

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет

вычислительной техники

Кафедра

«Вычислительная техника»

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

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

ДИПЛОМНЫЙ ПРОЕКТ

на тему

**СИСТЕМА УПРАВЛЕНИЯ ЭЛЕКТРОПРИВОДОМ
ТОКАРНО-ВИНТОРЕЗНОГО СТАНКА**

Студент
(подпись, дата)

(ФИО полностью)

Котовский Максим Александрович

Руководитель
(подпись, дата)

(фамилия, инициалы)

Тарасов Д.А.

Консультанты: по
(наименование раздела)(подпись, дата)

(фамилия, инициалы)

Нормоконтролёр
(подпись, дата)

(фамилия, инициалы)

А.С.Бычков

Рецензент

(должность, место работы) (подпись, дата)

(фамилия, инициалы)

Работа допущена к защите (протокол зас. кафедры от 22.05.2026 г. № 14)

Заведующий кафедрой
(подпись)

(фамилия, инициалы)

М.А. Митрохин

Работа защищена с отметкой (протокол заседания ГЭК от 10 июня 2026 г. № 5)

Секретарь ГЭК

Е.И. Гурин

Пенза 2026

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

Факультет
вычислительной техники

Кафедра
«Вычислительная техника»

«Утверждаю»
Заведующий кафедрой
М.А. Митрохин
« 11 » 02 2026 г.

**ЗАДАНИЕ
НА ДИПЛОМНЫЙ ПРОЕКТ**

1. Студент Котовский Максим Александрович гр. 21ВОЭ1 факультета Вычислительной техники
2. Тема работы «Система управления электроприводом токарно-винторезного станка»

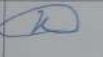
Тема утверждена приказом ПГУ № 122/о от "11.02.2026" 2026 г.

3. Руководитель работы к.т.и доцент кафедры ВТ Тарасов Д.А.

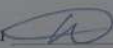
4. Задание на работу (назначение разработки, исходные данные и т.п.)
Назначением разработки является создание управляющей программы и интерфейса рабочего стола оператора токарно винторезного станка для автоматизации процессов обработки торца, поверхности детали и нарезания резьбы. Система должна обеспечивать ввод параметров оператором с панели управления, выбор режима работы и автоматическое выполнение процесса, а также предоставлять удобный интерфейс оператора для ввода параметров и отслеживания выполнения процесса в реальный момент времени. Исходными данными для разработки являются целевая платформа (OBERN ПЛК154ИМ, преобразователь частоты, электродвигатель), а так же возможности среды разработки Codesys.

5. Перечень подлежащих разработке вопросов
В ходе выполнения дипломного проекта необходимо провести анализ предметной области и существующих аналогов, обосновать актуальность разработки и сформулировать техническое задание. Следует выполнить выбор технологического стека для реализации управляющей программы и интерфейса. Необходимо спроектировать архитектуру приложения, включая UML-диаграммы вариантов использования и состояний. Требуется разработать управляющую программу на основе языков программирования CFC и ST, используя возможности среды Codesys 2.3. Также необходимо создать простой интуитивно понятный интерфейс на основе встроеной в среду разработки системы визуализации. В составе пояснительной записки должны быть представлены разделы по безопасности жизнедеятельности и экономическому обоснованию проекта.

6. Календарный график выполнения работы

№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1.	Изучение предметной области и анализ существующих аналогов	40	11.02.2026-16.02.2026	
2.	Анализ требований и постановка задачи	20	17.02.2026-20.02.2026	
3.	Выбор технологий, применяемых при проектировании и разработке (Codesys 2.3)	30	21.02.2026-25.02.2026	
4.	Проектирование архитектуры системы.	60	26.02.2026-06.03.2026	
5.	Разработка управляющей программы	80	07.03.2026-27.03.2026	
6.	Разработка интерфейса панели оператора	80	01.05.2026-11.05.2026	
7.	Тестирование системы в режиме эмуляции и на лабораторном стенде	40	12.05.2026-17.05.2026	
8.	Оформление пояснительной записки (включая БЖД и экономику)	20	18.05.2026-21.05.2026	

Дата выдачи задания " 14 " 02 2026 г.

Руководитель работы 
(подпись, дата)

Задание к исполнению принял студент 
(подпись, дата)

Работу к защите допустить

Декан факультета  Л.Р. Рисуев
03.06.2026
(подпись, дата)

Реферат

Пояснительная записка содержит 140 листов, 120 рисунков, 7 таблиц, 13 источников, и 7 приложений.

СИСТЕМА УПРАВЛЕНИЯ ЭЛЕКТРОПРИВОДОМ ТОКАРНО-ВИНТОРЕЗНОГО СТАНКА

Цель работы – разработать систему автоматизированного управления электроприводом токарно-винторезного станка на базе программируемого логического контроллера, обеспечивающего контроль параметров в реальном времени и автоматическое выполнение процессов обработки.

Объектом разработки является управляющая программа для ПЛК и графический интерфейс, которые должны обеспечивать автоматическое выполнение обработки торца детали, цилиндрической поверхности, нарезания различных видов резьбы, а так же обеспечение безопасности и контроль ошибок.

Разработка проводилась в среде программирования CODESYS V2.3. В качестве языков программирования были выбраны Structured Text (ST) для реализации алгоритмов управления и CFC для визуализации технологических блоков. Графический интерфейс оператора разработан во встроенном редакторе визуализации CODESYS.

В результате проведенной работы разработана система, удовлетворяющая поставленной задаче, а именно: автоматическое управление токарно-винторезным станком по заданной программе, интерфейс оператора с отображением панели управления и графиков.

					ПГУ 1.090501.05.001 ПЗ		
Изм.	Лист	№ докум.	Подпись	Дата			
Разраб.		Котовский М.А.		05.04.23	Система управления электроприводом токарно-винторезного станка Пояснительная записка		
Пров.		Тарасов Д.А.		05.05.23			
Н. Контр.		Бычков А.С.		05.06.23	ФВТ, каф. ВТ гр. 21ВОЭ1		
Утв.		Митрохин М.А.		06.06.23			
					Литера	Лист	Листов
						2	140

Содержание

Перечень сокращений и обозначений.....	5
Введение	6
1 Программа управления подачей токарно-винторезного станка.....	9
1.1 Анализ предметной области.....	9
1.1.1 Актуальность разработки.....	9
1.1.2 Обзор аналогов	13
1.1.3 Анализ требований к разработке	17
1.1.4 Выбор архитектуры проекта.....	18
1.1.5 Выбор языка программирования.....	26
1.1.6 Выбор среды разработки.....	26
1.2 Проектирование и разработка.....	34
1.2.1 Конфигурирование ПЛК 154	35
1.2.2 Разработка управляющей программы	39
1.2.3 Конфигурирование панели оператора.....	83
1.3 Анализ результатов.....	96
1.3.1 Тестирование и отладка управляющей программы и интерфейса	96
1.3.2 Тестирование проекта на стенде целевой платформы	110
2 Охрана труда	111
2.1 Анализ негативных факторов, воздействующих на оператора ПЭВМ и способы их устранения.....	111
2.1.1 Оборудование рабочего места	112
2.1.2 Освещенность рабочего места	112
2.1.3 Электробезопасность	113
2.1.4 Пожарная безопасность	114
2.1.5 Режим труда и отдыха.....	116
3 Экономика	118
Заключение	121
Список использованных источников	124

Приложение А – UML - диаграммы.....	126
Приложение Б – Схемы программы.....	129
Приложение В – Презентация	133

					ПГУ 1.090501.05.001 ПЗ	Лист
						4
Изм.	Лист	№ докум.	Подпись	Дата		

Перечень сокращений и обозначений

В настоящей работе применяют следующие обозначения и сокращения:

ПК – персональный компьютер

ПЛК – программируемый логический контроллер

ПЛК ОВЕН 154 – программируемый логический контроллер фирмы ОВЕН модель 154

ПЭВМ – персональная электронно-вычислительная машина

СЖО – система жидкостного охлаждения

ЧП – частотный преобразователь

CodeSys – среда программирования для создания алгоритмов работы программируемых логических контроллеров

CFC (Continuous Function Chart) – это графический язык программирования высокого уровня, используемый для создания системы автоматизации (ПЛК)

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

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

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

					ПГУ 1.090501.05.001 ПЗ	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дата		

Введение

Качественная металлообработка на токарно-винторезных станках невозможна без точного и отказоустойчивого управления электроприводами, от которых зависит как производительность оборудования, так и соблюдение технологических допусков при изготовлении деталей. В зависимости от типа станка, серийности производства и квалификации персонала возникает необходимость в гибкой настройке режимов резания, скорости вращения шпинделя и параметров подачи суппорта. На машиностроительных предприятиях целесообразно опираться на современные средства автоматизации и доступную компонентную базу. Решение данной проблемы достигается путем внедрения специализированной системы управления на базе программируемого логического контроллера, что позволяет минимизировать брак, исключить ошибки ручного управления и обеспечить стабильность обработки при нарезании резьбы.

Эффективность работы оборудования во многом определяется тем, насколько удобно и безопасно организовано взаимодействие оператора со станком. Данное взаимодействие строится по принципу системы «человек – техника», где оператор задает целевые режимы, а исполнительные механизмы отрабатывают их с заданной точностью. Ключевая задача управляющей системы заключается в своевременной обработке команд оператора (пуск, останов, реверс, изменение частоты) и предоставлении ему достоверных сведений о текущем состоянии узлов станка – частоте вращения, нагрузке на двигатель, положении суппорта, наличии аварийных ситуаций.

По мере усложнения технологических задач автоматизированная система управления включает в себя разнородные компоненты: контроллер, преобразователь частоты, кнопки, датчики положения и тока, устройства индикации. При этом способы передачи информации и выдачи команд оператором становятся более разнообразными – от отдельных кнопок до multifunctional пультов с цифровыми индикаторами. Человек по-прежнему остается центральным звеном, принимающим решения, однако его

					ПГУ 1.090501.05.001 ПЗ	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дата		

функции постепенно смещаются от непосредственного управления к контролю и корректировке автоматических режимов. Такая модель взаимодействия называется человеко-машинным интерфейсом.

В промышленной автоматизации для организации диспетчерского контроля и визуализации технологических процессов широко применяются SCADA-системы (Supervisory Control And Data Acquisition). Данный термин изначально означал лишь сбор данных и дистанционное управление, однако современные SCADA-решения включают в себя развитые средства для создания графических интерфейсов, архивации параметров, сигнализации и формирования отчетов. Применительно к токарно-винторезному станку функции визуализации и оперативного управления могут быть реализованы как через стандартный кнопочный пост, так и через сенсорную панель оператора с отображением мнемосхем и трендов.

Управление технологическим процессом механической обработки строится по иерархическому принципу. Программируемый логический контроллер опрашивает датчики (сигналы конечных выключателей, кнопок), выполняет алгоритмы пуска и торможения двигателей, управляет реверсированием, а также обеспечивает синхронизацию вращения шпинделя и перемещения суппорта при нарезании резьбы. Контроллер также отвечает за безопасность – он отслеживает состояние защитного кожуха, тепловую защиту двигателей и при возникновении нештатной ситуации немедленно отключает исполнительные механизмы. Помимо этого, ПЛК обменивается данными с другими системами по промышленным сетям (RS-485, Modbus RTU), передавая информацию о режимах работы и принимая новые уставки.

Рабочее место оператора представляет собой кнопочный пост и ламповую индикацию, которые позволяют подавать базовые команды и визуально оценивать состояние станка. В более развитых конфигурациях используется панель оператора с текстовым или графическим дисплеем, на которой отображаются текущие параметры (фактическая частота вращения, шаг резьбы, пройденное расстояние), а также предоставляется возможность оперативно изменять технологические

					ПГУ 1.090501.05.001 ПЗ	Лист
						7
Изм.	Лист	№ докум.	Подпись	Дата		

настройки без остановки процесса. Такая организация управления снижает нагрузку на персонал, повышает наглядность информации и создает условия для ведения электронного журнала обработки деталей.

Таким образом, автоматизация управления электроприводом токарно-винторезного станка на базе современных программно-аппаратных средств позволяет повысить точность и качество обработки, обеспечить безопасность работы оператора, снизить его утомляемость, расширить технологические возможности станка (в частности, реализовать точное нарезание резьбы) и создать основу для дальнейшей интеграции станка в автоматизированные производственные участки.

Целью выпускной работы является:

- анализ существующих схем управления и обоснование выбора аппаратной платформы;
- создание алгоритмов управления двигателем шпинделя (пуск, реверс, плавная регулировка частоты) и двигателем подачи;
- реализацию функции нарезания резьбы с возможностью задания шага и длины резьбы из программы;
- проектирование управляющей программы в среде CODESYS 2.3 на языке функциональных блоков SFC для контроллера ОВЕН ПЛК154;
- разработку системы блокировок и аварийной защиты для безопасной работы оператора;
- тестирование разработанных алгоритмов в симуляторе или на лабораторном стенде.

Предлагаемая система ориентирована на модернизацию действующего парка токарно-винторезных станков на машиностроительных предприятиях. Ее внедрение позволит повысить точность обработки, расширить номенклатуру выполняемых резьб, снизить риск аварийных ситуаций и создать основу для дальнейшей интеграции станков в автоматизированные производственные линии.

1 Программа управления подачей токарно-винторезного станка

1.1 Анализ предметной области

1.1.1 Актуальность разработки

В современной промышленности токарно-винторезные станки остаются одним из ключевых видов оборудования. Они используются во многих производственных отраслях и находят применение от частных мастерских и мелкосерийных производств до крупнейших предприятий.

Токарно-винторезный станок – это универсальное оборудование, предназначенное для выполнения широкого спектра задач: обработки цилиндрических или конических поверхностей, сверления, растачивания отверстий, подрезки торцов и фасок и нарезания всех видов резьбы. Основными узлами станка являются станина, передняя и задняя бабки, коробка подач, фартук и суппорт с резцедержателем. На рисунке 1 представлены общий вид и органы управления токарно-винторезного станка 16К20. Внешний вид станка 16К20 показан на рисунках 2 и 3.



Рисунок 1 - Общий вид и органы управления станка 16К20



Рисунок 2 – Токарно-винторезный станок 16K20



Рисунок 3 – Токарно-винторезный станок 16K20

Все токарно-винторезные станки устроены однотипно, а главным аспектом применимости станка в том или ином производстве, является его класс точности. Классы точности станков делятся на нормальную, повышенную, высокую точность, и особо-точные. Использование оборудования, не обеспечивающего необходимую точность, не допустимо, например, в аэро-космической отрасли, в ядерной энергетике, в производствах медицинского оборудования или точных измерительных приборов. Также и закупка дорогих высокоточных и особо-точных станков нецелесообразна, к примеру, для производства потребительских или декоративных товаров, к которым не предъявляются высокие требования. Согласно ГОСТ 18097-93 [4], класс точности станка определяется множеством параметров. Измеряется прямолинейность перемещения суппортов, одновысотность осей вращения, точность установки направляющих, биение вращающихся агрегатов и многое другое.

Эффективность работы токарно-винторезных станков, точность обработки деталей и надежность напрямую зависят не только от конструктивных особенностей, обеспечивающих соответствие заявленному классу точности, но системы управления приводом. На данный момент в подавляющем большинстве производств используются советские станки с релейно-контакторной системой управления приводом. Их устройство предполагает наличие магнитных пускателей, которые управляют электродвигателями. Защиту от перегрузки обеспечивает тепловое реле, а ограничители перемещения каретки выполнены с помощью концевых выключателей. Для нарезания резьбы в этих станках используется коробка подач, которая передает вращение шпинделя к суппорту, котором закреплен резец. Выбор режимов работы коробки подач осуществляется вручную. Установив деталь в токарный патрон, оператор подводит суппорт станка в исходное положение, настраивает необходимые параметры резьбы, выбирая режим коробки подач при помощи рукояток. Сверяясь со специальной таблицей, вращением рукояток оператор выбирает частоту вращения, вид резьбы, метрическую или дюймовую, шаг резьбы и направление. Кроме того, для метрической и дюймовой резьбы должны быть установлены сменные зубчатые

					ПГУ 1.090501.05.001 ПЗ	Лист
						11
Изм.	Лист	№ докум.	Подпись	Дата		

колеса, передаточное число которых оператор должен рассчитать по специальной формуле. Только после этих действий оператор замыкает гайку ходового винта, включает станок и выполняет первый проход. Когда резец проходит нужную длину, оператор, рукояткой управления фрикционной муфтой, отключает вращение шпинделя и одновременно с этим быстро отводит резец рукояткой поперечной подачи. Затем оператор переставляет рукоятку управления фрикционной муфтой в положение реверс, тем самым отводя суппорт в начальное положение. Регулируя глубину реза рукояткой поперечной подачи, оператор выполняет таким же образом еще несколько проходов.

При работе на станке с конструкцией привода, описанной выше, вероятность ошибки, вызванной человеческим фактором, возрастает многократно. Оператор может быть недостаточно квалифицирован, так же его невнимательность или усталость может привести к появлению производственного брака. Кроме того, весь процесс занимает существенное время. Модернизация устаревших станков или разработка новых систем управления позволяет повысить производительность труда и улучшить качество выпускаемой продукции. К примеру, при производстве блоков и шатунно-поршневых групп для двигателя ВАЗ-21186 необходимо обеспечить номинальный зазор между поршнем и стенкой цилиндра, который составляет от 0,025 до 0,045 миллиметра. Предельно допустимый зазор – 0,015 миллиметра. Для того же двигателя зазор между коренной или шатунной шейкой коленчатого вала и вкладышем составляет 0,020-0,070мм, а между стержнем и направляющей клапана 0,025-0,055мм на впуске и 0,040-0,070мм на выпуске. Нарушение хотя бы одного из этих допусков приводит к повышенному износу агрегата и ускоренному выходу его из строя. В условиях крупного серийного производства двигателей этой модели и запасных частей к ним (сотни тысяч штук), завод изготовитель должен обеспечить высочайшую точность обработки перечисленных деталей, идеальную повторяемость и возможность быстрой перенастройки оборудования на разные детали, чего невозможно достичь, используя устаревшие или некачественные станки.

					ПГУ 1.090501.05.001 ПЗ	Лист
						12
Изм.	Лист	№ докум.	Подпись	Дата		

В рамках данного проекта разрабатывается система управления электроприводом токарно-винторезного станка на базе современной элементной базы. Актуальность работы обусловлена необходимостью замены морально и физически устаревших релейно-контакторных схем управления на более надежные и функциональные решения с использованием программируемых логических контроллеров (ПЛК) и преобразователей частоты.

Разрабатываемая система имеет ряд преимуществ перед классическими решениями:

- возможность плавного пуска и торможения электродвигателей, что снижает механические нагрузки на узлы станка;
- высокая точность поддержания частоты вращения шпинделя, влияющая на качество обработки;
- возможность гибкой настройки и изменения логики работы без переделки силовых цепей;
- повышение надежности и ремонтпригодности оборудования за счет замены релейной автоматики на промышленный контроллер;
- возможность интеграции станка в единую производственную сеть предприятия (АСУ ТП);
- повышение электробезопасности и снижение энергопотребления.

Применение преобразователя частоты позволяет не только регулировать скорость вращения трехфазного асинхронного двигателя, но и реализовывать режимы позиционирования, что особенно важно для токарных работ, требующих реверса и точного останавливания суппорта.

1.1.2 Обзор аналогов

На сегодняшний день существует несколько подходов к построению систем управления металлорежущими станками. Условно их можно разделить на устаревшие релейно-контакторные системы, современные системы на базе универсальных ПЛК и специализированные системы ЧПУ.

Для сравнительного анализа были выбраны три подхода, наиболее часто встречающихся при эксплуатации и модернизации токарно-винторезных станков:

					ПГУ 1.090501.05.001 ПЗ	Лист
						13
Изм.	Лист	№ докум.	Подпись	Дата		

- традиционная релейно-контакторная схема управления;
- система управления на базе универсального ПЛК и частотного преобразователя (разрабатываемое решение);
- специализированная система ЧПУ (например, на базе контроллера и блока питания, широко представленная на рынке).

Краткий анализ систем управления представлен в таблице 1.

Таблица 1 – Анализ преимуществ и недостатков аналогов системы управления

Тип системы управления	Преимущества	Недостатки
Релейно-контакторная	Простота схемы для нерегулируемого привода, низкая стоимость оборудования и компонентов, ремонтпригодность	Невозможность плавной регулировки оборотов, низкая надежность (механический износ контактов), жесткая логика (сложность модернизации), громоздкость шкафов управления.
Система "ПЛК + ПЧ"	Гибкость логики управления, высокая надежность, плавный пуск и точное регулирование скорости, энергоэффективность, компактность, возможность расширения функционала.	Требуется квалифицированный специалист для настройки и программирования, более высокая начальная стоимость по сравнению с релейной схемой.
Специализированная система ЧПУ	Максимальная автоматизация (возможность обработки сложных деталей по программе), высочайшая точность	Чрезмерно высокая стоимость для простых токарных работ, сложность эксплуатации и настройки, избыточность функционала для задач, требующих только ручного управления.

В таблице 2 предоставлен сравнительный анализ некоторых токарно-винторезных станков разных производителей.

Таблица 2 – Сравнительный анализ станков

Модель / Производитель	Класс точности	Система управления / Особенности	Цена, руб.	Ключевые особенности и применение
1K62 (СССР)	Нормальный	Релейно-контакторная. Классическая механика с магнитными пускателями и конечными выключателями. Механическая КП (23 ступени) .	250 000 - 500 000 (б/у)	Универсальный станок. Самая массовая модель в СССР. Высокая надежность, ремонтпригодность. Нарезание всех типов резьб. До сих пор работает на тысячах предприятий.
16K20 (СССР)	Повышенной	Релейно-контакторная Усиленная конструкция, Механическая КП (22 ступени) .	400 000 - 700 000 (б/у)	Более жесткий и точный, чем 1K62. Широко применяется в тяжелом машиностроении.
Proma SPI- 1500	Нормальный	Ручное управление. Механическая КП (6 ступеней).	420 000	Компактный станок для небольших мастерских, обучения и мелкосерийного производства.
JET GHB- 1330A DRO (Германия)	Нормальный	Ручное управление по 3 осям.	842 000	Профессиональный станок для мастерских и автосервисов. Двухвинтовая конструкция (ходовой винт и ходовой вал).
OPTIM UM TH4210	Нормальный	Ручное управление. Механическая КП (16 ступеней).	1 028 075	Промышленный станок для серийного производства. Переключение резьб рукоятками без сменных шестерен.

