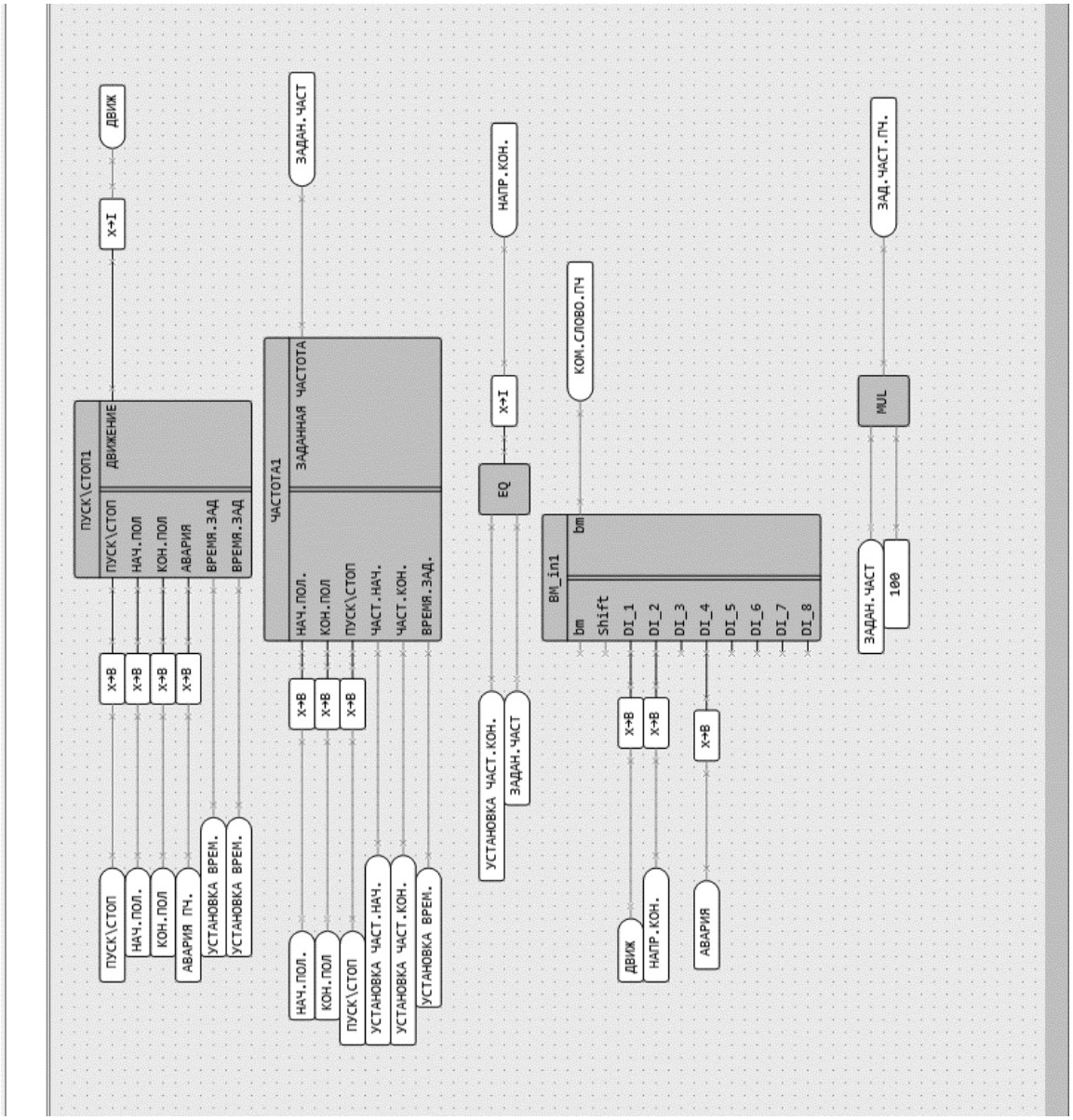


Схема функциональных блоков макроса «ПУСК\СТОП»