Модель / Производитель	Класс точности	Система управления / Особенности	Цена, руб.	Ключевые особенности и применение
Stalex C6266A/ 3000	Нормальный	Ручное управление.	2 800 000	Для средних производств и ремонта тяжелых деталей.
16K40/3 000 (Россия)	Повышен- ной	Ручное управление. Механическая КП (22 ступени).	3 200 000	Универсальный станок для серийного производства и ремонтных мастерских. Все типы резьб.
JET GH- 26120Z H DRO (Германия)	Нормальный	Ручное управление.	3 600 000	Тяжелый станок для длинномерных заготовок. 65 предустановленных режимов подач.
Metal Master ZH 66300 DRO RFS	Особо высокой точности?)	Прецизионные работы, револьверная головка для быстрой смены инструмента.	4 900 000	Высокая точность, автоматизация процессов, цифровая индикация с точностью до 0,001 мм .
Модель / Производитель	Класс точности	Система управления / Особенности	Цена, руб.	Ключевые особенности и применение
1И250В (Россия)	Высокий	Ручное управление.	5 350 000	Для высокоточных работ с деталью диаметром до 240 мм. Механизм-верньер позволяет перемещаться с точностью 0,005 мм.
Stalex CQ6280/ 3000	Нормальный	Ручное управление с гидравлической системой подачи.	6 500 000	Для тяжелой промышленности, обработка заготовок до 8 тонн.

Модель / Производитель	Класс точности	Система управления / Особенности	Цена, руб.	Ключевые особенности и применение
DMTG CW6263 E/5000	Нормальный	Ручное управление. ЧПУ-совместимость.	8 500 000	Рекордная длина обработки (5 м), грузоподъемность до 5 тонн. Требуется больших производственных площадей.
250ИТВ М.03	Высокий	Ручное управление с преселективным переключением скоростей.	2 200 000	Высокоточный станок. беззазорная фиксация резцовой головки, точность поперечных перемещений 0,005 мм.
ТПК 125 ВНМ (СССР)	Высокий	Для малогабаритных деталей с микронной точностью.	646 800 (б/у)	Прецизионный станок с ЧПУ для обработки малогабаритных деталей с микронной точностью. 6-позиционная револьверная головка.

На основании проведенного анализа можно сделать вывод, что для задач модернизации парка обычных токарно-винторезных станков, система управления на базе ПЛК и преобразователя частоты является наиболее оптимальным решением. Она сочетает в себе надежность и функциональность современных средств автоматизации с невысокой стоимостью и простотой эксплуатации.

1.1.3 Анализ требований к разработке

В рамках проекта разрабатывается система управления электроприводом токарно-винторезного станка, ориентированная на замену устаревших релейных схем или создание новых станков с ручным управлением и вспомогательной автоматикой. Система должна обеспечивать надежную работу в условиях промышленного цеха (запыленность, вибрация, колебания температуры).

Требуется реализовать следующий функционал:

- управление главным приводом: пуск, останов, плавное регулирование частоты вращения, реверс;
- управление приводом суппорта: быстрый ход, рабочая подача;

- управление вспомогательными механизмами (например, насосом СЖО, освещением рабочей зоны);
- контроль состояния конечных выключателей и блокировок (защита от перегрузок, контроль защитных кожухов);
- сбор и отображение информации о состоянии механизмов на панели оператора (индикаторами или текстом);
- обеспечение возможности ручного управления станком с кнопочного поста;

Управление электроприводом должно быть организовано таким образом, чтобы исключить возможность опасных ситуаций для оператора (непреднамеренный пуск, работа без использования защитных кожухов).

1.1.4 Выбор архитектуры проекта

Архитектура программного обеспечения для управления электроприводом токарно-винторезного станка должна обеспечивать надежность, безопасность и простоту отладки. Исходя из анализа требований, была выбрана модульная циклическая архитектура, характерная для систем реального времени на базе ПЛК.

Работа программы представляет собой бесконечный цикл, последовательно выполняющий следующие шаги:

- чтение входов: опрос физических входов контроллера (кнопки пуска, останова, конечные выключатели, датчики) и запись их состояний в память.
- выполнение программы: обработка логики управления согласно алгоритму. На этом этапе, на основе состояний входов и текущего состояния механизмов, вычисляются новые значения для выходов.
- запись выходов: выдача управляющих сигналов на исполнительные механизмы (пускатели, контакторы, преобразователь частоты, сигнальные лампы).
- переход к шагу 1.

Время выполнения цикла зависит от сложности алгоритма и является достаточно малым (миллисекунды), что создает иллюзию параллельной работы всех механизмов.

Взаимодействие с оператором организовано через кнопочный пост, подключенный к дискретным входам. Система будет основана на архитектуре Master-Slave и будет состоять из одного ведущего и трех ведомых устройств. В качестве ведущего устройства выступает контроллер ПЛК154И-М, задавая логику работы и обрабатывая сигналы обратной связи. С ним в сети будет работать модуль расширения дискретного ввода/вывода ПРМ-220.1, позволяющий добавить 8 дискретных входов и 8 дискретных выходов, а также два преобразователя частоты ПЧВ1 для управления двумя электродвигателями. Общий вид контроллера, Модуля расширения и преобразователя частоты представлен на рисунках 4-6. Управление скоростью вращения шпинделя будет осуществляться путем отправки аналогового сигнала (0-10В) или цифровых команд на преобразователь частоты. Взаимодействие между устройствами будет осуществлено через интерфейс RS-485 по протоколу связи Modbus RTU. Интерфейс RS-485 может быть организован только по шине, топологии сетей связи типа «Звезда» и «Кольцо» не поддерживаются, поэтому опрос устройств происходит последовательно. На одной шине может быть подключено до 32 slave-устройств, число которых может быть увеличено до 247 при использовании повторителей интерфейса. В данном проекте с управляющим устройством будет работать только три управляемых, что гарантирует быстрое действие ввиду очень малого времени, затрачиваемого на выполнение циклов опроса. Кроме контроллера, МДВВ и частотных преобразователей, система будет включать в себя электродвигатель подачи суппорта, шаговый мотор поперечной подачи, кнопки, индикаторы, концевые выключатели, а также управлять электродвигателем шпинделя станка. Схема подключения исполнительных устройств представлены на рисунках 7-8.



Рисунок 4 – Контроллер ОВЕН ПЛК154 И-М



Рисунок 5 – Модуль расширения ввода/вывода ПРМ220.1



Рисунок 6 – Преобразователь частоты ПЧВ1

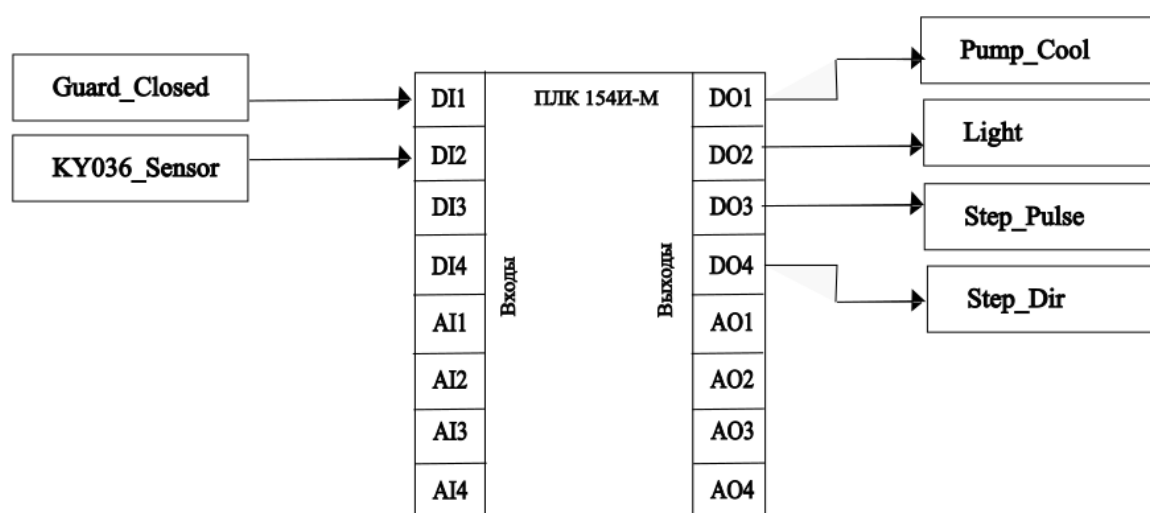


Рисунок 7 – Схема подключения исполнительных устройств

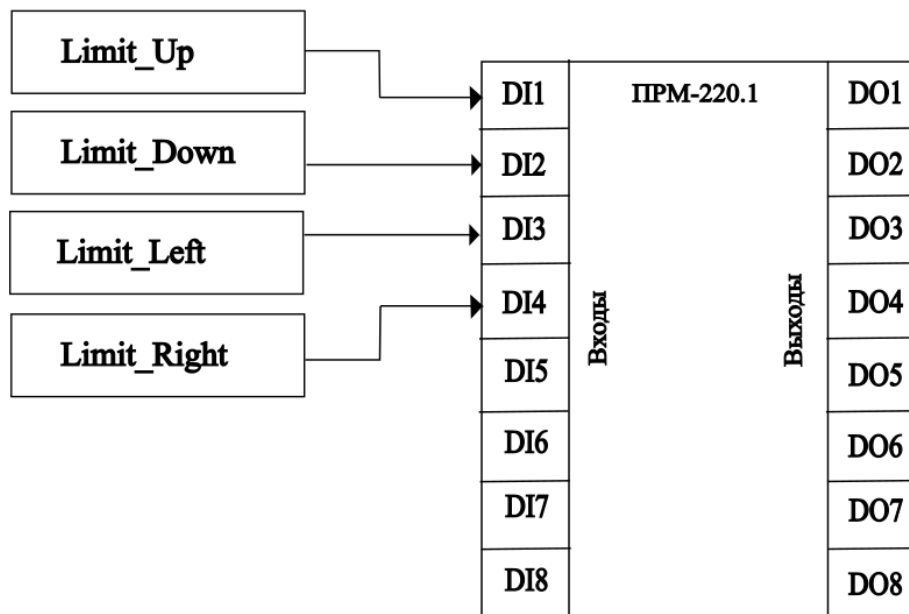


Рисунок 8 – Схема подключения исполнительных устройств

Ко входу ПЛК DI1 подключается концевой выключатель защитного кожуха. При закрытии кожуха он замыкается, сигнал передается в ПЛК и в переменную Guard_Closed записывается единица. Внешний вид концевой выключателя представлен на рисунке 9.



Рисунок 9 – Концевой выключатель

Ко входу ПЛК DI2 подключен сигнальный выход датчика касания КУ-036. Выходы этого датчика подключаются к обрабатываемой детали и к резцу во время ручного перемещения суппорта, и при их соприкосновении сигнал с датчика передается в ПЛК. Внешний вид датчика КУ-036 показан на рисунке 10.



Рисунок 10 – Датчик касания КУ-036

Выходы DO1 и DO2 управляют насосом СОЖ и осветительными приборами над рабочей зоной станка соответственно. Оператор может управлять этими устройствами в любой момент времени выполнения программы, взаимодействуя с ними через кнопки интерфейса.

Выходы DO3 и DO4 управляют шаговым двигателем поперечной подачи. С выхода DO3 на драйвер шагового двигателя передаются импульсы, а выход DO4 задает направление его вращения.

Ко входам DI1, DI2, DI3, DI4 подключены концевые выключатели, ограничивающие продольное и поперечное перемещение суппорта станка.

Исходя из аппаратной платформы (ОВЕН ПЛК154) и выбранной среды разработки (CoDeSys 2.3), был определен следующий набор инструментов: языки стандарта МЭК 61131-3 (в частности, FBD для основной логики и ST для обработки данных), среда программирования CoDeSys 2.3, ОС Microsoft Windows 10.

Программная часть проекта должна состоять из различных функций, каждая из которых отрабатывает определенную часть технологического процесса, выполняемого на токарно-винторезных станках. Должны быть реализованы такие

функции, как Установка нулевой точки, Обработка цилиндрической поверхности, Торцевание, Нарезание метрических и дюймовых резьб с разными параметрами, Управление освещением рабочей зоны и подачей СОЖ, Защита от перегрузок при помощи аварийной остановки, Контроль безопасности техпроцесса. Структура управляющей программы представлена на рисунке 11.



Рисунок 11 – Структура управляющей программы

1.1.5 Выбор языка программирования

Для реализации управляющей программы был выбран язык FBD (Function Block Diagram).

Язык функциональных блоков FBD является графическим языком программирования стандарта IEC 61131-3. Он идеально подходит для разработки алгоритмов логического управления, так как позволяет разработчику создавать программу из готовых компонентов (функциональных блоков), соединяя их подобно электрическим цепям. Это делает логику программы наглядной и интуитивно понятной.

В качестве базовых элементов используются блоки логических операций (И, ИЛИ, НЕ), триггеры (RS, SR), таймеры (TON, TOF), счетчики (CTU, CTD), а также более сложные блоки для математических операций.

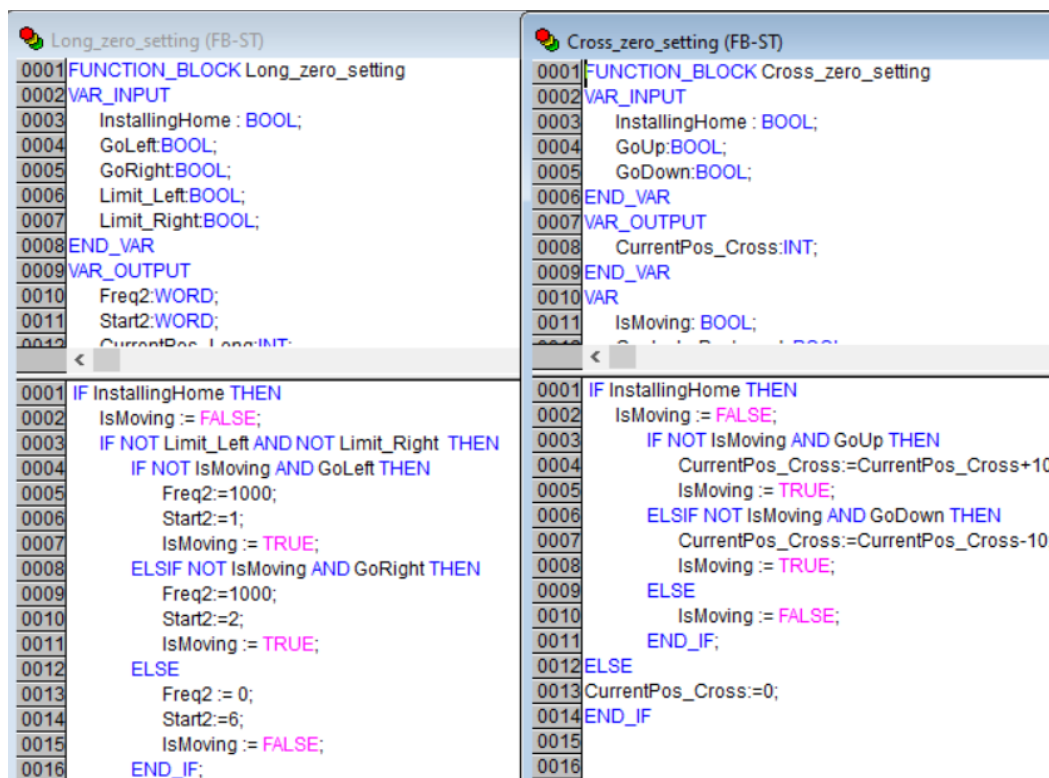


Рисунок 13 – Пример кода на языке структурированного текста

1.1.6 Выбор среды разработки

В качестве среды разработки, исходя из целевой платформы проекта, была выбрана среда программирования CoDeSys V2.3 (The Controller Development System).

CoDeSys V2.3 – это мощная, многоплатформенная среда программирования для промышленных контроллеров, соответствующая международному стандарту IEC 61131-3. Она поддерживает все шесть языков стандарта, что дает гибкость при разработке различных частей проекта. Выбор обусловлен следующими факторами:

- поддержка целевого контроллера: ОВЕН ПЛК154 полностью поддерживается в CoDeSys V2.3. Для него существуют необходимые Target-файлы и библиотеки, например, библиотеки для работы с Modbus.

- функциональность: Среда предоставляет не только редакторы языков, но и мощные инструменты отладки (просмотр переменных в реальном времени, установка брейкпоинтов, симуляция), менеджер библиотек и конфигуратор оборудования.

– конфигурирование оборудования: Вкладка «Конфигурация ПЛК» позволяет настроить аппаратные модули ввода-вывода, задать сетевые параметры для работы по Modbus RTU или TCP/IP, что необходимо для интеграции станка в возможную будущую сеть предприятия.

Среда разработки CoDeSys V2.3 является общедоступной и бесплатной, файл для загрузки представлены на сайте производителя ПЛК ОВЕН. Сайт owen.ru со страницей загрузки CoDeSys V2.3 показан на рисунке 14.

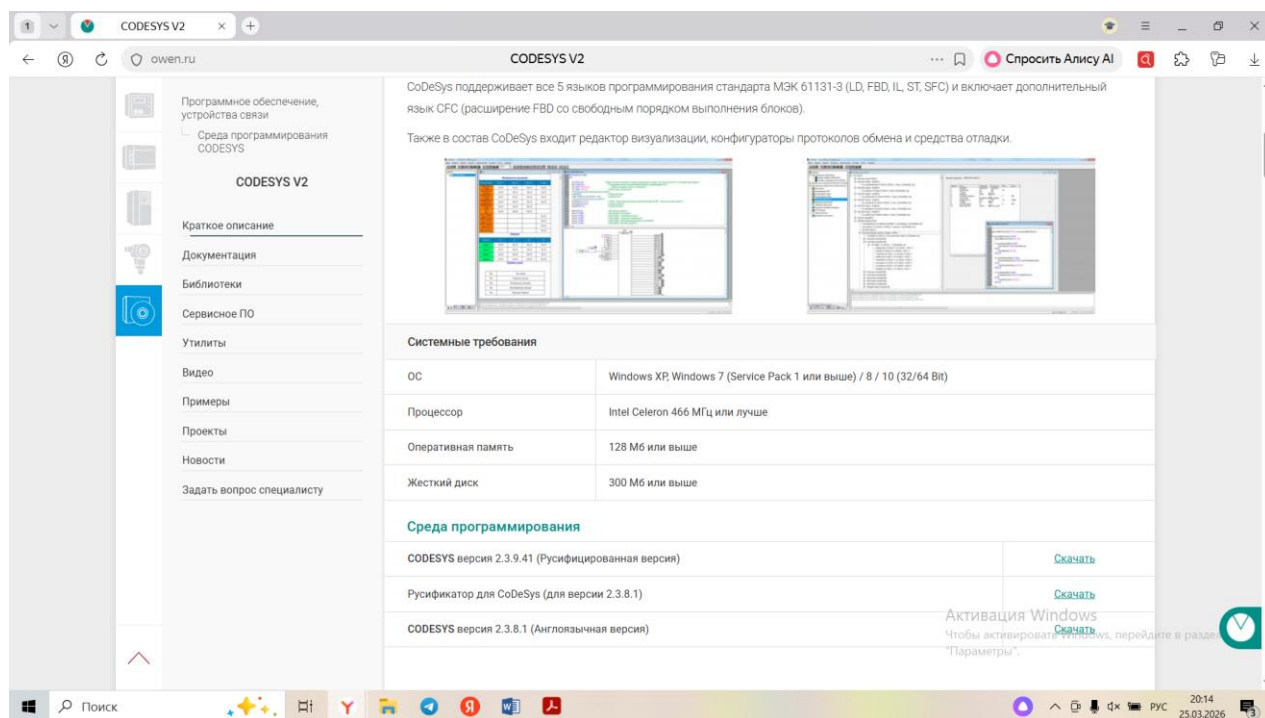


Рисунок 14 – Страница загрузки среды разработки на сайте ОВЕН

Кроме ссылки для загрузки файла установки, страница содержит краткое описание среды, информацию о системных требованиях, а также ссылки на руководства пользователя и иную техническую документацию.

Для загрузки была выбрана последняя представленная на сайте версия CoDeSys - версия 2.3.9.41 (Русифицированная версия) При нажатии кнопки «Скачать» будет загружен архив, содержащий единственный файл установки CoDeSys_v23941.exe.

Установка начинается после запуска исполняемого файла Процесс установки представлен на рисунках 15-18. После запуска установочного файла открывается

Мастер установки CoDeSys, происходит распаковка файлов. Для продолжения установки требуется нажать «Next». Окно Мастера установки показано на рисунке 15.

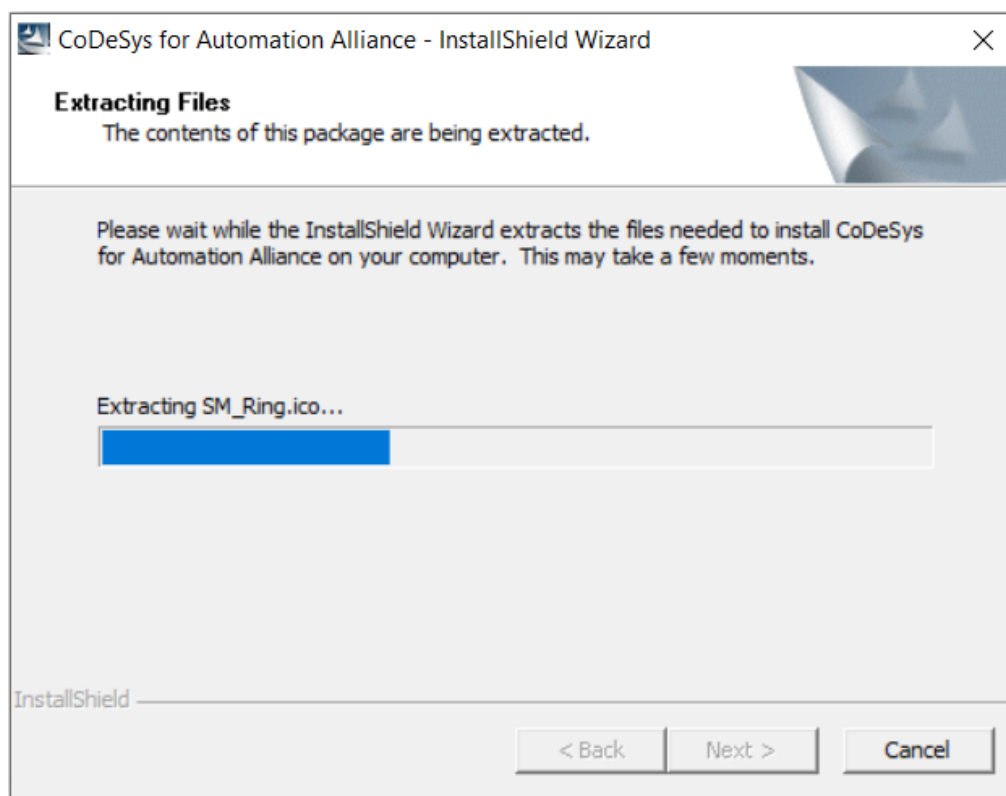


Рисунок 15 – Мастер установки. Распаковка файлов

Далее требуется выбрать необходимые компоненты для установки на рисунке 8, здесь же необходимо выбрать расположение файлов на диске (рисунок 16). Если не указывать расположение, будет выбрана папка по умолчанию на диске C. Для продолжения необходимо нажать «Next».

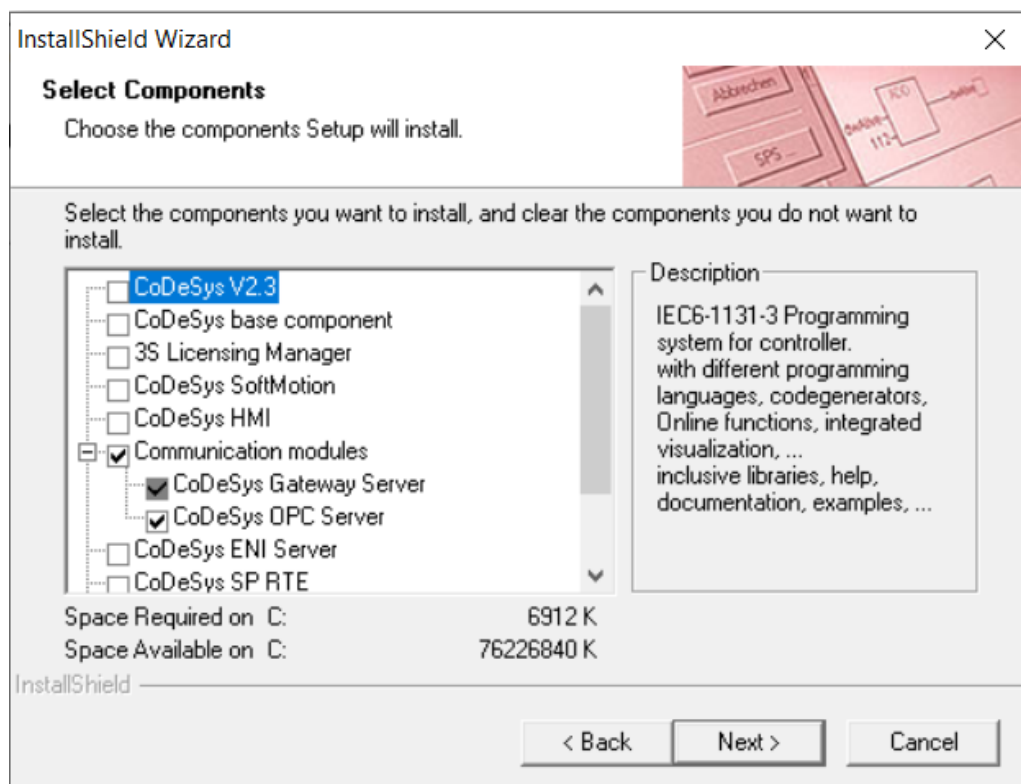


Рисунок 16 – Мастер установки. Выбор компонентов

Далее начинается процесс установки среды разработки и выбранных компонентов (рисунок 17).

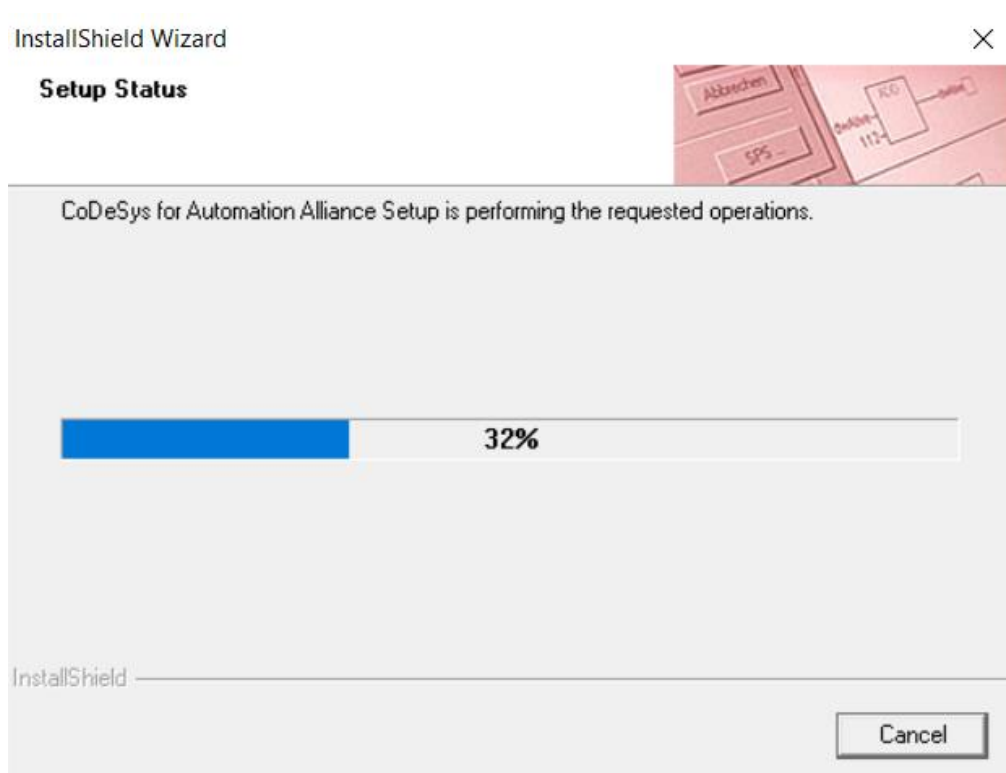


Рисунок 17 – Процесс установки

В конце установки Мастер уведомляет о ее завершении (рисунок 18).

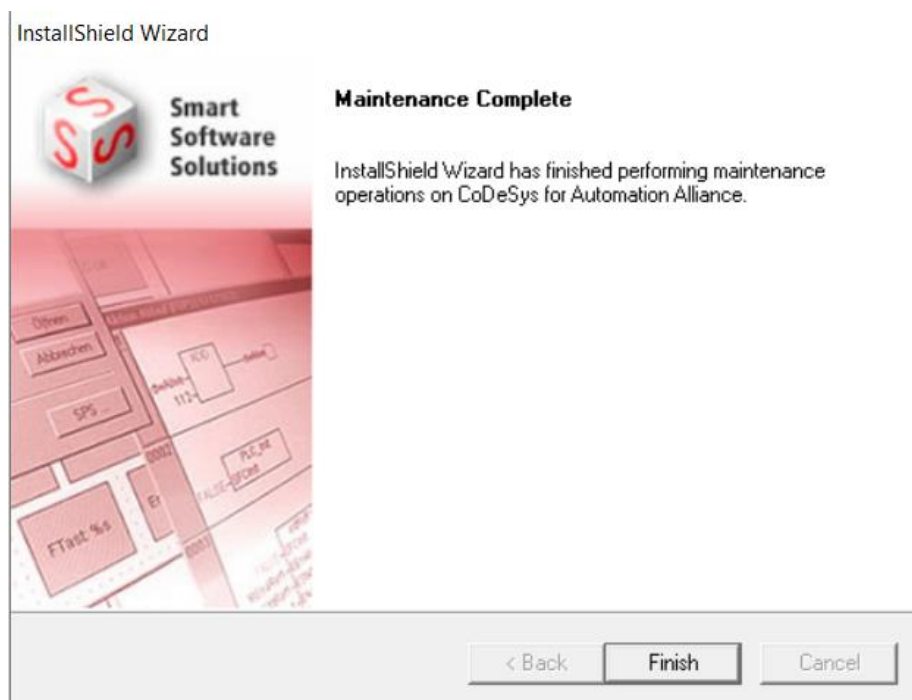


Рисунок 18 – Завершение процесса установки

При запуске среды разработки открывается рабочая область CodeSys 2.3 (рисунок 19). Вместе с запуском создается проект, открывается окно выбора целевой платформы.

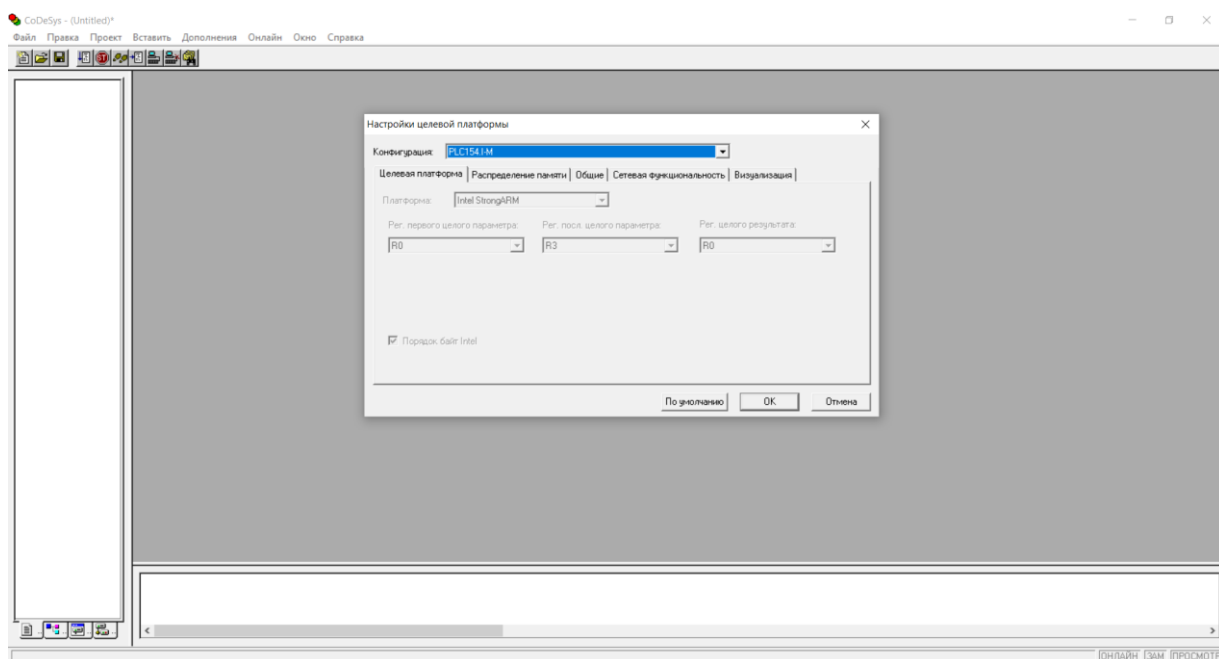


Рисунок 19 – Рабочая область

После выбора целевой платформы создается новый программный компонент (рисунок 20). Необходимо задать его имя, тип и язык реализации.

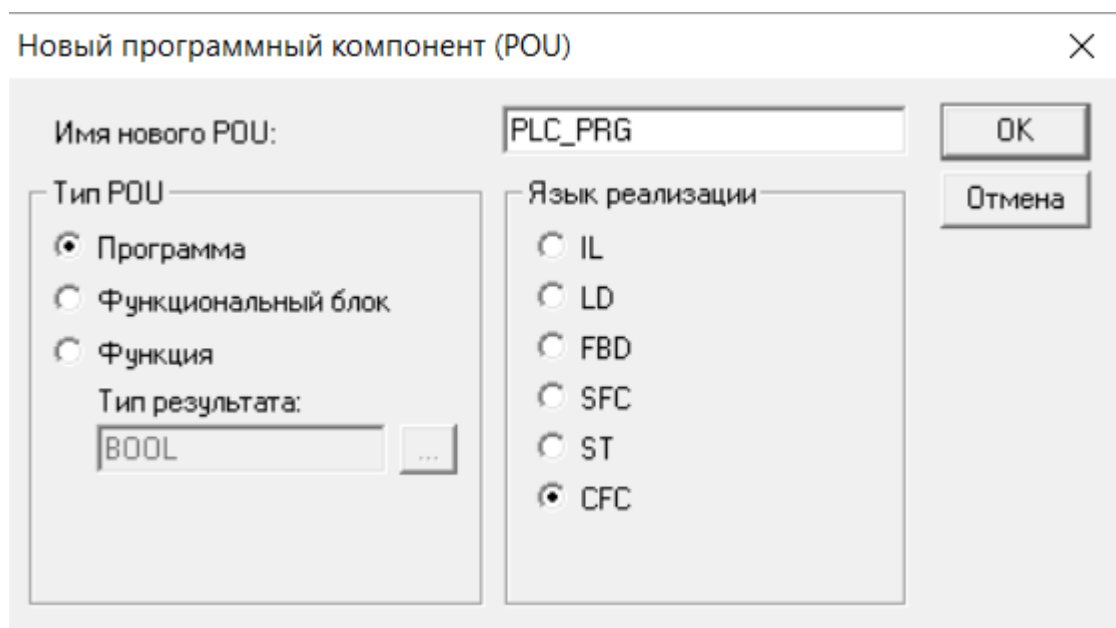


Рисунок 20 – Создание программного компонента

В верхней части рабочей области расположена панель инструментов (рисунок 21). В ней находятся кнопки быстрого доступа «Старт», «Стоп», «Подключение», «Отключение», а так же кнопки «Вырезать», «Копировать», «Вставить», кнопки поиска, и кнопки создания элементов, входных и выходных мереманных, меток, комментариев.

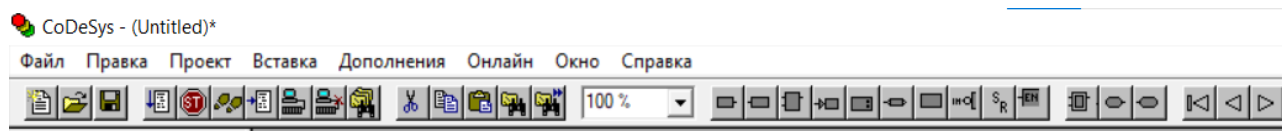


Рисунок 21 – Панель инструментов

Для отладки проекта без использования контроллера и исполнительных устройств используется «Режим эмуляции». Включить его можно в меню «Онлайн» - «Режим эмуляции» (рисунок 22).

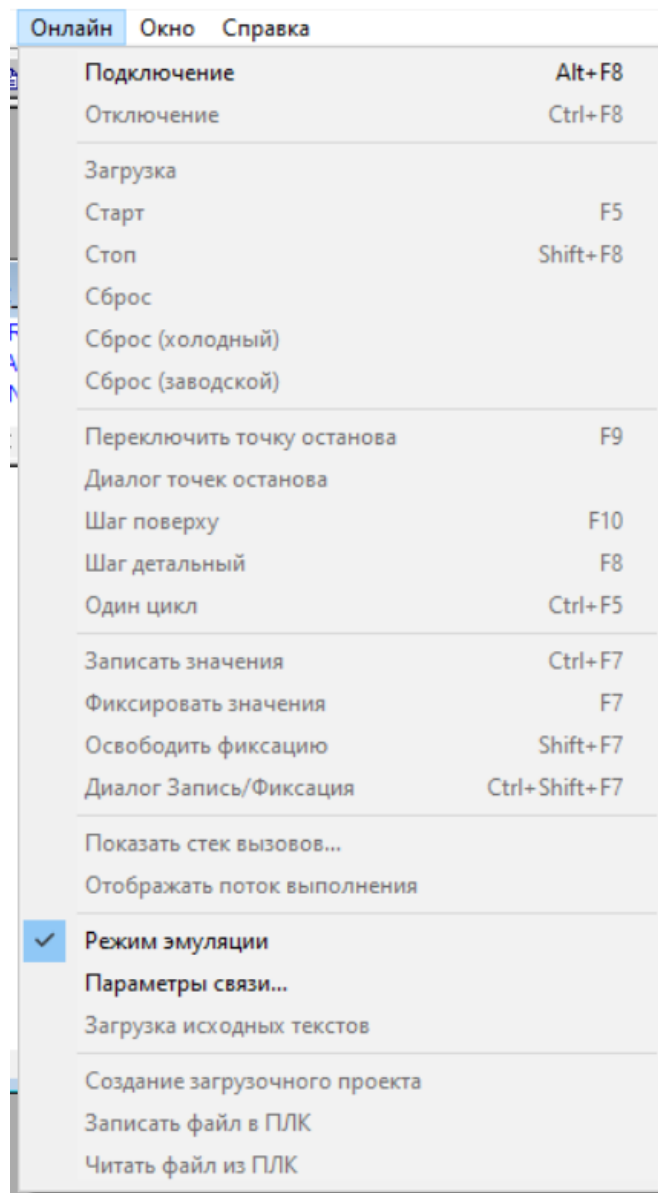


Рисунок 22 – Включение режима эмуляции

В разделе «Ресурсы», - «Конфигурация ПЛК», задаются параметры контроллера, его входы и выходы, а также настраиваются устройства, подключенные по интерфейсу RS-485, такие как модули дистанционного ввода-вывода или преобразователи частоты (рисунок 23).

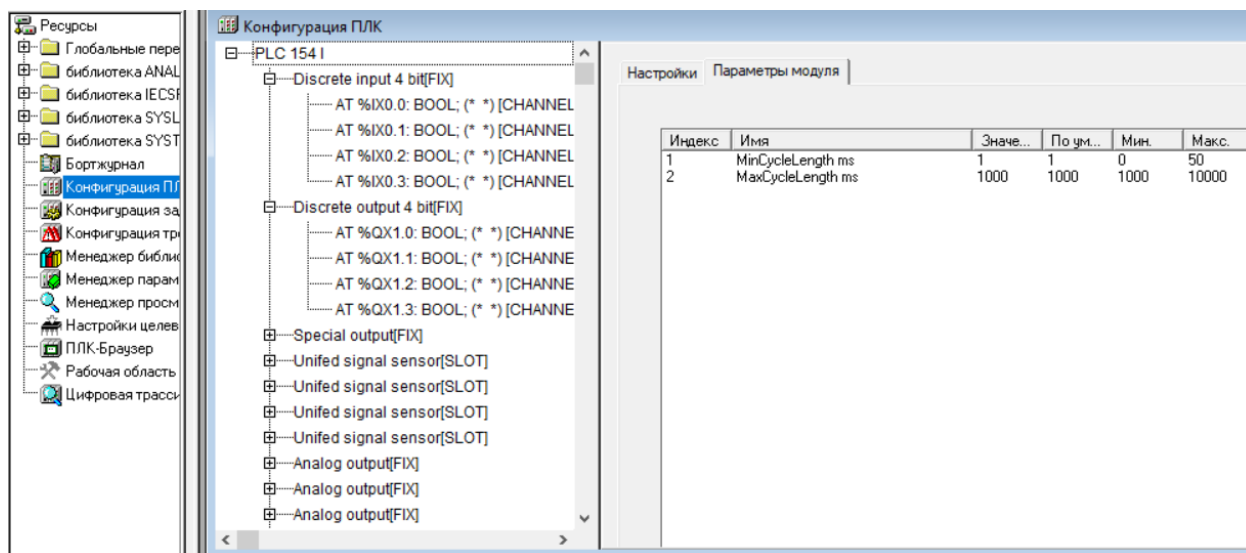


Рисунок 23 – Конфигурация ПЛК

В разделе «Ресурсы» - «Менеджер библиотек» могут быть добавлены библиотеки компонентов, необходимых при разработке (рисунок 24).

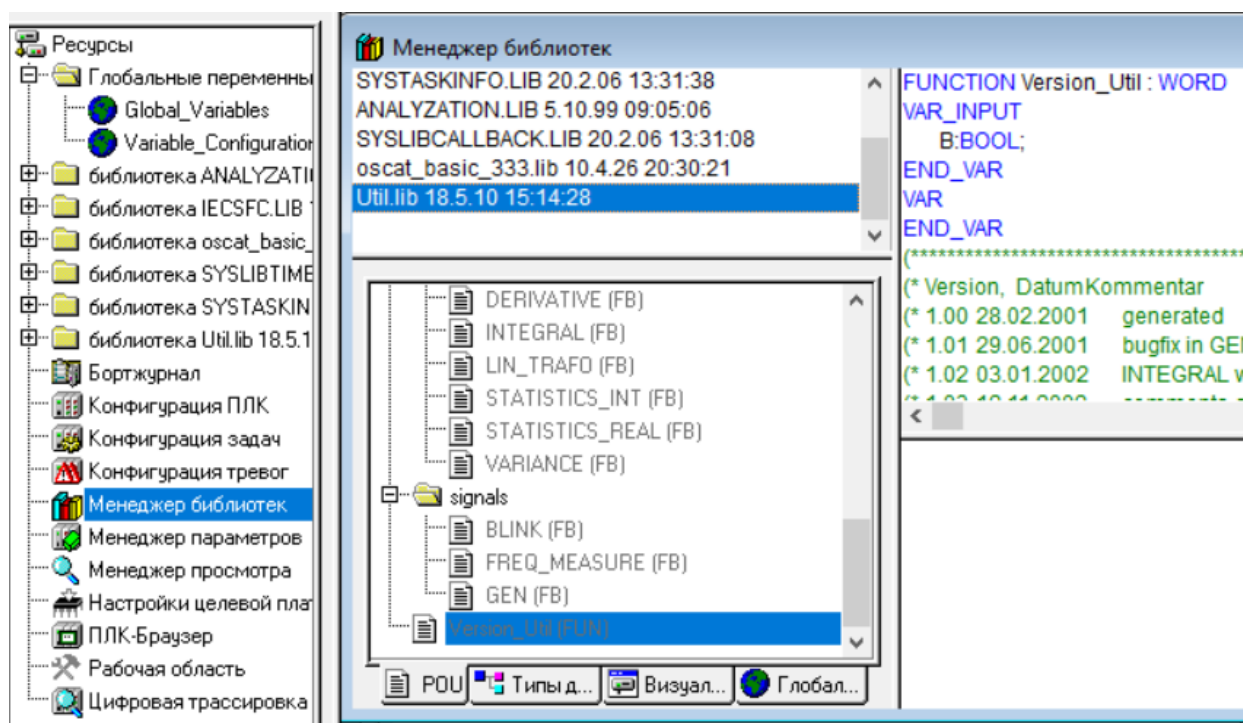


Рисунок 24 – Менеджер библиотек

В Codesys есть встроенные средства визуализации, позволяющие создать простой графический интерфейс для управляющей программы. Чтобы создать визуализацию необходимо перейти на вкладку «Визуализации», по нажатию

правой кнопки мыши в открывшемся меню выбрать пункт «Добавить объект» (рисунок 25).

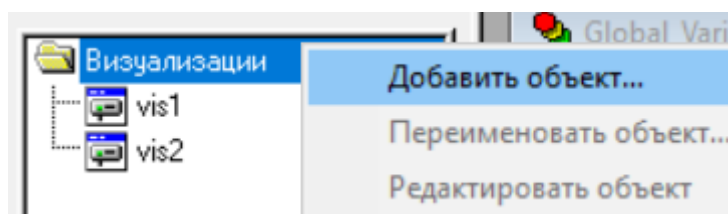


Рисунок 25 – Создание визуализации

После создания визуализации открывается окно с ее рабочей областью. В панели инструментов отображаются элементы, необходимые для создания графического интерфейса (рисунок 26).