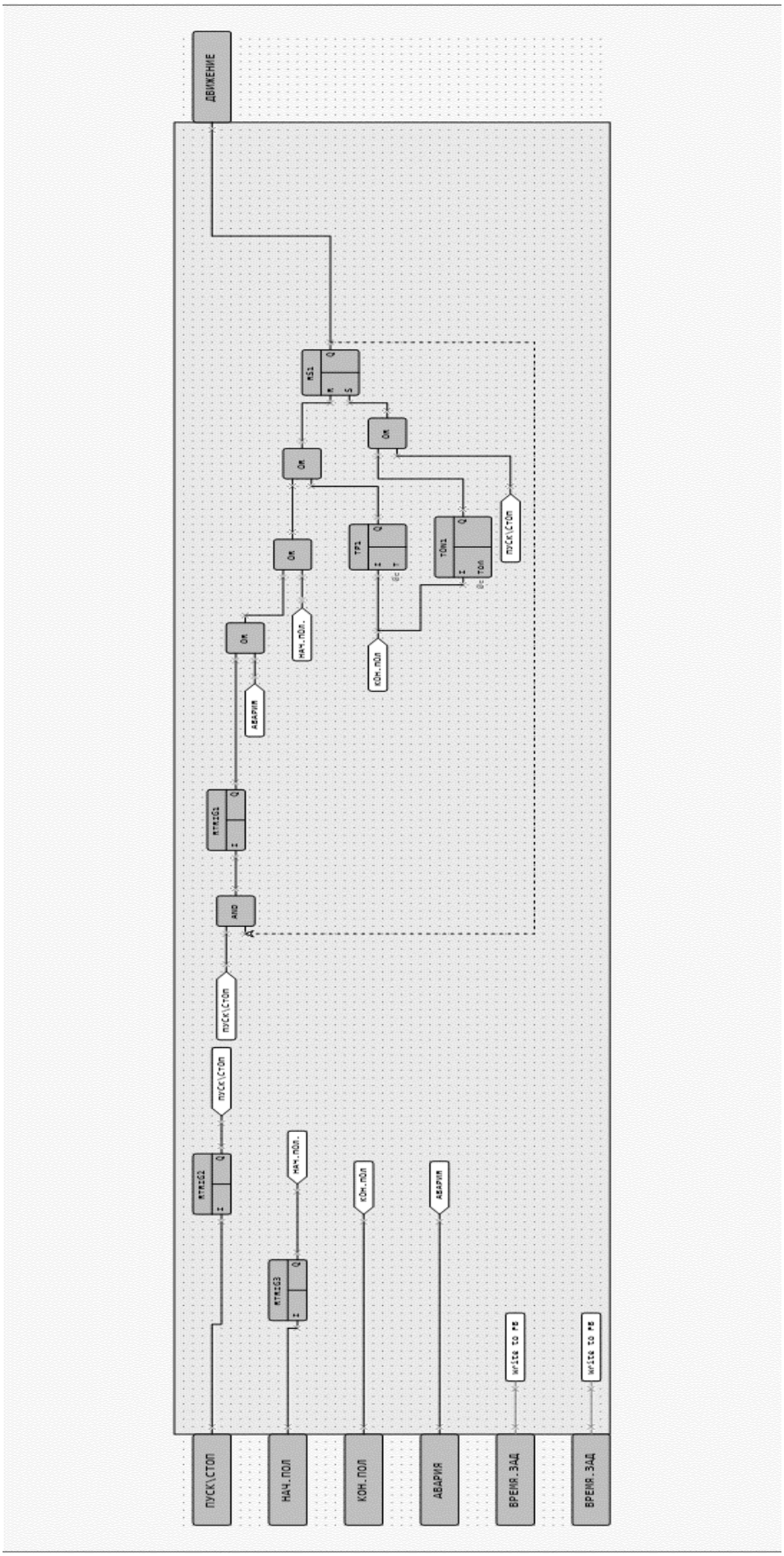


Схема функциональных блоков макроса «ЧАСТОТА»