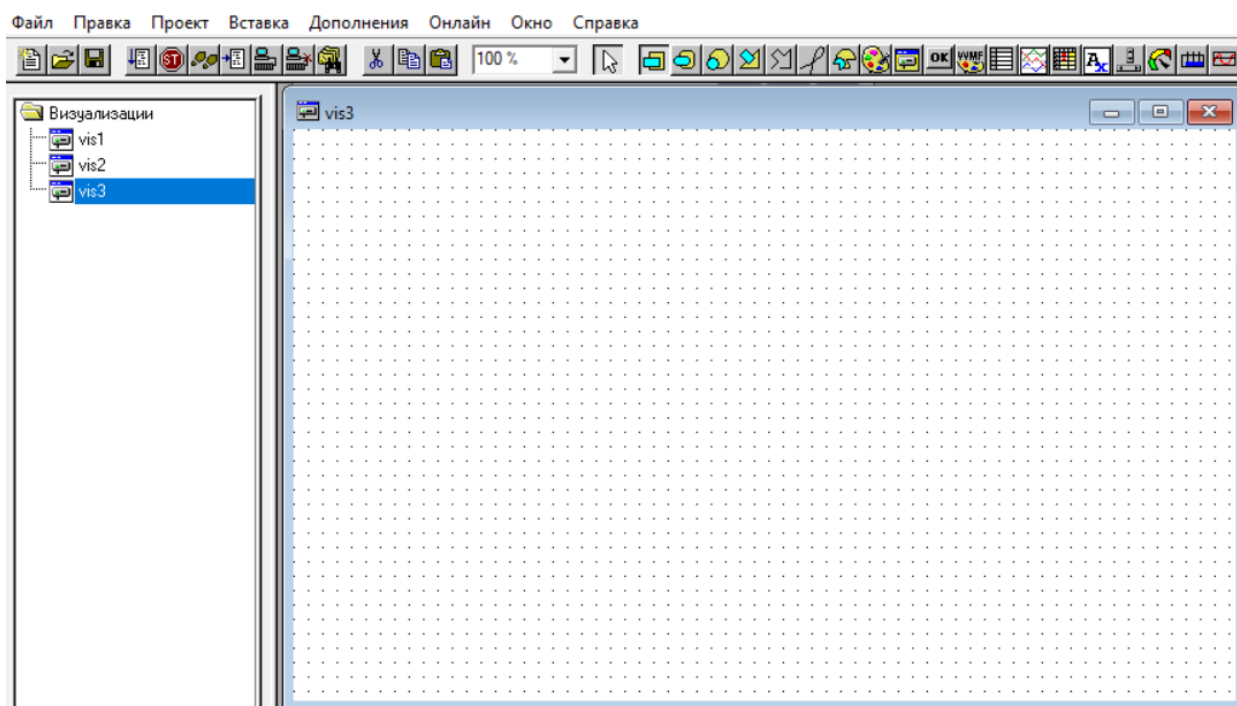


Рисунок 26 – Окно визуализации

1.2 Проектирование и разработка

Проект состоит из управляющей программы и простого графического интерфейса встроенной визуализации codesys. Контроллер ПЛК154И-М, модуль дистанционного ввода-вывода, преобразователи частоты запускаются после

подключения питания. Оператор загружает программу в ПЛК и взаимодействует с ней через интерфейс визуализации.

Подключив ПЛК, оператор выбирает нужный режим, вводит все необходимые параметры и запускает процесс на выполнение. Корректность ввода данных, контроль аварийных ситуаций и вывод сообщений об ошибках обеспечивается при помощи функцией контроля ошибок, автоматически и на протяжении всего процесса. После ввода данных и запуска в работу, процесс выполняется без участия оператора, завершается по достижении заданных параметров. Оператор следит за изменением показаний цифровых полей вывода и графиков, и имеет возможность прервать процесс в любой момент времени.

1.2.1 Конфигурирование ПЛК 154

При создании проекта в качестве целевой платформы был выбран Контроллер ОВЕН ПЛК154И-М (рисунок 27). Он имеет по 4 собственных дискретных входа и выхода, а так же по 4 аналоговых входа и выхода. Для подключения модулей расширения и управления внешними устройствами используется интерфейс RS-485, а для взаимодействия со средой разработки и загрузки управляющих программ – Ethernet.

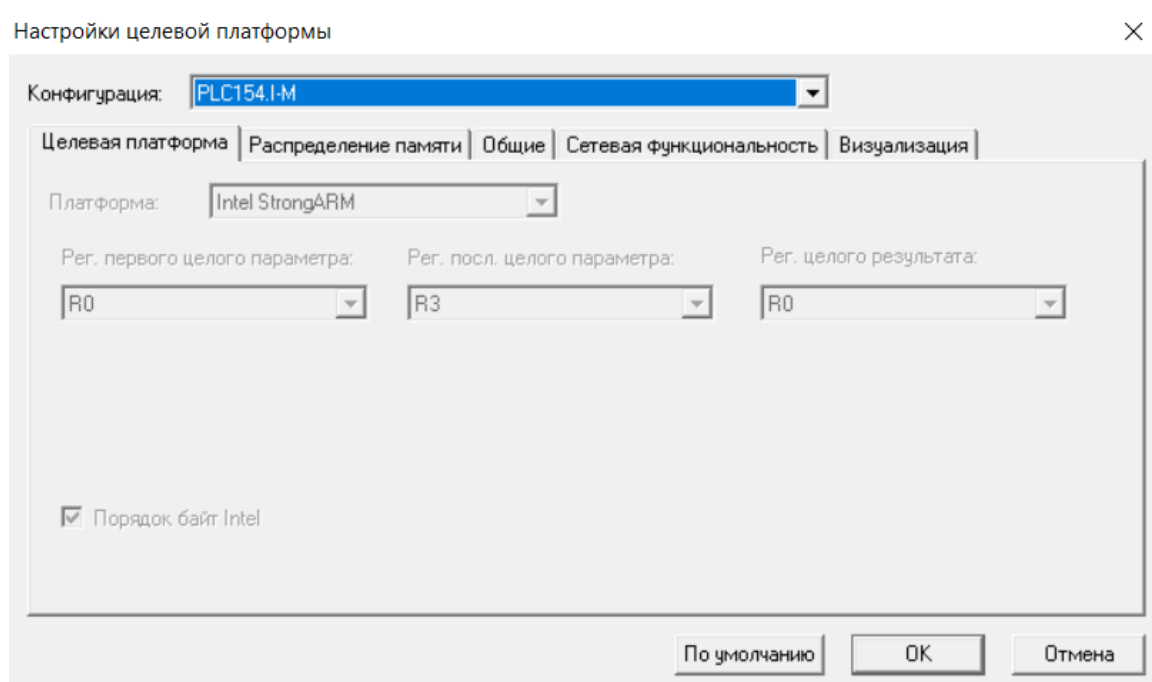


Рисунок 27 – Настройки целевой платформы

После создания проекта настройка контроллера происходит в разделе «Конфигурация ПЛК». В меню «Параметры модуля» все настройки остаются по умолчанию. Далее настраиваются дискретные входы и выходы. В разделе «Discrete input 4 bit» создаются две переменные Guard_Closed и KY036_Sensor, они имеют логический тип и отвечают за прием сигнала с концевого выключателя защитных кожухов и сенсора касания KY036 соответственно. В разделе «Discrete output 4 bit» создаются четыре переменные типа BOOL: Pump_Cool – управляет включением/отключением насоса СОЖ; Light – управляет включением/отключением осветительных приборов над рабочей зоной; Step_Pulse – Импульсы на шаговый двигатель поперечной подачи; Step_Dir – направление вращения шагового двигателя. Аналоговые входы в проекте использоваться не будут. Конфигурация дискретных входов и выходов представлена на рисунке 28.

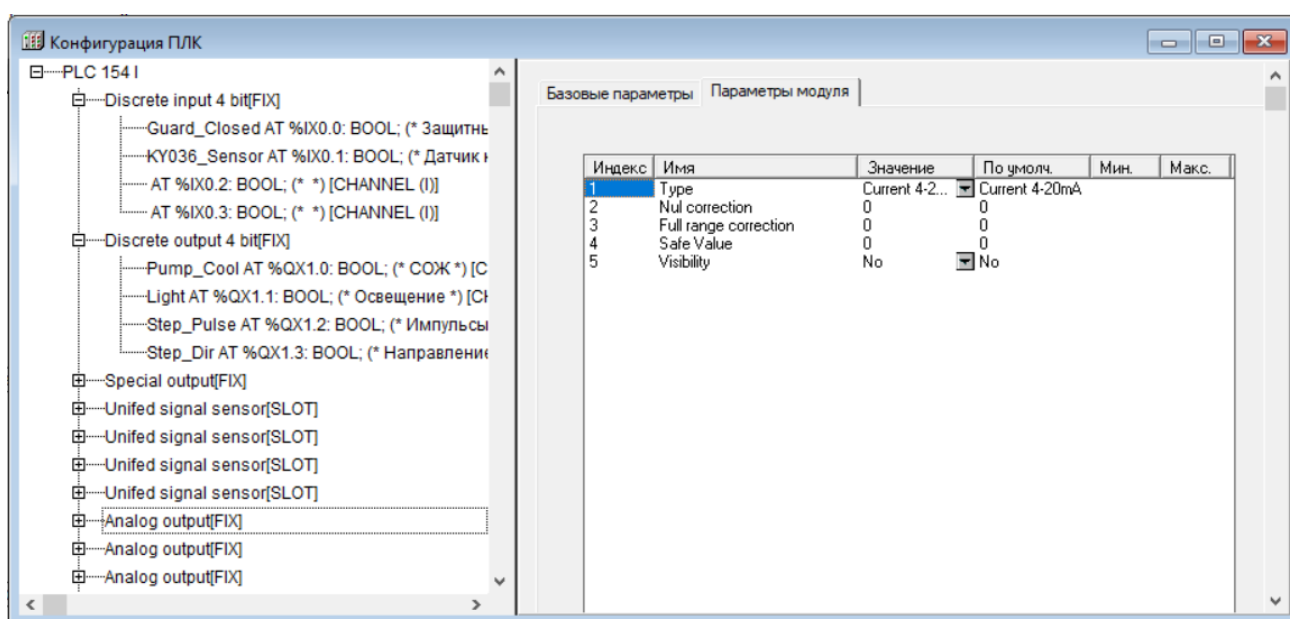


Рисунок 28 – Конфигурация входов и выходов

Далее был создан и настроен модуль дистанционного ввода-вывода. В конфигурацию был добавлен «Universal Modbus device», добавлен подэлемент 8 bit input module. В нулевой бит назначена переменная Limit_Up, в нее записывается сигнал переднего концевого выключателя поперечной подачи. В первый второй и третий бит назначаются переменные Limit_Down, Limit_Left,

Limit_Right. В них записываются сигналы с заднего концевого выключателя поперечной подачи, с левого и правого коцевых выключателей продольной подачи соответственно. Конфигурация МДВВ представлена на рисунке 29.

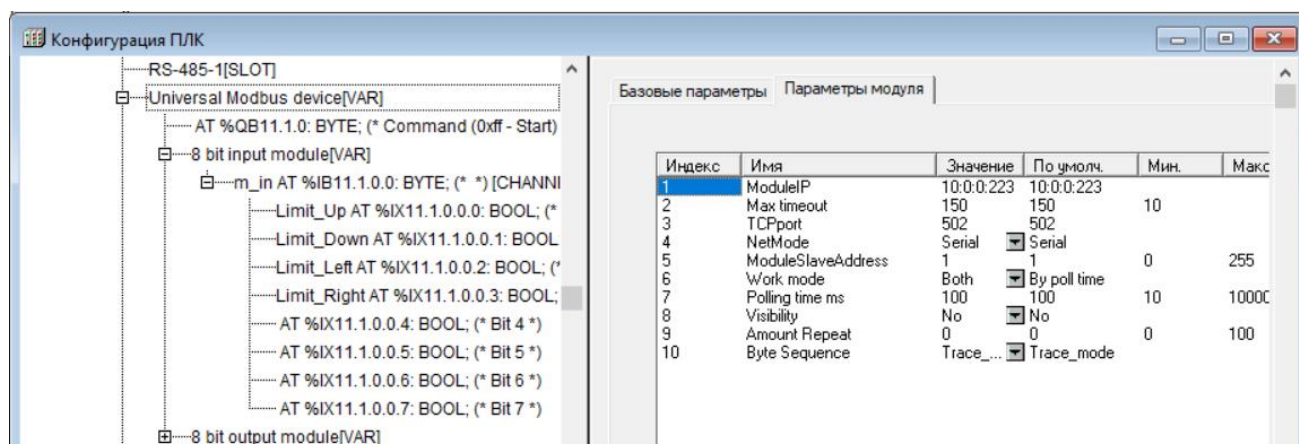


Рисунок 29 – Конфигурация МДВВ

После этого конфигурируются два частотных преобразователя. Были добавлены два «Universal Modbus Device». В них созданы следующие подэлементы: «Register output module», который записывает в регистр 2000h частотного преобразователя управляющие команды; Register output module», который записывает в регистр 2001h частотного преобразователя заданную частоту вращения; Register input module», который получает из регистра 2104h частотного преобразователя ток нагрузки. Конфигурация частотных преобразователей представлена на рисунках 30-32.

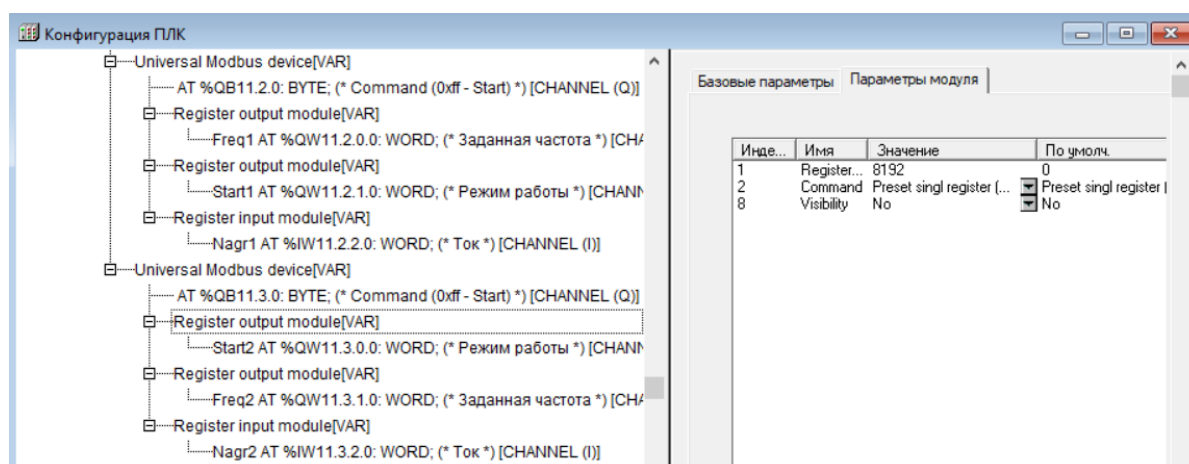


Рисунок 30 – Конфигурация преобразователей частоты

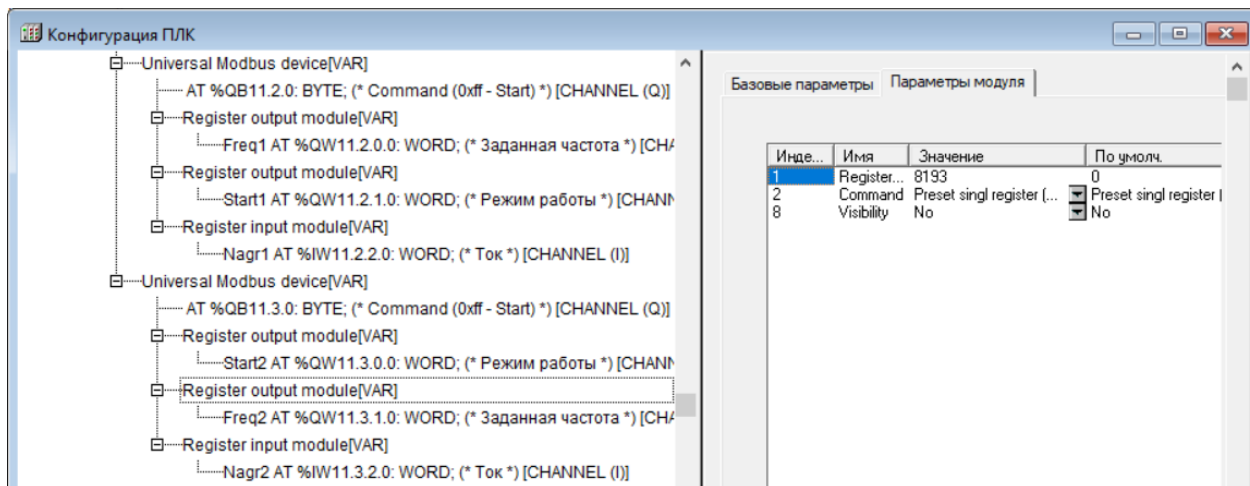


Рисунок 31 – Конфигурация преобразователей частоты

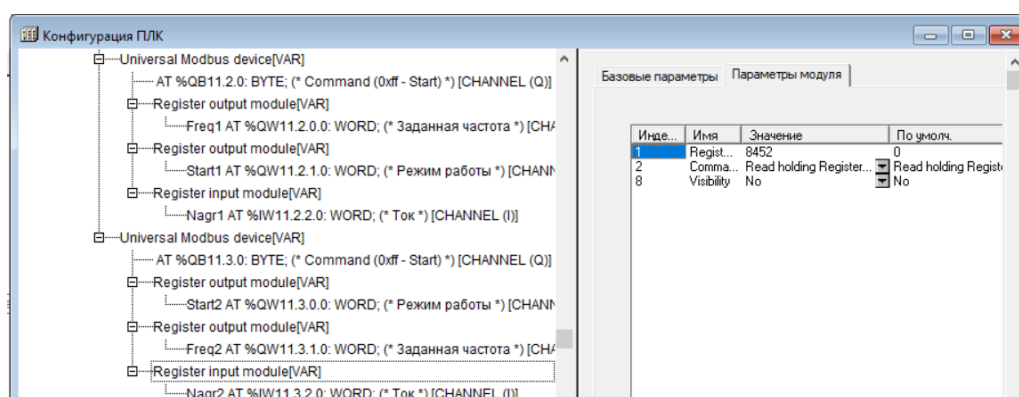


Рисунок 32 – Конфигурация преобразователей частоты

Все сетевые переменные, задействованные в проекте, собраны в таблице 3.

Таблица 3 – Сетевые переменные

Переменная	Тип переменной	Комментарий
Guard_Closed	BOOL	Защитный кожух
KY036_Sensor	BOOL	Датчик касания
Pump_Cool	BOOL	Насос СОЖ
Light	BOOL	Освещение
Step_Pulse	BOOL	Импульсы шагового мотора
Step_Dir	BOOL	Направление шагового мотора
Limit_Up	BOOL	Концевик подачи

Переменная	Тип переменной	Комментарий
Limit_Down	BOOL	Концевик подачи
Limit_Left	BOOL	Концевик подачи
Limit_Right	BOOL	Концевик подачи
Freq1	WORD	Заданная частота шпинделя
Start1	WORD	Управляющая команда шпинделя
Nagr1	WORD	Ток двигателя шпинделя
Freq2	WORD	Заданная частота подачи
Start2	WORD	Управляющая команда подачи
Nagr2	WORD	Ток двиг. Подачи

1.2.2 Разработка управляющей программы

Для обеспечения ввода и вывода данных, проведения вычислений различных параметров, были созданы глобальные переменные. Они используются внутри управляющей программы и не связаны с физическими входами и выходами контроллера. Полный список глобальных переменных с их типами и комментариями представлен на рисунках 33-34.

```

(*Параметры общие*)
Err_Msg:STRING;    (*Сообщение об ошибке*)
Err_Ind:BOOL;      (*Индикатор ошибки*)
Nagr_Danger: WORD:=2500;    (*Перегрузка по току*)
Diameter:REAL;     (*Диаметр заготовки*)
Radius:REAL; (*Радиус заготовки*)
Current_Depth :REAL;    (*Продольная подача в моменте*)
Current_Feed: REAL;     (*Поперечная подача в моменте*)
Pulse_Counter : INT;    (*Счетчик импульса*)
bPulse_Counter : WORD;  (*Счетчик импульса*)
reset: BOOL          ;(*Резет*)
LeadScrewPitch: WORD;   (*Шаг ходового винта*)

(*флаги*)
bHomingMode : BOOL; (*Флаг режима устанок нуля*)
bFacingMode : BOOL;  (*Флаг режима торцевания*)
bTurningMode : BOOL; (*Флаг режима обработки*)
bThreadingMode: BOOL; (*Флаг режима резьбы*)
bMetric:BOOL;    (*Флаг метрической резьбы*)
blnch:BOOL; (*Флаг дюймовой резьбы*)
bFacingComplete : BOOL; (*Флаг завершения торцевания*)
bTurningComplete : BOOL; (*Флаг завершения обработки*)
bThreadingComplete : BOOL; (*Флаг завершения резьбы*)

(*параметры обработки*)
(* Торцевание *)
Facing:BOOL;    (*Кнопка торцевания*)
Facing_Feed : REAL;    (*Длина торцевания*)

(* Обработка поверхности *)
Turning:BOOL;    (*Кнопка обработки*)
Turning_Feed : REAL;    (*Длина обработки*)
Turning_Depth :REAL;    (*Глубина обработки*)

```

Рисунок 33 – Глобальные переменные

```

(*Нарезка резьбы*)
Threading: BOOL;      (*Кнопка резьбы*)
indMetric: BOOL;      (*Индикатор метрической резьбы*)
indInch: BOOL;      (*Индикатор дюймовой резьбы*)
ThreadType : INT;      (* 0-метрическая ГОСТ, 1-дюймовая UNC/UNF *)
ThreadLength : REAL;      (* Длина резьбы в мм *)
Pitch_mm : REAL;      (* Шаг резьбы в мм *)
Depth_mm : REAL;      (* Глубина резьбы в мм *)
rTPI : REAL; (*расчет дюймовой резьбы*)
rFreq1: REAL; (* расчет частоты двиг. шпинделя*)
rFreq2: REAL; (*Расчет частоты двиг. подачи*)

(*позиционирование*)
GoUp: BOOL;      (*Кнопка вверх*)
GoDown: BOOL;      (*Кнопка вниз*)
GoLeft: BOOL;      (*Кнопка влево*)
GoRight: BOOL;      (*Кнопка вправо*)

(*управление шаговым двигателем*)
Start_Generator : BOOL; (*запуск генератора импульсов*)

(*вспомогательные*)
tCurrent_Depth : REAL; (*расчет глубины обработки Turning*)
thCurrent_Depth : REAL; (*расчет глубины обработки Threading*)
CurrentPos_Cross : REAL; (*Корректный вывод глубины обработки*)
CurrentPos_Long : REAL; (*Корректный вывод длины обработки*)
Homing_Dis: BOOL; (*Запрет ввода Установка нуля*)
Facing_Dis: BOOL; (*Запрет ввода Торцевание*)
Turning_Dis: BOOL; (*Запрет ввода Обработка поверхности*)
Threading_Dis: BOOL; (*Запрет ввода Резьба*)

```

Рисунок 34 – Глобальные переменные. Продолжение

Управляющая программа разделена на макросы. Так же в управляющей программе присутствуют пользовательские функциональные блоки. Полный вид управляющей программы представлен на рисунках 35-40.

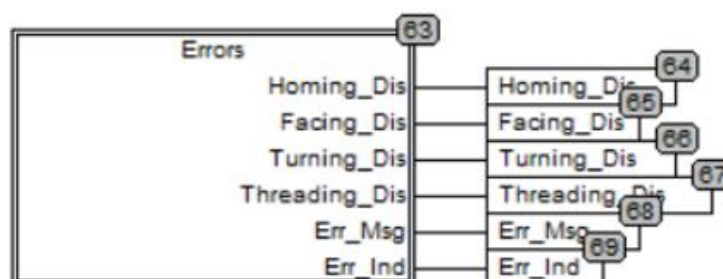
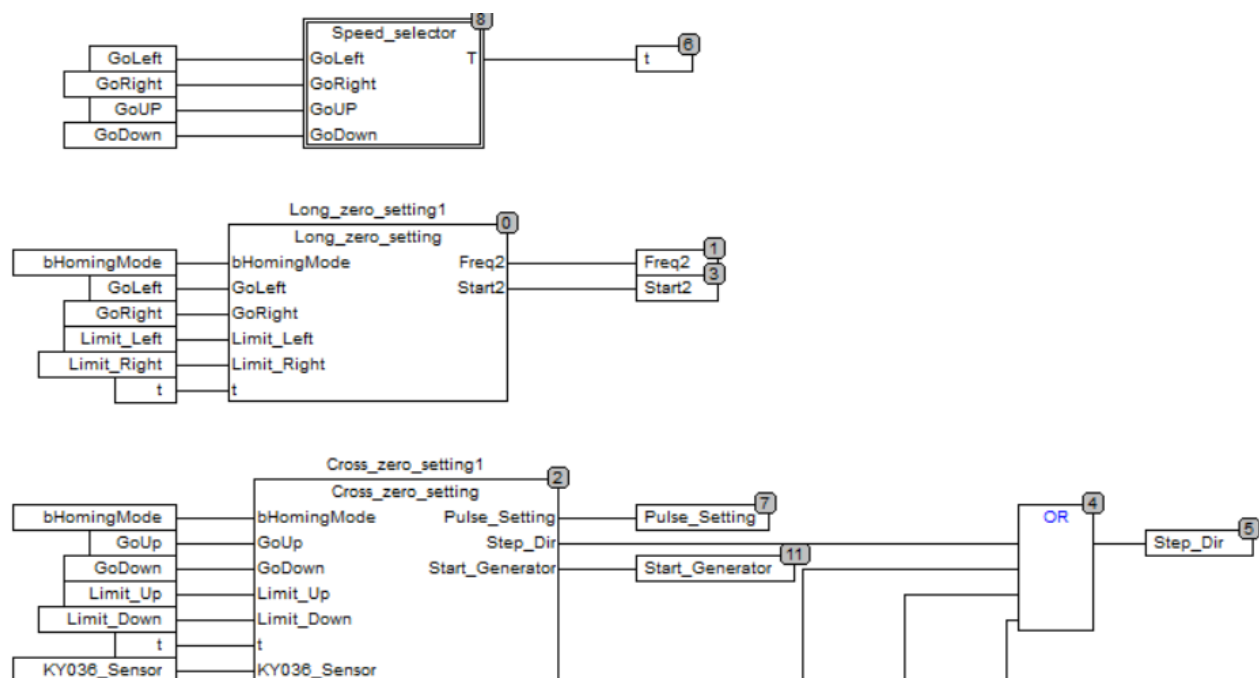


Рисунок 35 – Управляющая программа



К выходам макроса Facing,
выходам функциональных
блоков Cross_Steps1,
Cross_Steps2

Рисунок 36 – Управляющая программа. Продолжение

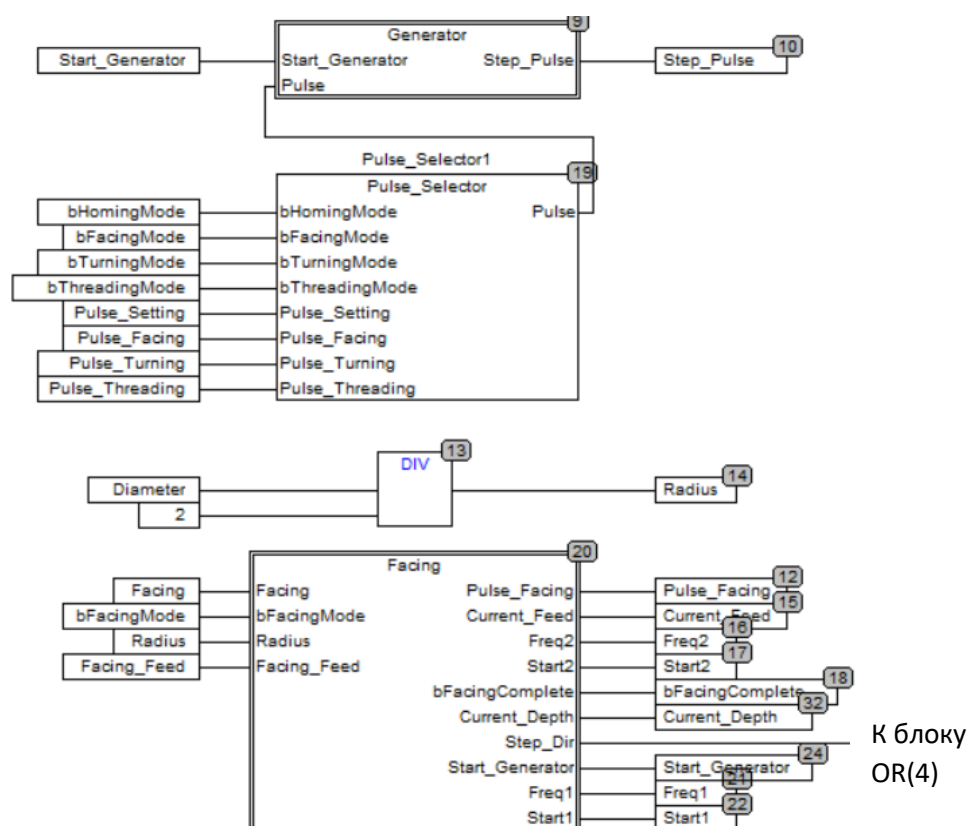


Рисунок 37 – Управляющая программа. Продолжение

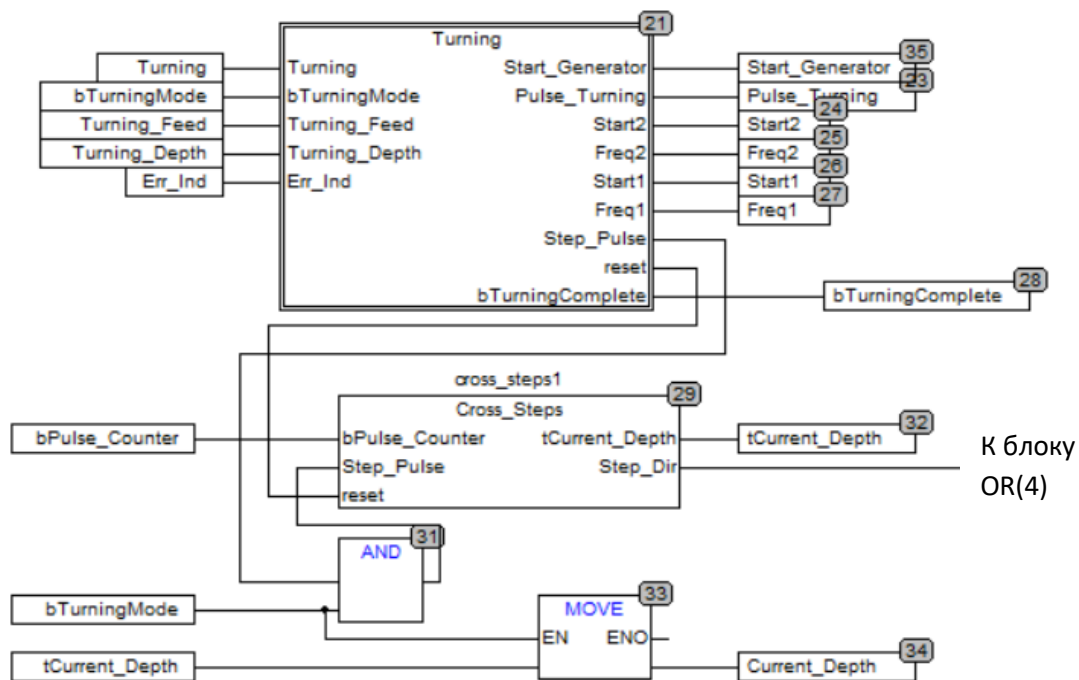


Рисунок 38 – Управляющая программа. Продолжение

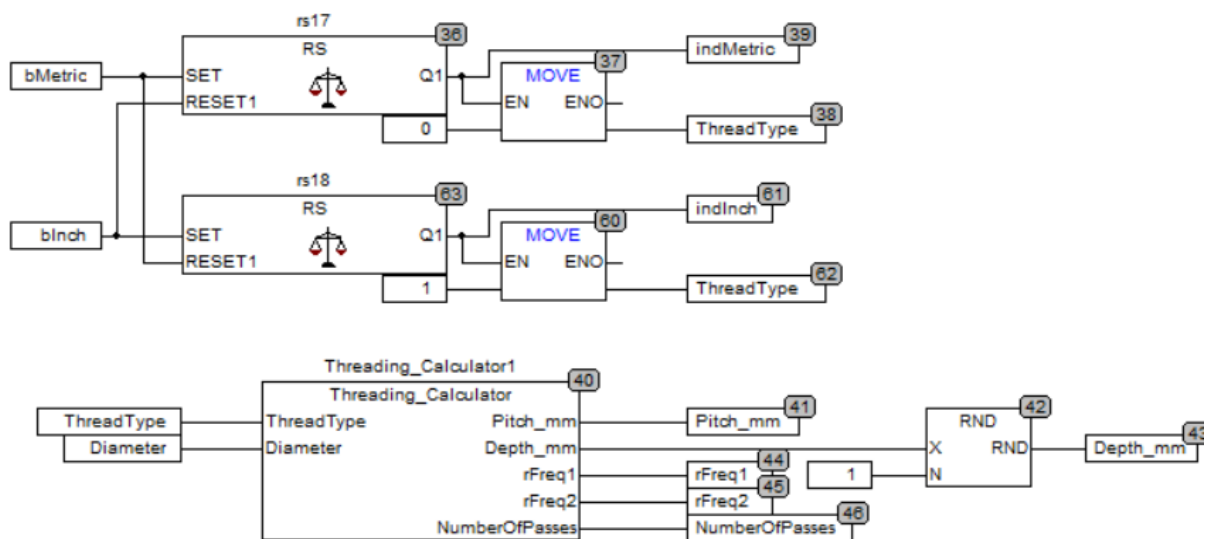


Рисунок 39 – Управляющая программа. Продолжение

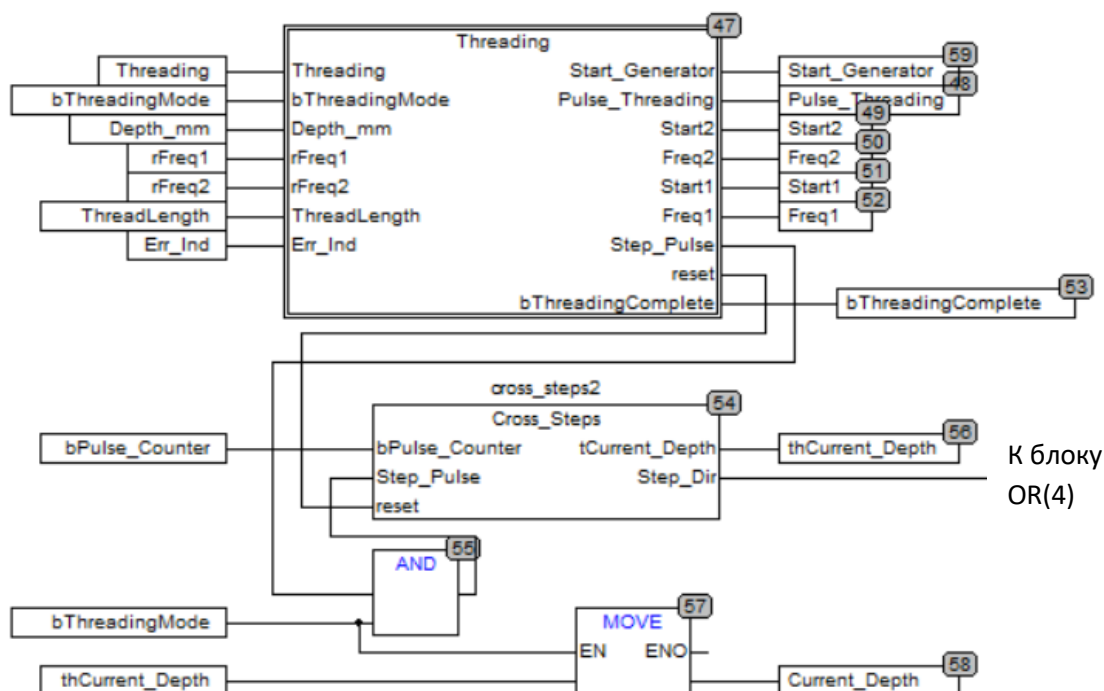


Рисунок 40 – Управляющая программа. Продолжение

Макрос «Errors», при одном выбранном режиме работы, устанавливает запрет ввода любого из остальных режимов, а так же выводит на панель операторов сообщения об ошибках при их наличии и управляет индикатором Ошибка, осуществляет контроль безопасности.

Макрос «Speed_Selector» подсчитывает количество нажатий на кнопки ручного управления подачей (кнопки Влево, Вправо, Вперед, Назад) и определяет режим ее работы: по первому нажатию задается малая скорость подачи, по второму нажатию – ускоренная подача. Последующие нажатия, начиная с третьего, меняют режимы циклически.

Функциональный блок «Long_zero_setting» управляет двигателем продольной подачи в режиме Установка нулевой точки. В зависимости от нажатия на кнопки ручного управления подачей (Вправо или Влево), функциональный блок подает команды на частотный преобразователь, включает прямое направление вращения или обратное. В зависимости от режима подачи, полученного на выходе макроса «Speed_Selector», Функциональный блок «Long_zero_setting» подает на

частотный преобразователь заданную частоту, малую или увеличенную соответственно.

Функциональный блок «Cross_zero_setting» управляет шаговым двигателем поперечной подачи в режиме Установка нулевой точки. В зависимости от нажатия на кнопки ручного управления подачей (Вперед или Назад), функциональный блок подает команды драйвер шагового мотора, включает прямое направление вращения или обратное. В зависимости от режима подачи, полученного на выходе макроса «Speed_Selector», Функциональный блок «Long_zero_setting» подает на функциональный блок «Pulse_Selector» (задает интервалы генератору импульсов) время, соответствующее интервалу между импульсами. Длинный временной промежуток для малой скорости подачи, короткий – для увеличенной скорости.

Макрос «Generator» Принимает на вход временной интервал от функционального блока «Pulse_Selector», генерирует импульсы по интервалу и передает на драйвер шагового двигателя управления поперечной подачей суппорта.

Функциональный блок «Pulse_Selector» принимает на вход временные промежутки от макросов «Cross_zero_setting», «Facing», «Turning», «Threading» и, в зависимости от выбранного режима работы, передает в «Generator» нужный интервал.

Макрос «Facing» принимает на вход радиус заготовки и длину обработки торца, задаваемые оператором, управляет двигателем продольной подачи и шаговым мотором поперечной подачи, выполняет торцевание заготовки по заданным параметрам.

Функциональный блок «Cross_Steps» управляет двигателем поперечной подачи по заранее определенному сценарию во время работы режимов Обработка поверхности и Резьба.

Макрос «Turning» принимает на вход глубину и длину обработки поверхности заготовки, задаваемые оператором, управляет двигателем продольной подачи и шаговым мотором поперечной подачи взаимодействуя с функциональным блоком «Cross_Steps», выполняет обработку цилиндрической поверхности заготовки по заданным параметрам.

Функциональный блок «Threading_Calculator» принимает на вход диаметр заготовки и тип резьбы, рассчитывает шаг и глубину резьбы, рассчитывает и синхронизирует частоту электродвигателей шпинделя и подачи, передает эти параметры в макрос «Threading».

Макрос «Threading» принимает на вход размер и длину резьбы, вводимые оператором и все параметры, рассчитанные в функциональном блоке «Threading_Calculator». Макрос управляет двигателем продольной подачи и шаговым мотором поперечной подачи взаимодействуя с функциональным блоком «Cross_Steps», выполняет нарезание резьбы по заданным параметрам.

					ПГУ 1.090501.05.001 ПЗ	Лист
						46
Изм.	Лист	№ докум.	Подпись	Дата		

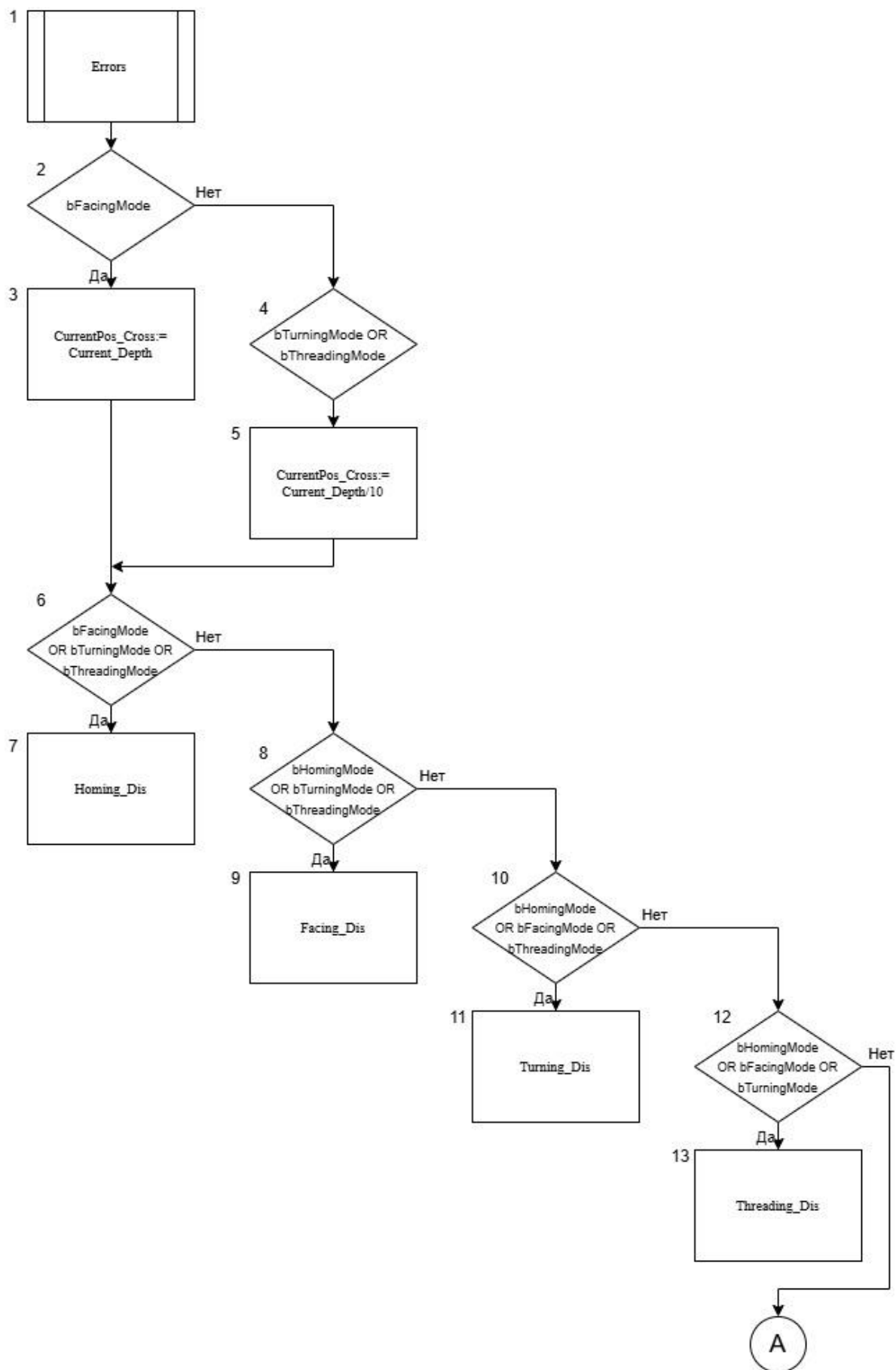


Рисунок 41 – Схема макроса Errors

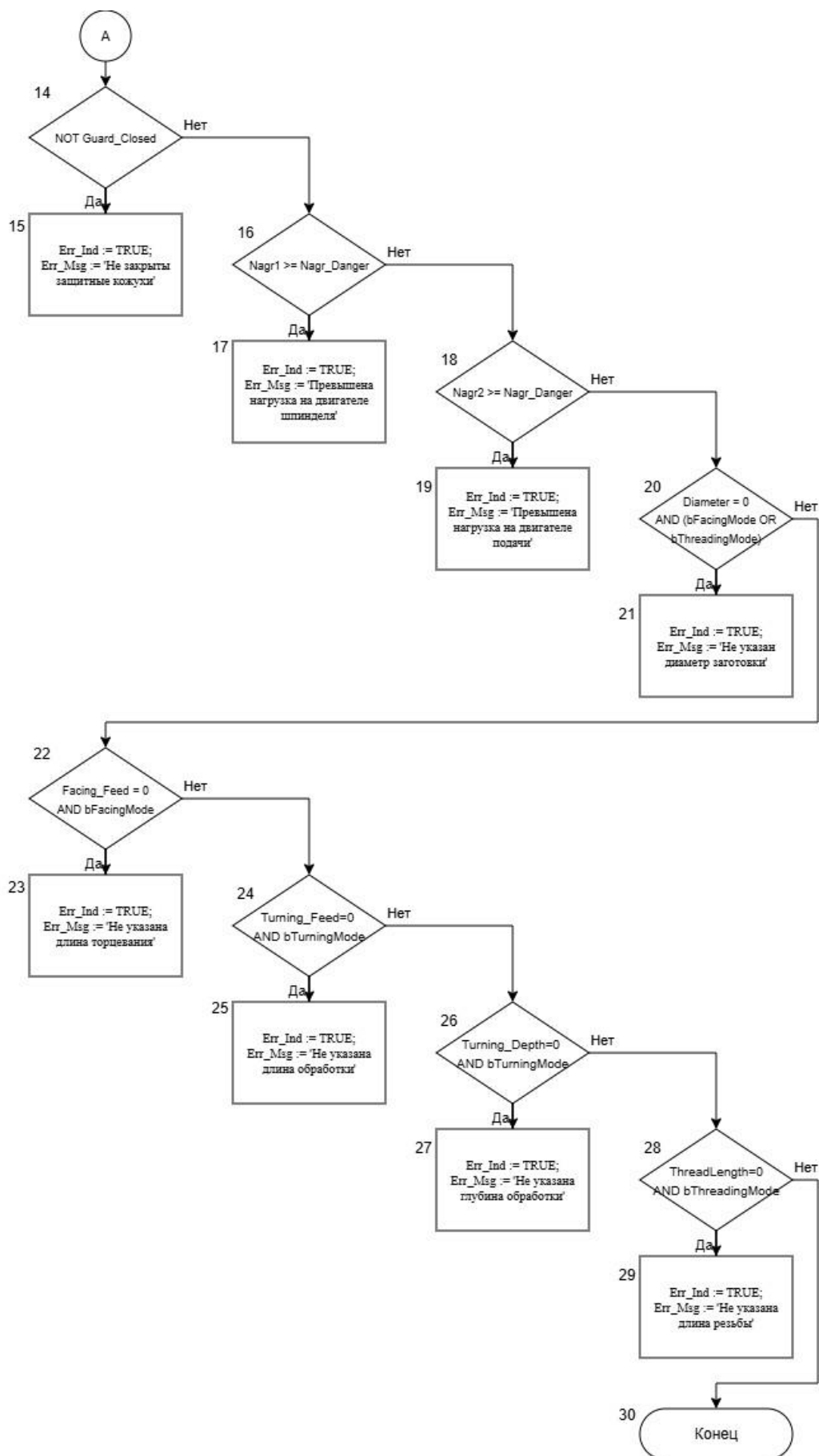


Рисунок 42 – Схема макроса Errors. Продолжение

На рисунках 41 – 42 представлена схема макроса Errors. Его функцией является обеспечение корректности вывода данных в специальные поля панели оператора. На всем протяжении работы программы при расчетах позиции продольной и поперечной подач в настоящий момент времени, все значения умножаются на 10. Это необходимо для избежания накопленной ошибки при многократном умножении десятичных дробей. Для того, чтобы значения, выводимые в панели оператора, соответствовали реальным позициям подач, значения, полученные в ходе работы программы, делятся на 10, результата заносится в переменные CurrentPos_Cross и CurrentPos_Long для поперечной и продольной подачи соответственно, после чего выводятся в панель оператора. Фрагмент кода, выполняющий эту функцию, представлен на рисунке 43.