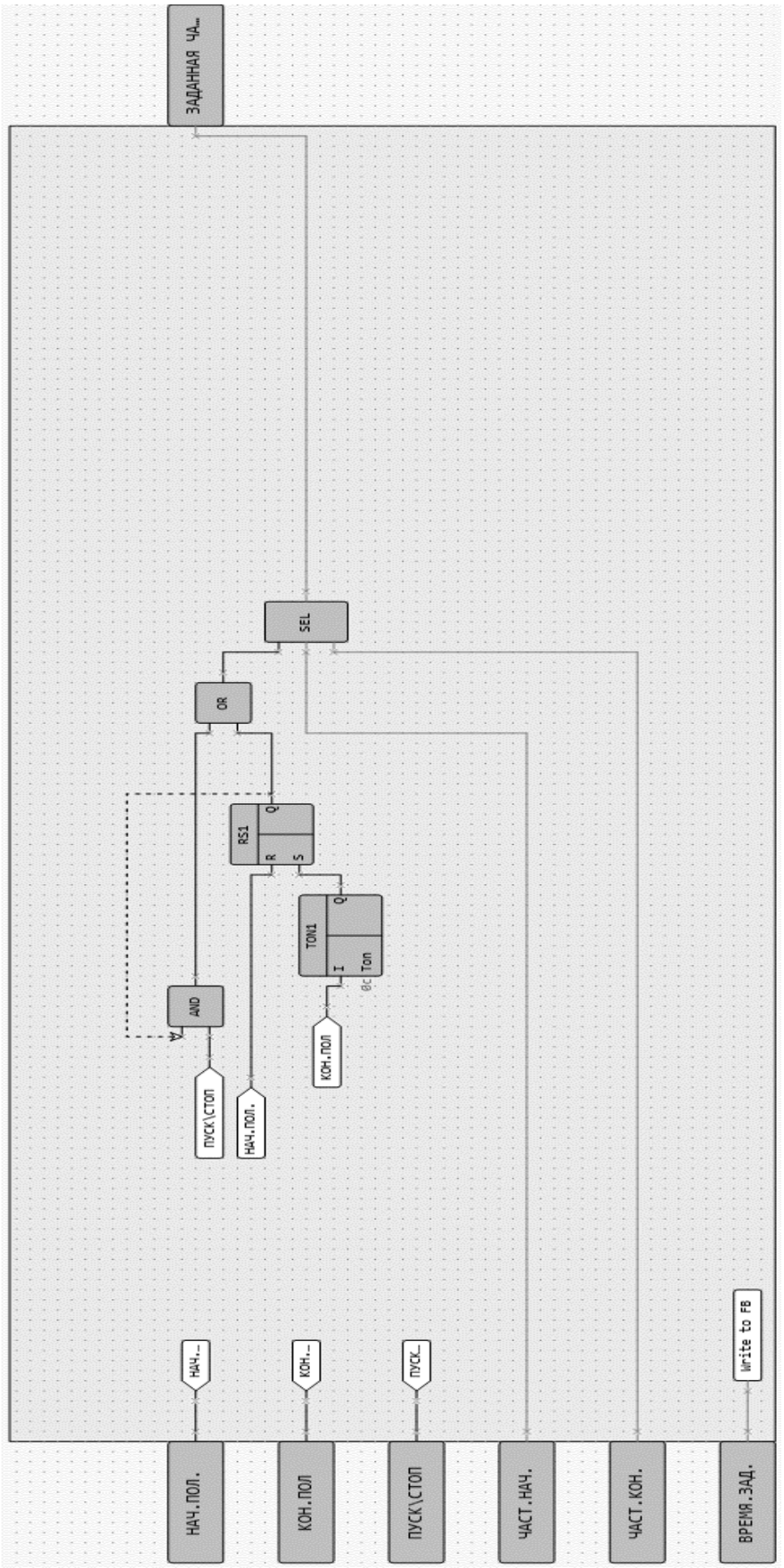


Схема определения направления

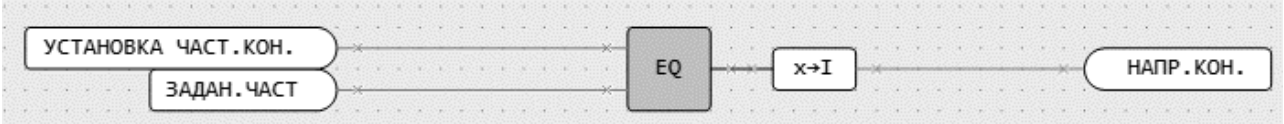
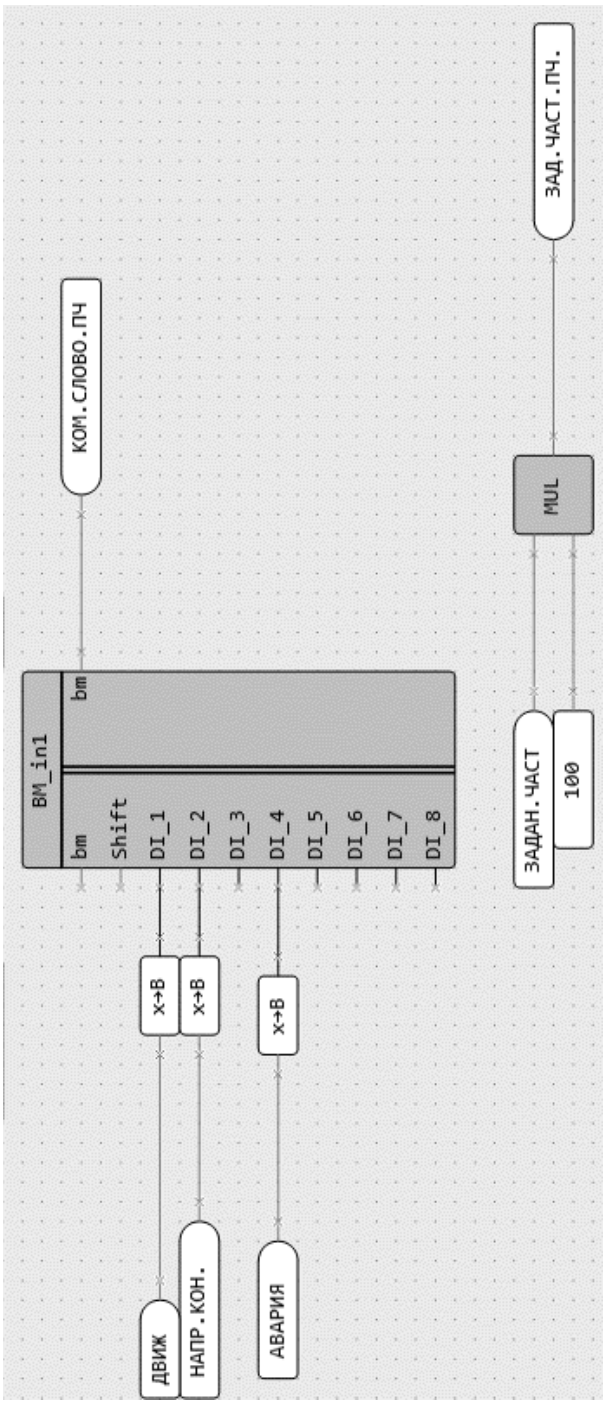


Схема битовой маски для преобразователя частоты



Приложение В

(Обязательное)

Программа управления приводом продольного перемещения суппорта токарного станка

Презентация

Программа управления приводом продольного перемещения суппорта токарного станка

Автор работы: студент 21ВВВ1 Киреев Данила Андреевич
Научный руководитель: доцент, к.т.н. Тарасов Денис Александрович

Постановка цели

Цель работы – разработать программу для контроллера, которая управляет преобразователем частоты, а тот в свою очередь приводом продольного перемещения суппорта токарного станка, который совершает возвратно-поступательное движение.

Разработка данного проекта является актуальной задачей, способствующей улучшению производственных процессов и повышению их эффективности.

Анализ предметной области

Основными пользователями данного проекта являются инженеры и операторы станков, производственные предприятия, а также разработчики оборудования, которые стремятся повысить эффективность механических процессов.

Данный проект позволяет улучшить управление и автоматизацию процессов разработки, что в итоге ведет к повышению производительности.