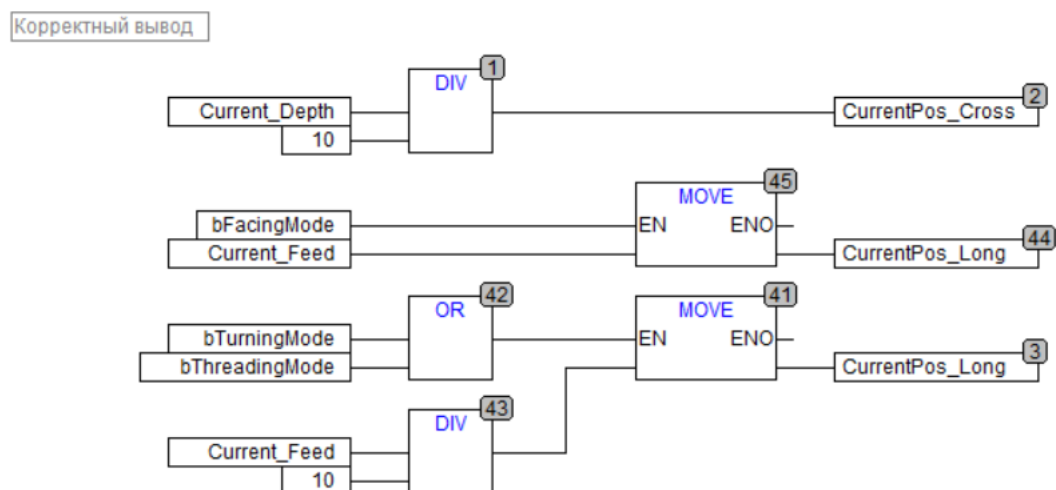


Рисунок 43 – Корректный вывод значений в настоящий момент времени

Далее обрабатываются запреты ввода режимов. Они служат для того, чтобы при включенном одном из режимов работы, оператор не имел возможности включить другой режим работы. Фрагмент кода представлен на рисунке 44. Таким образом, если переменная bFacingMode или bTurningMode или bThreadingMode обращается в логическую единицу (то есть выбран один из перечисленных режимов), в переменную Homing_Dis выводится логическая единица. Если одна из переменных bHomingMode или bTurningMode или bThreadingMode обращается в

логическую единицу, переменная Facing_Dis получает значение TRUE, что соответствует запрету на ввод. Аналогичным образом работает запрет ввода, организованный с помощью переменных Turning_Dis и Threading_Dis.

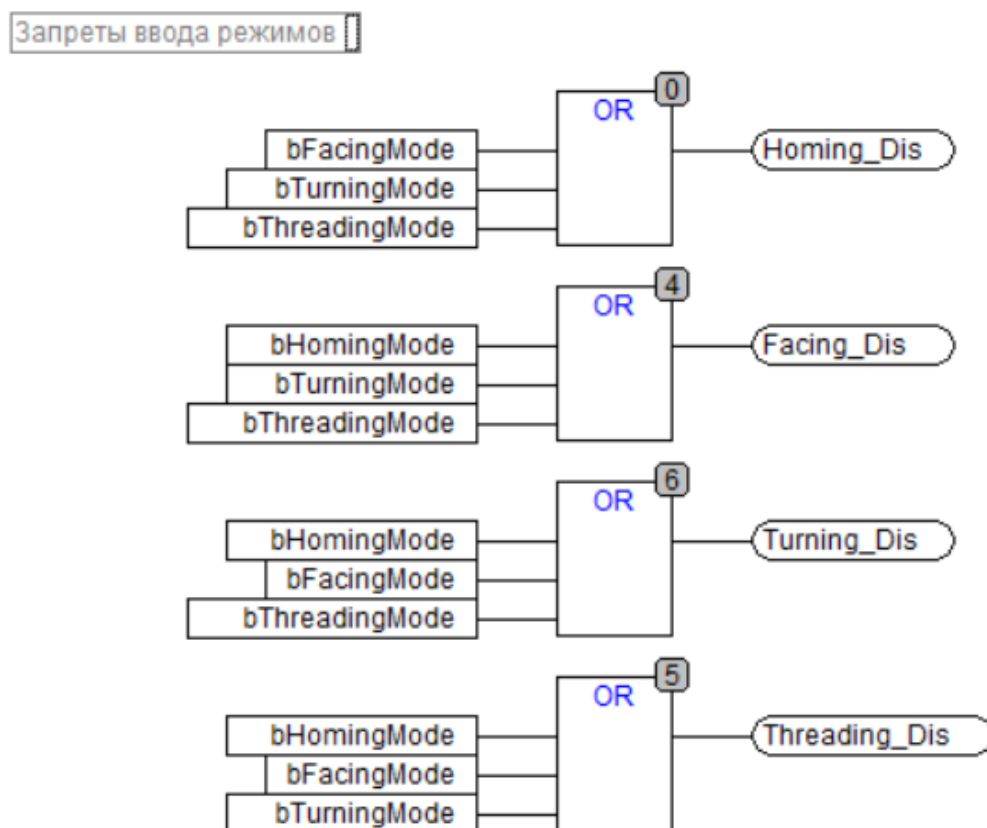


Рисунок 44 – Запрет ввода режимов

Переменные Homing_Dis, Facing_Dis, Turning_Dis, и Threading_Dis вносятся в поле «Запрет ввода» в окнах настройки кнопок интерфейса, отвечающих за выбор режима работы (рисунки 45-48).

Конфигурирование элемента (#1) X

Категория	Переменные	
Форма	Невидимость:	<input type="text"/>
Текст	Запрет ввода:	<input type="text" value=".Homing_Dis"/>
Вид текста	Изм. цвета:	<input type="text" value=".bHomingMode"/>
Ширина линии	Выв_текста:	<input type="text"/>
Цвета	Conversion base:	<input type="text"/>
Переменные цвета	Conversion factor:	<input type="text"/>
Положение	Подсказка: display:	<input type="text"/>
Деформация		
Переменные		
Ввод		
Текст подсказки		
Безопасность		
Программируемость		

Рисунок 45 – Настройка запрета ввода режима

Конфигурирование элемента (#6) X

Категория	Переменные	
Форма	Невидимость:	<input type="text"/>
Текст	Запрет ввода:	<input type="text" value=".Facing_Dis"/>
Вид текста	Изм. цвета:	<input type="text" value=".bFacingMode"/>
Ширина линии	Выв_текста:	<input type="text"/>
Цвета	Conversion base:	<input type="text"/>
Переменные цвета	Conversion factor:	<input type="text"/>
Положение	Подсказка: display:	<input type="text"/>
Деформация		
Переменные		
Ввод		
Текст подсказки		
Безопасность		
Программируемость		

Рисунок 46 – Настройка запрета ввода режима

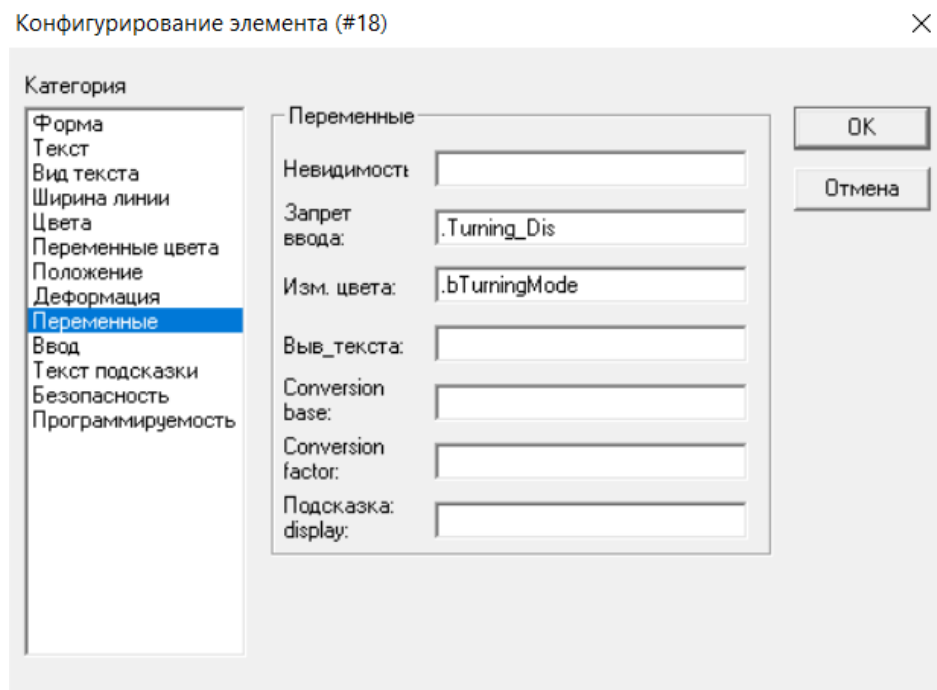


Рисунок 47 – Настройка запрета ввода режима

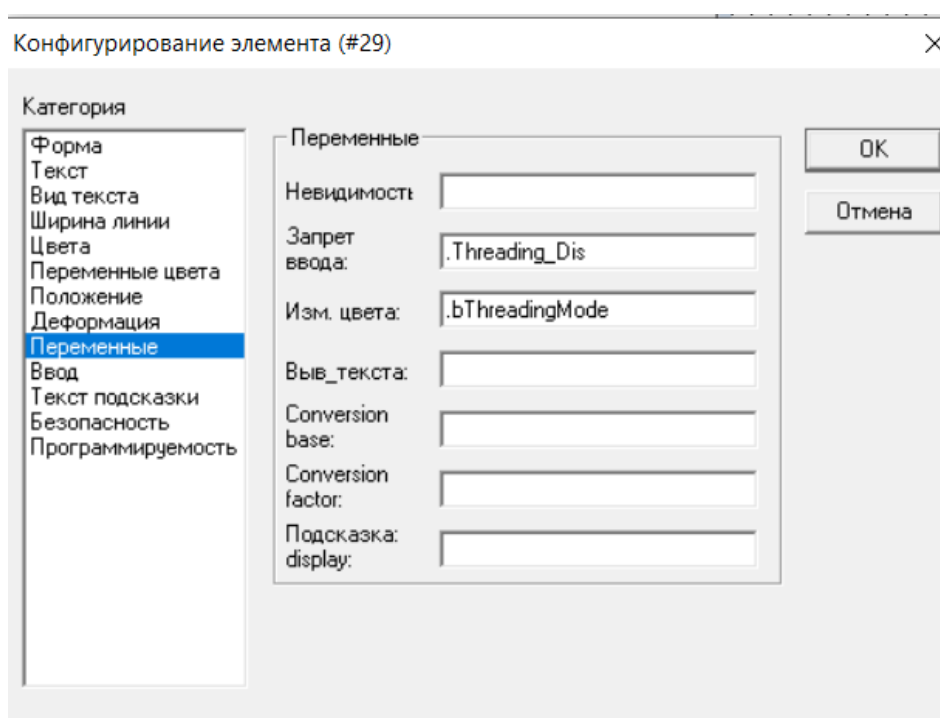


Рисунок 48 – Настройка запрета ввода режима

Далее происходит обработка сообщений об ошибках. Фрагменты кода представлены на рисунках 49-51.

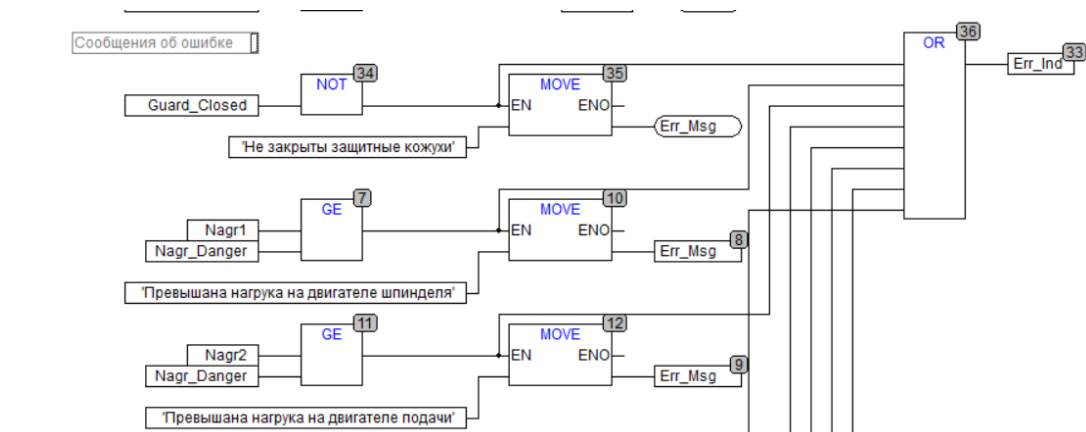


Рисунок 49 – Сообщения об ошибках

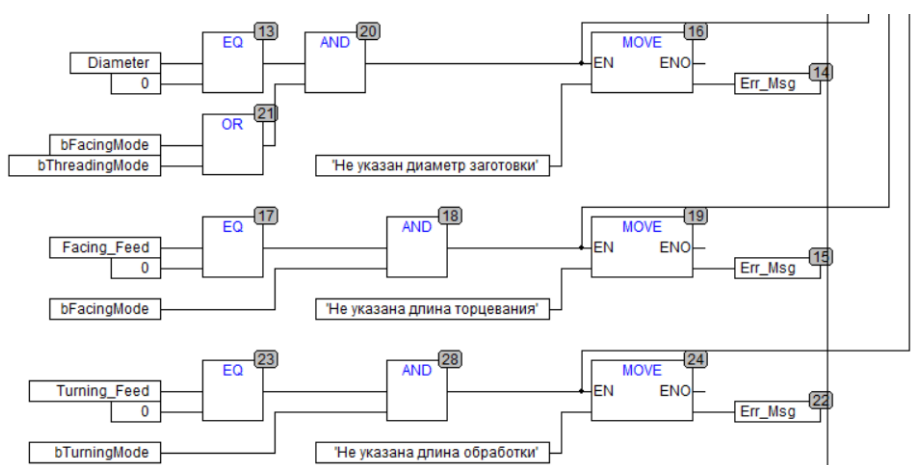


Рисунок 50 – Сообщения об ошибках

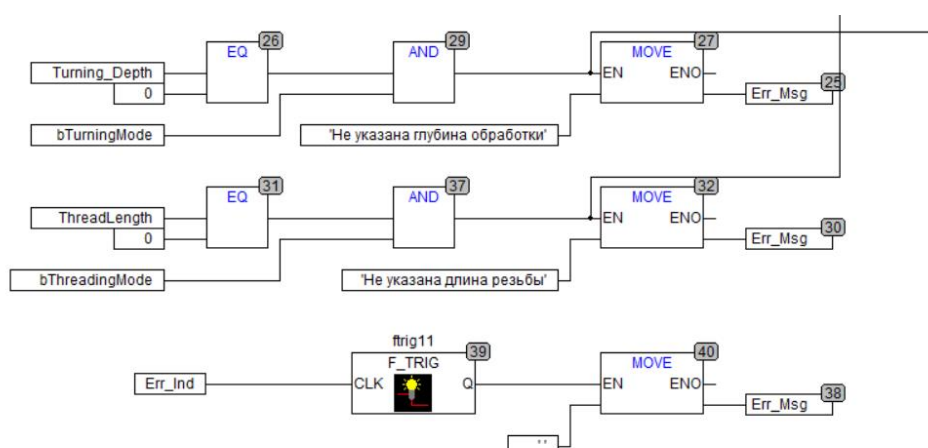


Рисунок 51 – Сообщения об ошибках

При отсутствии логической единицы в переменной Guard_Closed сигнал подается на переменную Err_Ind (включает индикатор ошибки) и выводит

сообщение «Не закрыты защитные кожухи». Далее переменная Err_Ind будет принимать положительное значение при возникновении каждой ошибки. Когда значение переменной Nagr1 превышает значение Nagr_Danger, выводится сообщение «Превышена нагрузка на двигателе шпинделя». Когда значение переменной Nagr2 превышает значение Nagr_Danger, выводится сообщение «Превышена нагрузка на двигателе Подачи». Таким же образом, с помощью оператора сравнения EQ и операторов AND и OR производится вывод сообщений «Не указан диаметр заготовки», «Не указана длина торцевания», «Не указана длина обработки», «Не указана глубина обработки», «Не указана длина резьбы». Также в поле сообщений об ошибке выводится пробел по спаду сигнала переменной Err_Ind, что соответствует исправлению ошибки, когда оператор исключает факторы ее возникновения.

В логике этих фрагментов управляющей программы используются такие логические элементы, как NOT – логическое отрицание, AND – логическое И. Также блок EQ, на выходе которого появляется сигнал, когда сигналы на обоих его входах равны, блок Move с входами EN и ENO. Когда на входе EN появляется единица, значение переменной на входе блока MOVE записывается в переменную на выходе этого блока. На выход ENO выводится то же самое значение, которое поступило на вход EN. Блок MOVE со входами EN и ENO представлены на рисунке 52. Также используется блок F_TRIG. Его выход генерирует импульс, когда на входе происходит спад сигнала. Блок F_TRIG представлен на рисунке 53.

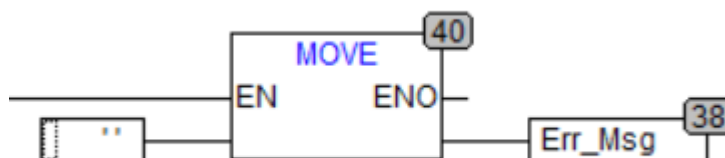


Рисунок 52 - Блок MOVE

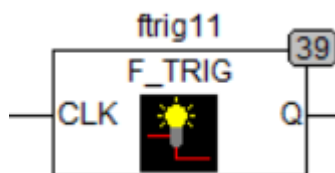


Рисунок 53 - Блок MOVE

Макрос Speed_selector управляет скоростью подачи каретки в режиме установки нулевой точки. Его схема представлена на рисунках 54-55. В этом режиме направление и перемещения подачи задатется оператором вручную с помощью интерфейсных кнопок, а скорость – количеством нажатий на кнопку. По первому нажатию устанавливается уменьшенная скорость, по второму – увеличенная. С третьего и последующих нажатий скорость меняется циклично. На рисунке 56 представлен фрагмент программы, который выполняет функцию установки флага направления перемещения подачи в зависимости от нажатой кнопки. Для установки используются блоки RS. При поступлении сигнала на вход SET блока RS, на выходе Q1 устанавливается логическая единица. Она сбрасывается только тогда, когда сигнал поступает на вход RESET блока RS.

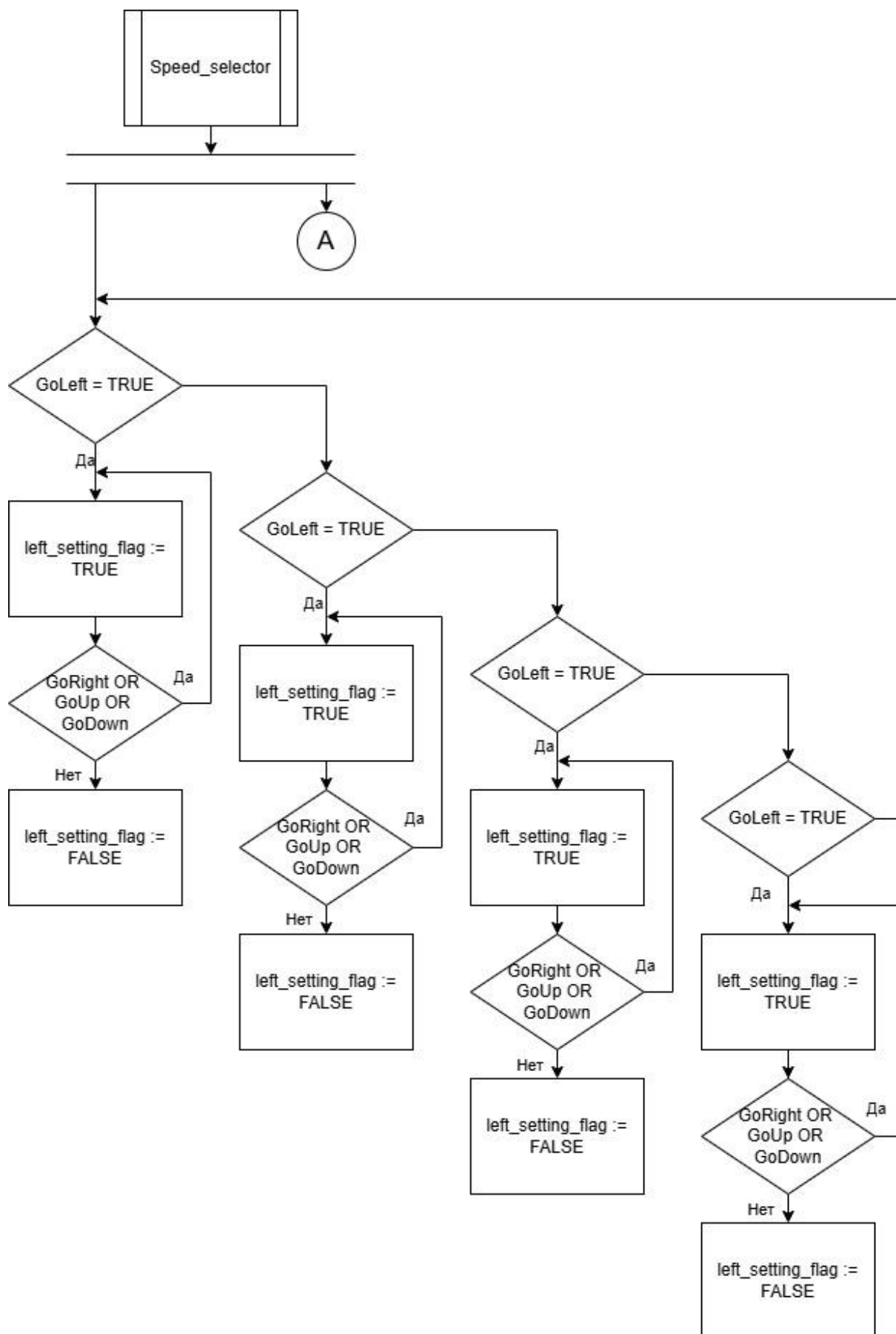


Рисунок 54 – Схема макроса Speed_Selector

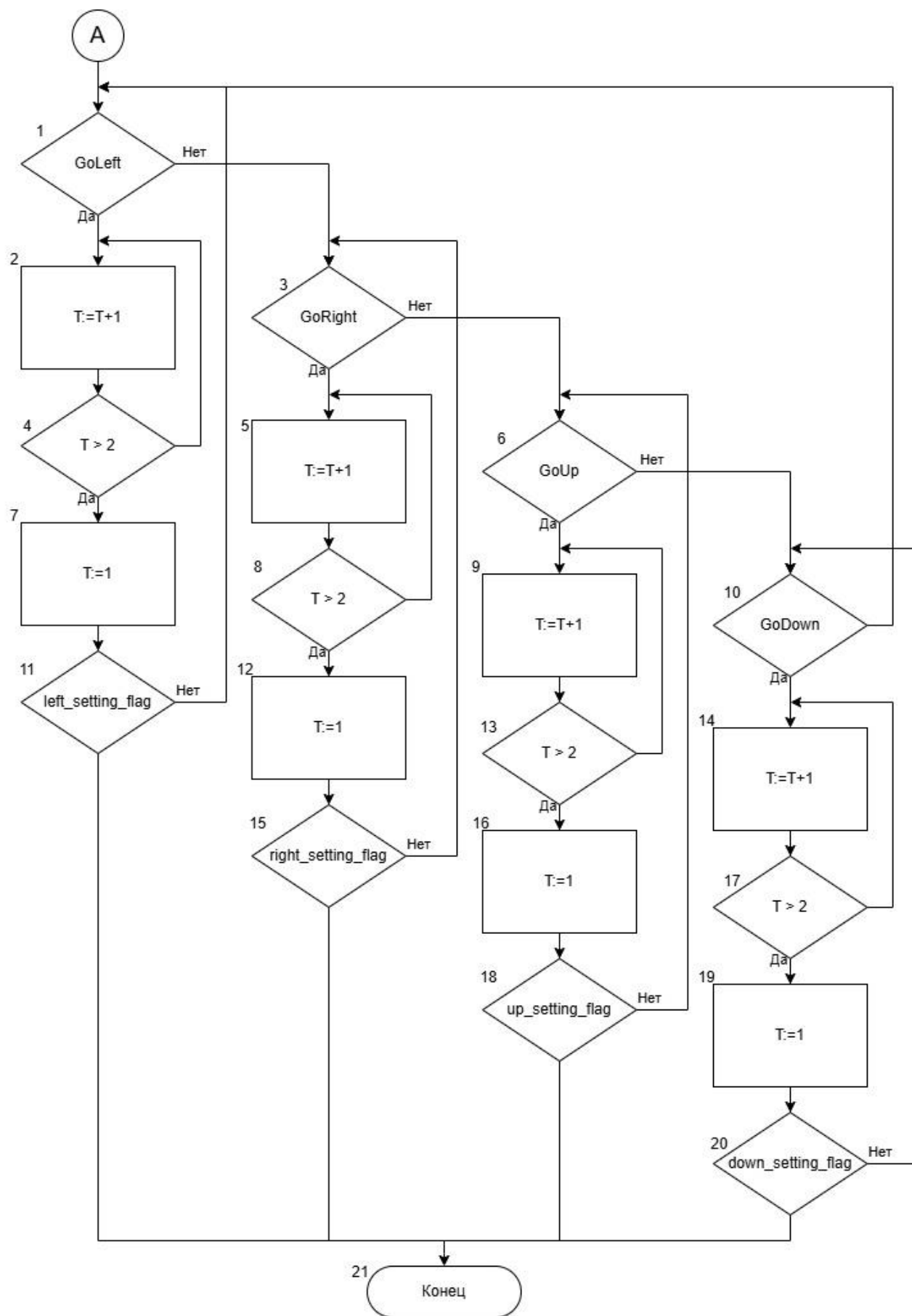


Рисунок 55 – Схема макроса Speed_Selector

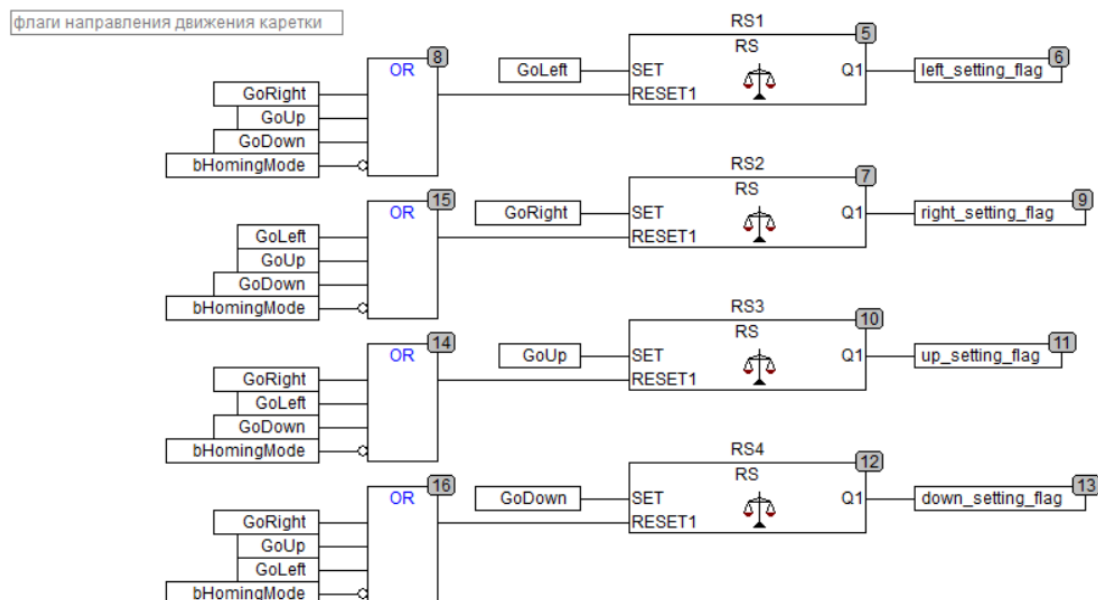


Рисунок 56 – Флаги направления

На рисунке 57 представлен фрагмент макроса, отвечающий за расчет скорости подачи. Инкрементные счетчики STU1 и STU2 ведут подсчет нажатий на кнопки. В момент, когда значение выхода CV счетчиков больше двух, на вход LOAD записывается число 1. Таким образом, при каждом последующем нажатии на кнопку, на выходе счетчика получается значение 1, либо 2, оно же записывается в переменную T. Для каждого из четырех счетчиков на входы RESET подключены переменные – флаги направления. Когда их значение становится равным логическому нулю, счетчики сбрасываются.

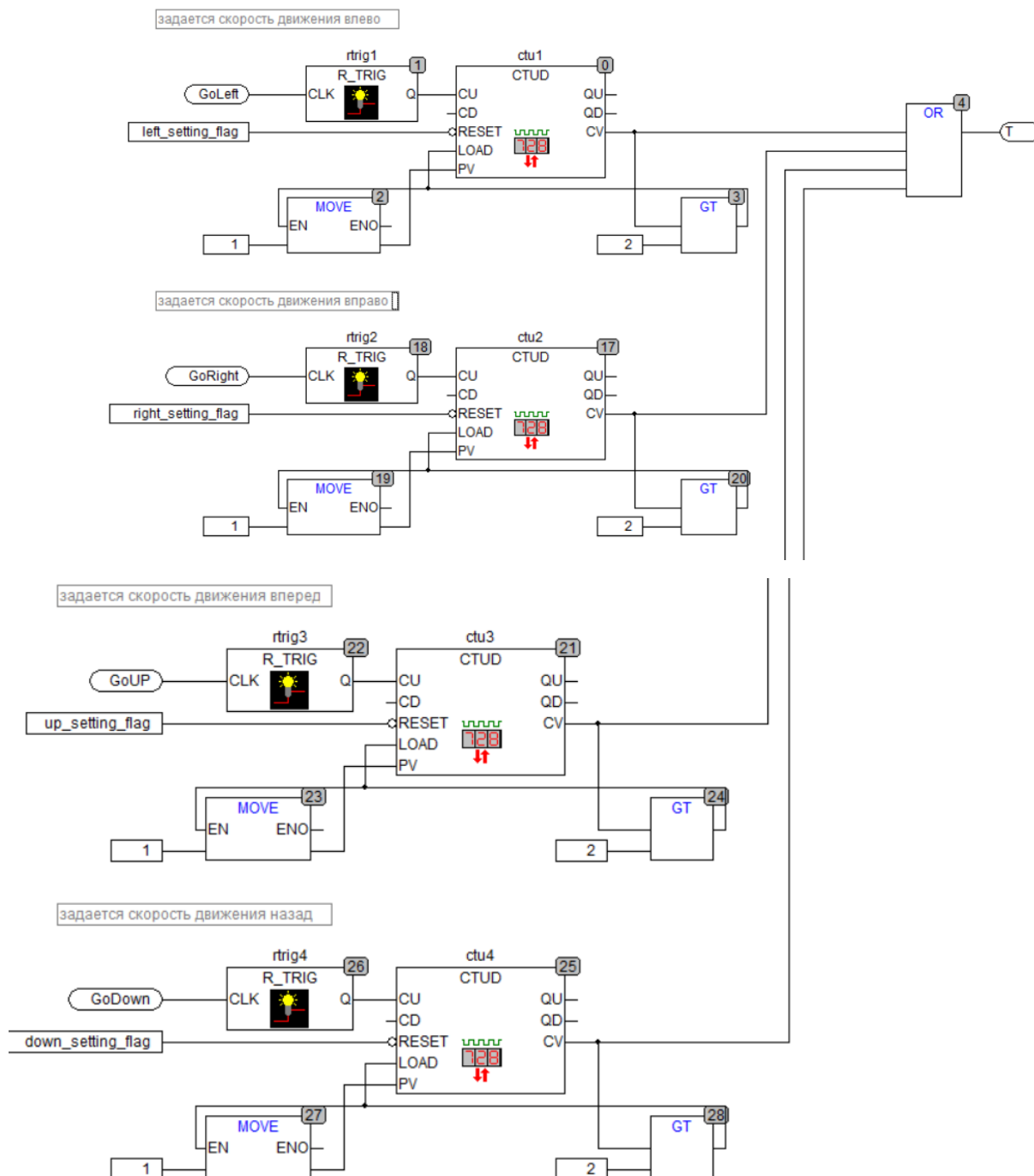


Рисунок 57 – Расчет скорости подачи

Продольное ручное перемещение суппорта регулируется при помощи функционального блока Long_zero_setting. Его схема представлена на рисунке 58.

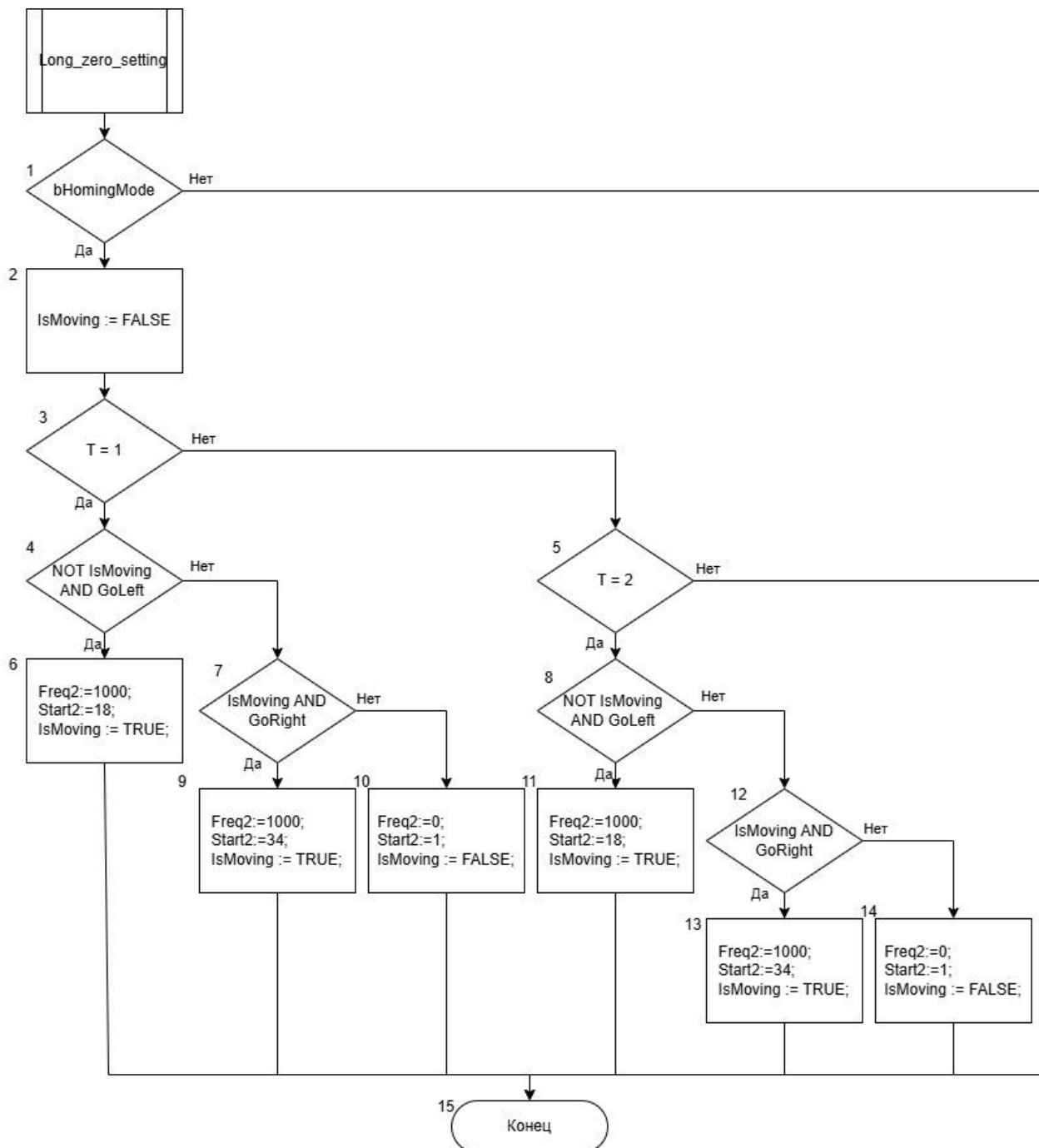


Рисунок 58 – Схема функционального блока Long_zero_setting

На вход этого функционального блока подается переменная T, значение которой было рассчитано при выполнении макроса Speed_Selector. Функциональный блок выполнен на основе оператора CASE OF. Условием оператора CASE является переменная T. Когда она равна 1, в переменную Freq2 записывается значение 1000. Когда T равно 2 – значение 5000. Это соответствует уменьшенной и увеличенной скорости подачи. При помощи условных операторов

задается направление вращения двигателя продольной подачи. Когда нажата кнопка «Влево» – 34.

Код функционального блока Long_zero_setting на языке ST:

```

IF bHomingMode THEN
    IsMoving := FALSE;
    IF NOT Limit_Left AND NOT Limit_Right THEN
CASE t OF
1:      IF NOT IsMoving AND GoLeft THEN
            Freq2:=1000;
            Start2:=18;
            IsMoving := TRUE;
        ELSIF NOT IsMoving AND GoRight THEN
            Freq2:=1000;
            Start2:=34;
            IsMoving := TRUE;
        ELSE
            Freq2 := 0;
            Start2:=1;
            IsMoving := FALSE;
        END_IF;
2:      IF NOT IsMoving AND GoLeft THEN
            Freq2:=5000;
            Start2:=18;
            IsMoving := TRUE;
        ELSIF NOT IsMoving AND GoRight THEN
            Freq2:=5000;
            Start2:=34;
            IsMoving := TRUE;
        ELSE
            Freq2 := 0;
            Start2:=1;
            IsMoving := FALSE;
        END_IF;
END_CASE;

```

					ПГУ 1.090501.05.001 ПЗ	Лист
						61
Изм.	Лист	№ докум.	Подпись	Дата		

```

ELSE
    Freq2 := 0;
    Start2:=8;
END_IF;
ELSE
    Freq2 := 0;
    Start2:=0;
END_IF;

```

Функциональный блок Cross_zero_setting регулирует поперечную подачу.
Его схема представлена на рисунке 59.

					ПГУ 1.090501.05.001 ПЗ	Лист
						62
Изм.	Лист	№ докум.	Подпись	Дата		

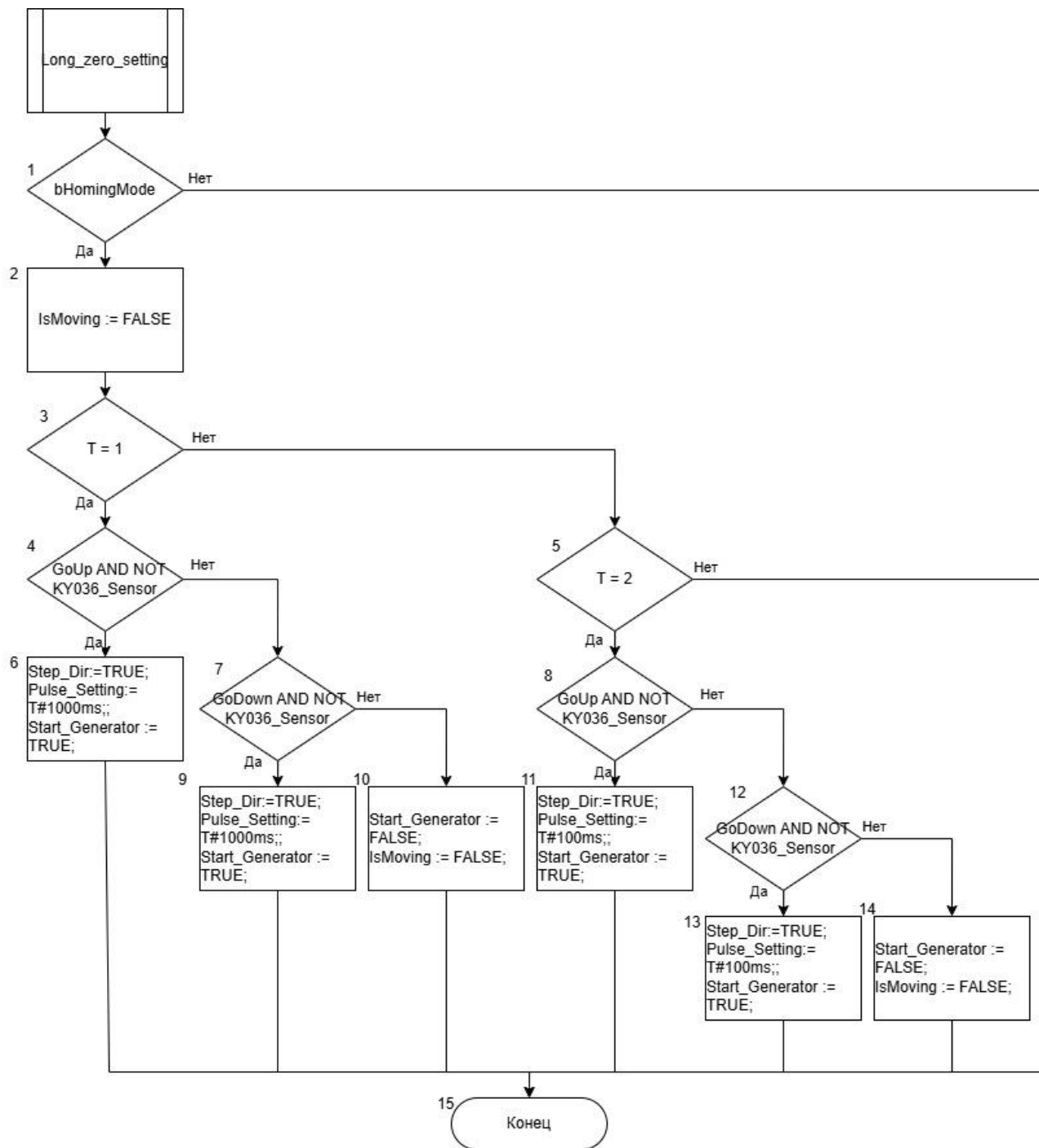


Рисунок 58 – Схема функционального блока Cross_zero_setting

Данный функциональный блок выполнен аналогично блоку Long_zero_setting, за исключением: При нажатии на кнопки сигнал передается в переменную Start_Generator, она запускает генератор импульсов для шагового двигателя поперечной подачи. В соответствии со значением переменной T, в переменную Pulse_Selector передается значение 1s или 100ms, что соответствует

уменьшенной и увеличенной скорости поперечной подачи. Направление вращения шагового двигателя задает переменная Step_Dir. Ниже представлен код функционального блока на языке ST.

```

IF bHomingMode THEN
    IsMoving := FALSE;
    IF NOT Limit_Up AND NOT Limit_Down THEN
CASE t OF
1:      IF NOT IsMoving AND GoUp AND NOT KY036_Sensor THEN
        Step_Dir:=TRUE;
        Pulse_Setting:=T#1000ms;
        IsMoving := TRUE;
        Start_Generator := TRUE;
    ELSIF NOT IsMoving AND GoDown THEN
        Step_Dir:=FALSE;
        Pulse_Setting:=T#1000ms;
        IsMoving := TRUE;
        Start_Generator := TRUE;
    ELSE
        IsMoving := FALSE;
        Start_Generator := FALSE;
    END_IF;
2:      IF NOT IsMoving AND GoUp AND NOT KY036_Sensor THEN
        Step_Dir:=TRUE;
        Pulse_Setting:=T#100ms;
        IsMoving := TRUE;
        Start_Generator := TRUE;
    ELSIF NOT IsMoving AND GoDown THEN
        Pulse_Setting:=T#100ms;
        Step_Dir:=FALSE;
        IsMoving := TRUE;
        Start_Generator := TRUE;
    ELSE
        IsMoving := FALSE;
        Start_Generator := FALSE;
    END_IF;

```

```

END_CASE;
END_IF;
ELSE
Step_Dir:=0;
END_IF

```

Макрос Generator воспроизводит на выходе импульсный сигнал для шагового двигателя. Его схема представлена на рисунке 59.