Siemens Sinumerik

Fanuc

Mitsubishi M70

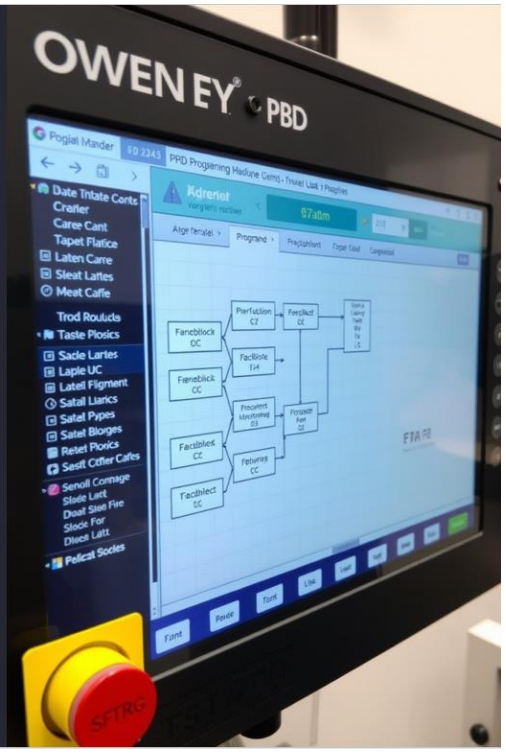
Обзор аналогов

В настоящее время существует несколько аналогичных систем управления для токарных станков

К ним относятся Siemens Sinumerik, Fanuc и Mitsubishi M70.

Требуемый функционал программы управления

– управление преобразователем частоты, который в свою очередь управляет приводом продольного перемещения суппорта токарного станка, совершая возвратно – поступательные движения;
– перемещение суппорта на заданной рабочей скорости начинается по кратковременному нажатию кнопки «ПУСК/СТОП».
– остановка по срабатыванию концевых выключателей;
– Привод стоит в течение заданного промежутка времени и быстро возвращает инструмент (реверс на заданной скорости холостого хода);
– контроль аварийных ситуаций;
–графический интерфейс на панели оператора;

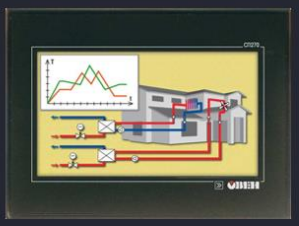


Выбор решения

Язык разработки:
Язык функциональных блоков (FBD)

Среды разработки:
OWEN Logic
Конфигуратор СП200

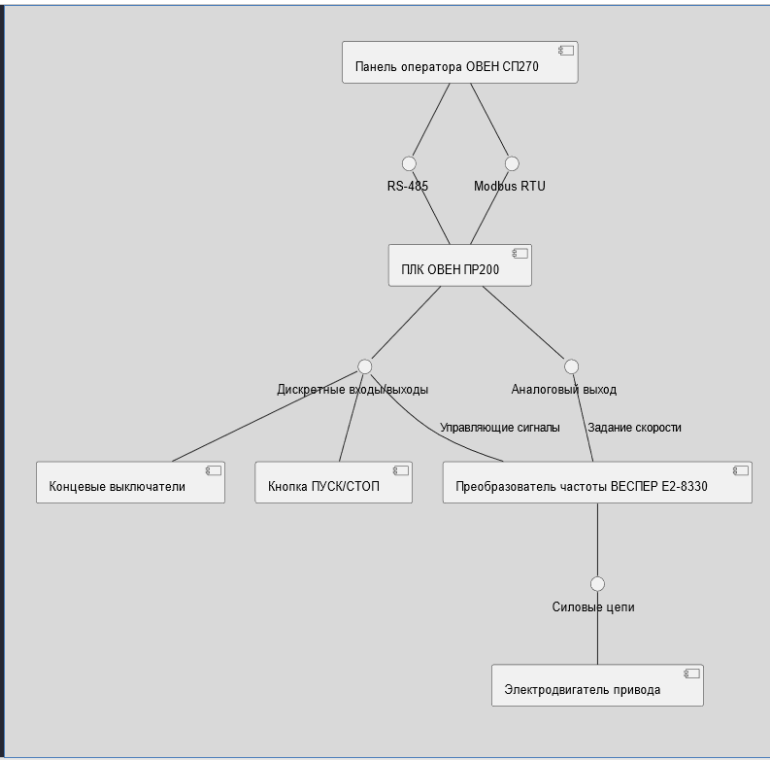
Целевая платформа:
ПЛК ОВЕН ПР200
Панель оператора ОВЕН СП270
ПЧ ВЕСПЕР Е2 - 8330



Owen Logic

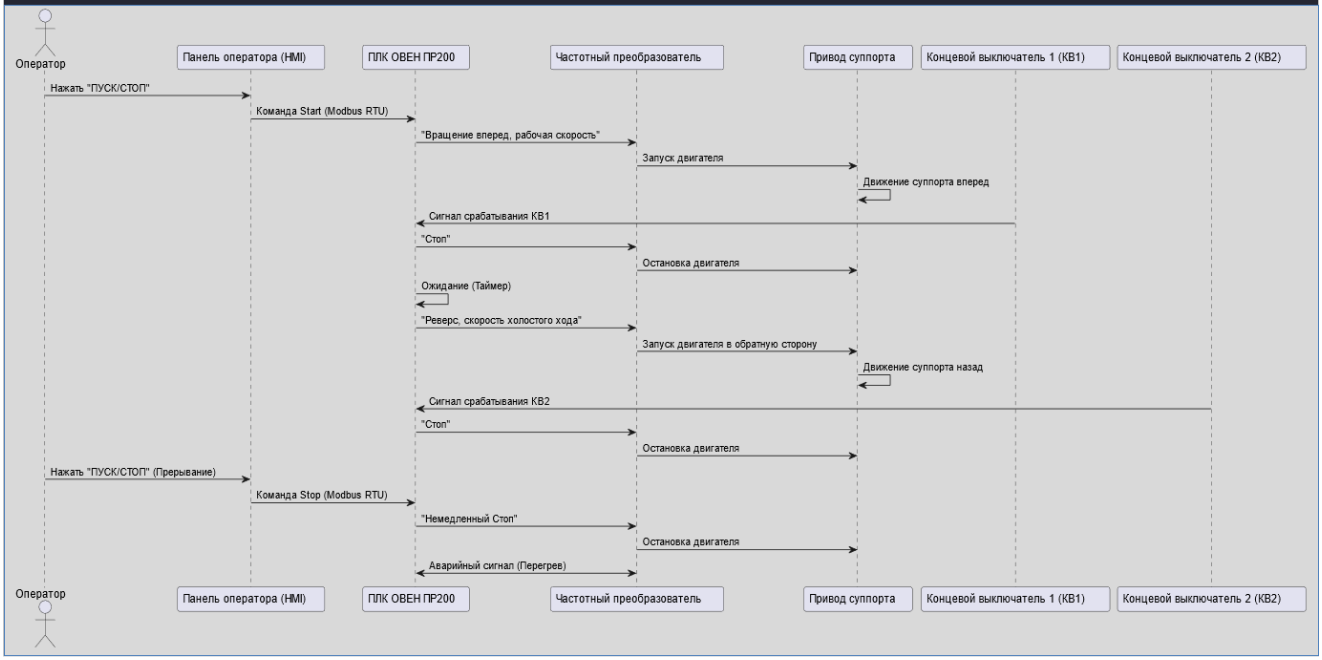
Процесс разработки

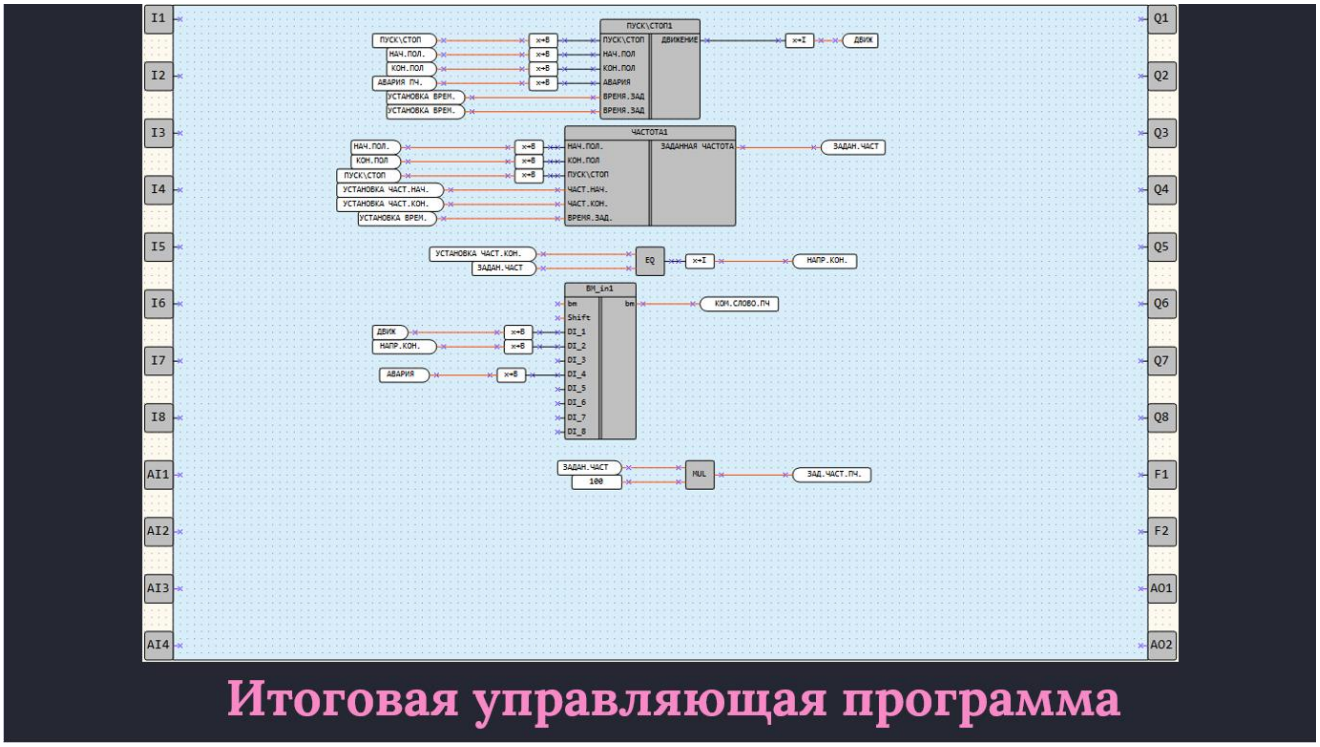
Диаграмма компонентов



Процесс разработки

Диаграмма последовательностей





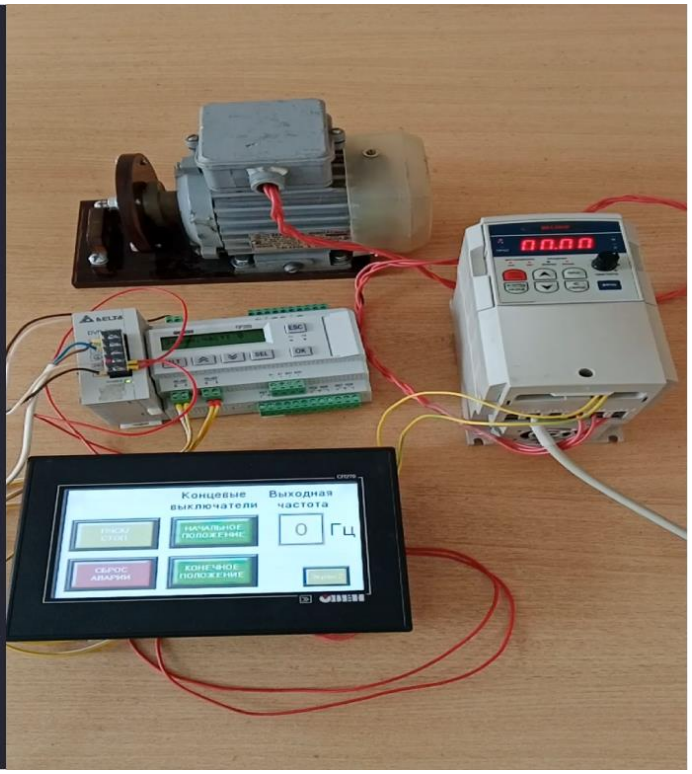
Итоговая управляющая программа

Интерфейс панели оператора

1

2

Тестирование программы на оборудовании



Заключение

В рамках данного дипломного проекта была успешно разработана программа управления приводом продольного перемещения суппорта токарного станка на базе ПЛК ОВЕН ПР200 и панели оператора ОВЕН СП270 с использованием частотного преобразователя ВЕСПЕР Е2–8330.
Основной целью работы являлось создание автоматизированной системы, обеспечивающей возвратно-поступательное движение суппорта с возможностью управления через интуитивный интерфейс оператора.