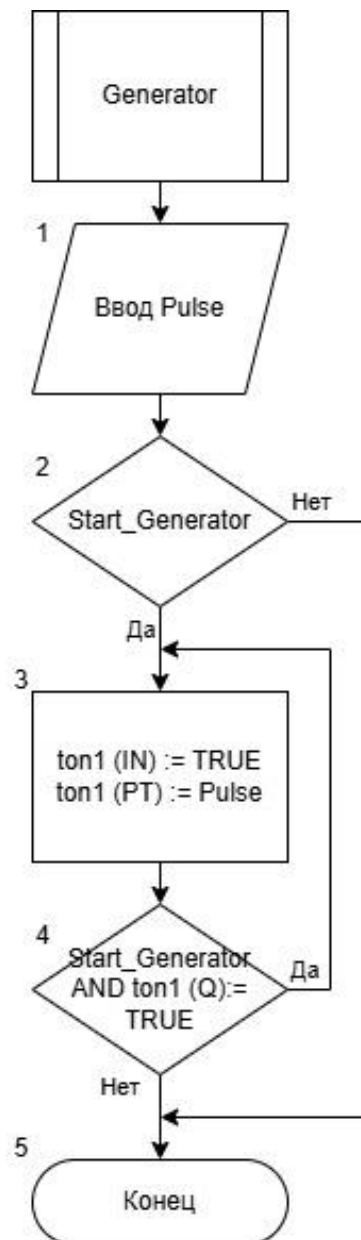


Рисунок 59 – Схема функционального блока Cross_zero_setting

При поступлении сигнала из переменной Start_Generator, генератор импульсов. Таймер с задержкой включению TON1 начинает работу, его временные промежутки задает переменная Pulse. Когда таймер заканчивает свою работу, сигнал с переменной Q через блок AND подается на вход таймера, и снова его запускает. По выходу ET через блок R_TRIG импульсные сигналы записываются в переменную Step_Pulse.

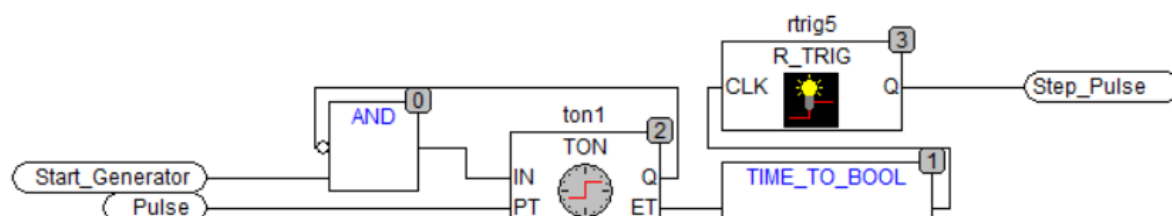


Рисунок 60 – Макрос Generator

Функциональный блок Pulse_Selector задает временные промежутки на вход макроса Generator. Его схема представлена на рисунке 61. При bHomingMode равной 1, переменная Pulse принимает значение переменной Pulse_Setting. При bFacingMode равной 1, переменная Pulse принимает значение переменной Pulse_Facing. При bTurningMode равной 1, переменная Pulse принимает значение переменной Pulse_Turning. При bThreadingMode равной 1, переменная Pulse принимает значение переменной Pulse_Threading.

Код функционального блока на языке ST:

```
IF bHomingMode THEN
    Pulse:=Pulse_Setting;
ELSIF bFacingMode THEN
    Pulse:=Pulse_Facing;
ELSIF bTurningMode THEN
    Pulse:=Pulse_Turning;
ELSIF bThreadingMode THEN
    Pulse:=Pulse_Threading;
END_IF
```

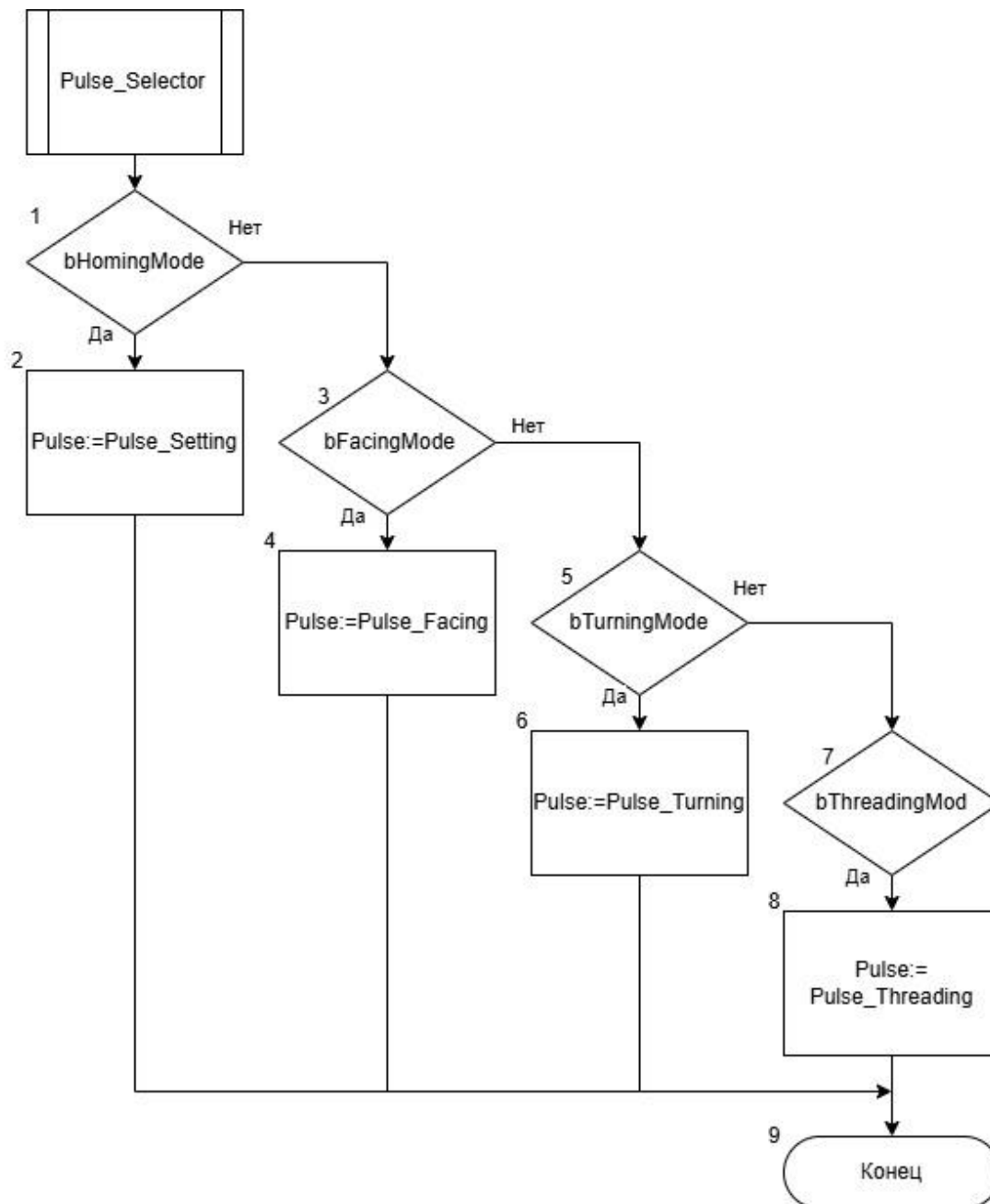


Рисунок 61 – Схема функционального блока Pulse_Selector

Макрос Facing управляет процессом обработки торца заготовки. На вход макроса подаются следующие параметры: радиус заготовки и длина обработки. Радиус вычисляется из переменной Diameter, вводимой оператором, ее значение делится на два и записывается в переменную Radius. Длина обработки записывается в переменную Facing_Feed.

Алгоритм работы функционального блока представлен на рисунке 62, его код на языке CFC представлен на рисунке 63. Процесс запускается, когда в

переменные bFacingMode и Facing записывается логическая единица. Начинает работу таймер TP1. Во время работы таймера в переменную Start2 записывается значение 18, а в переменную Freq2 – 100: двигатель продольной подачи вращается в прямом направлении с частотой 10 герц. Время, которое записывается на вход PT таймера TP1, рассчитывается исходя из шага ходового винта подачи, чтобы за один цикл работы таймера подача смещалась на 0.1 миллиметра. Переменная Current_Feed (длина обработки в настоящий момент времени) с каждым циклом работы таймера увеличивается на 0.1, эта информация будет выводиться на экран оператора. По завершении работы таймера в переменную Start2 записывается значение 1, а в переменную Freq2 – 0, команда на остановку двигателя подачи. В переменную Start_Generator записывается логическая единица, в этот момент запускается генератор импульсов шагового двигателя поперечной подачи. В переменную Step_Dir записывается единица, что соответствует прямому направлению вращения шагового двигателя. По каждому импульсу генератора, переменная Current_Depth (глубина обработки в настоящий момент времени) увеличивается на единицу. Когда значение переменной Current_Depth станет равным значению переменной Radius, переменная Step_Dir получает значение ноль, шаговый двигатель вращается в обратном направлении, а переменная Current_Depth уменьшается на единицу по каждому импульсу генератора. Далее процесс обработки циклически повторяется, пока значение переменной Current_Feed не станет равным значению Facing_Feed. После этого процесс будет остановлен.

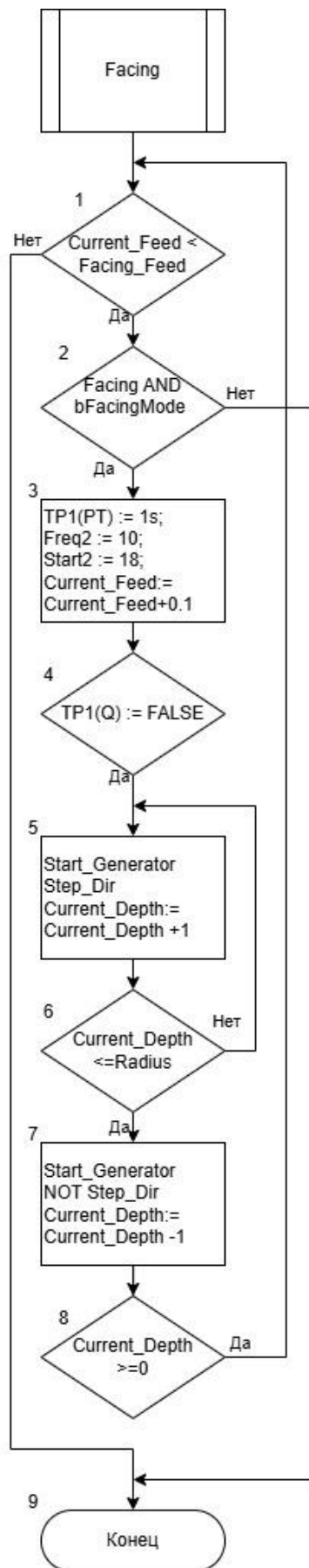


Рисунок 62 – Схема макроса Facing

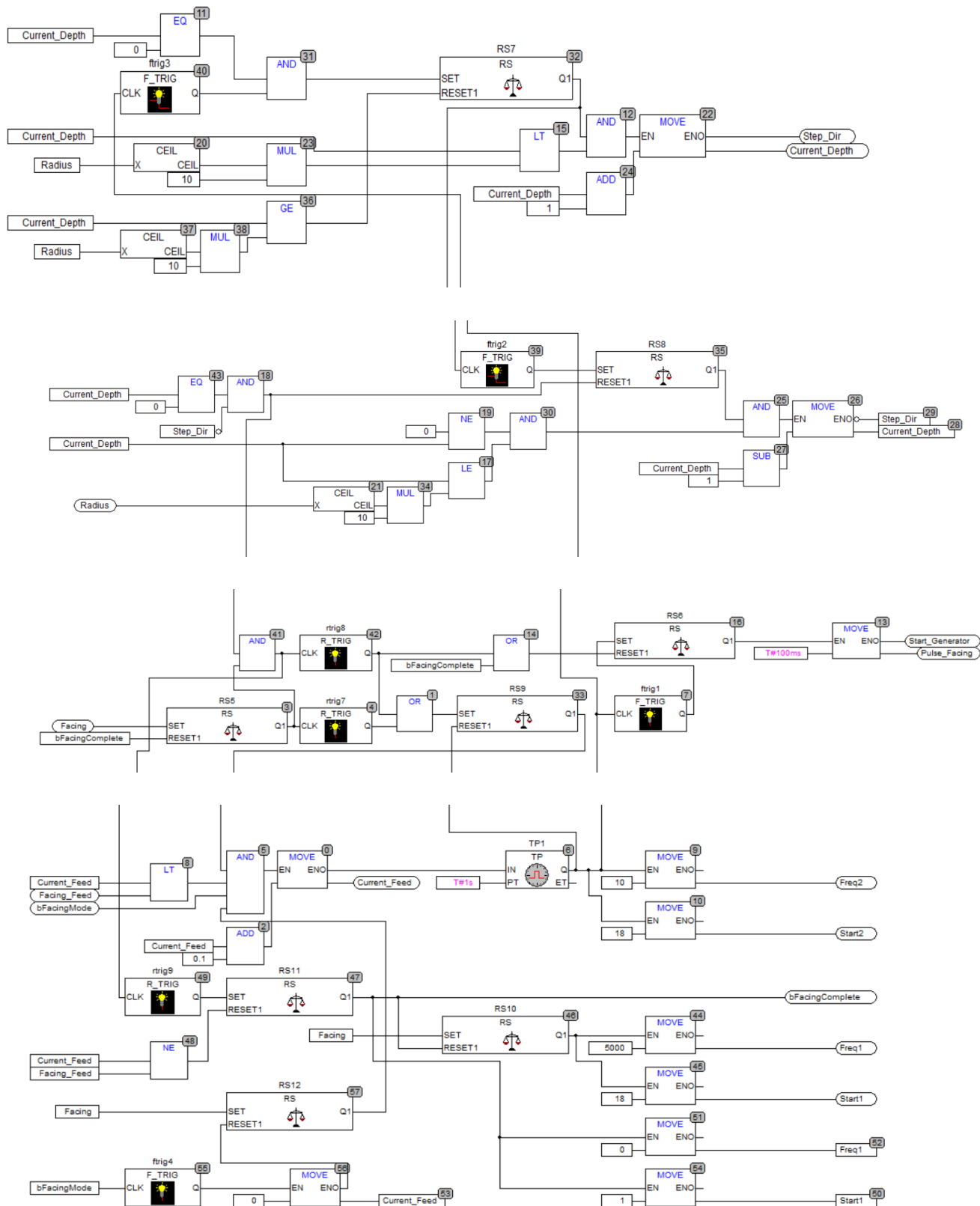


Рисунок 63 –Макрос Facing

Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.05.001 ПЗ

Лист

70

Функциональный блок Cross_Steps управляет движением шагового двигателя поперечной подачи в режимах обработки поверхности и нарезания резьбы. На вход поступают числа со счетчика, который подсчитывает импульсы генератора во время работы этих режимов, они записываются в переменную bPulse_Counter. Когда ее значение равно единице, переменная Step_Dir равна нулю, Current_Depth уменьшается на 1. При значениях bPulse_Counter 2 и 3 — в переменную Step_Dir записывается 1, а Current_Depth увеличивается на единицу. Схема функционального блока представлена на рисунке 64.

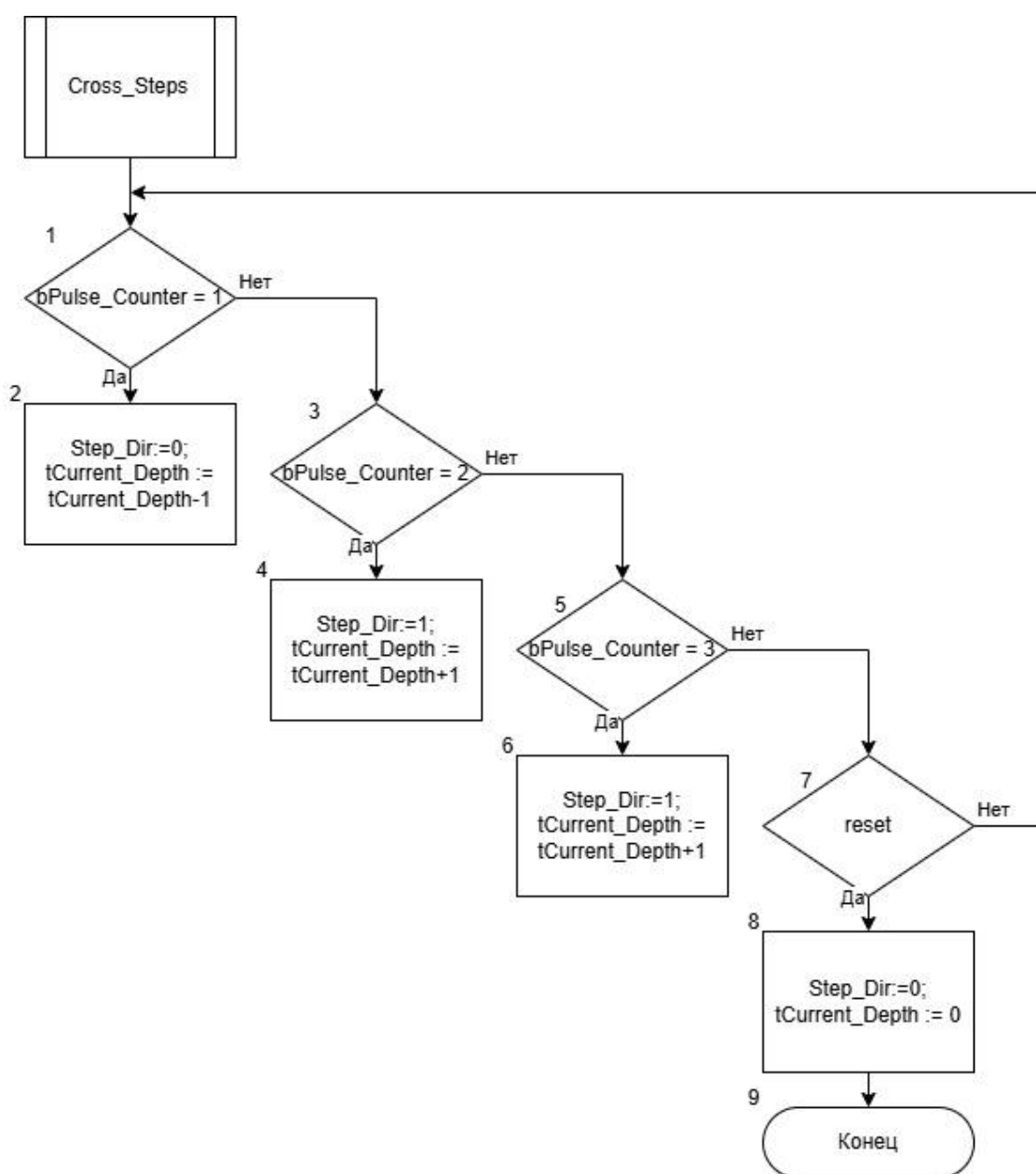


Рисунок 64 – Схема функционального блока Cross_Steps

Код функционального блока Cross_Steps на языке ST:

```
fbTrigger(CLK := Step_Pulse);
IF fbTrigger.Q THEN
  IF bPulse_Counter = 1 THEN
    Step_Dir:=0;
    tCurrent_Depth := tCurrent_Depth-1;
  ELSIF bPulse_Counter = 2 THEN
    Step_Dir:=1;
    tCurrent_Depth :=tCurrent_Depth+1;
  ELSIF bPulse_Counter = 3 THEN
    Step_Dir:=1;
    tCurrent_Depth := tCurrent_Depth+1;
  END_IF;
  ELSIF reset THEN
    Step_Dir:=0;
    tCurrent_Depth := 0;
  END_IF
```

Макрос Turning управляет процессом обработки поверхности. На вход подаются значения длины и глубины обработки, вводимые оператором. Они записываются в переменные Turning_Feed и Turning_Depth соответственно. Процесс начинается, когда значения переменных bTurningMode и Turning принимают значение 1. В переменную Start2 записывается значение 18, а в переменную Freq2 – 500: двигатель продольной подачи вращается в прямом направлении с частотой 5 герц. Переменная Current_Feed (длина обработки в настоящий момент времени) увеличивается на единицу до того момента, как значение переменной Current_Feed сравняется со значением Turning_Feed. В тот же момент переменная Start_Generator включает генератор импульсов. Генератор запускается на один импульс, его подсчитывает счетчик stu7. Его значение записывается в переменную bPulse_Counter. После этого в переменную Start2 записывается значение 34, а в переменную Freq2 – 5000: двигатель продольной подачи вращается в обратном направлении на максимальной скорости. Переменная Current_Feed уменьшается на единицу, пока значение переменной Current_Feed не

станет равным нулю. После этого включается генератор и счетчик увеличивает значение переменной bPulse_Counter, пока оно не будет равно 3. Далее процесс повторяется циклически, пока переменная Current_Feed не сравняется с переменной Turning_Depth. Алгоритм макроса представлен на рисунке 65, программа на языке CFC — на рисунке 66.

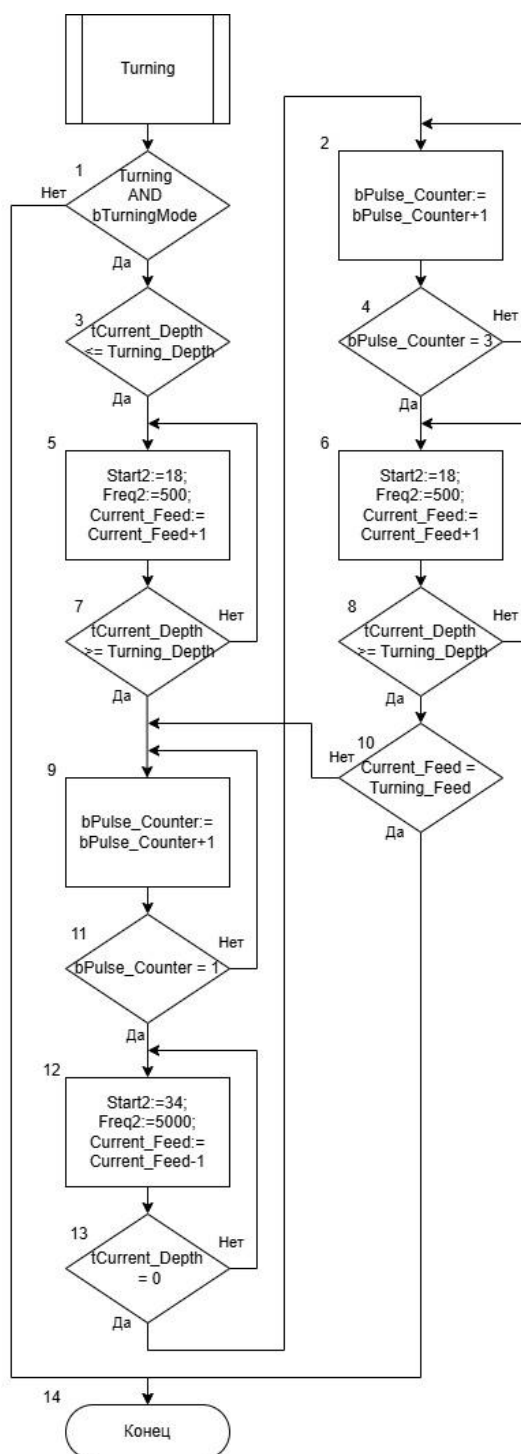


Рисунок 65 – Схема макроса Turning

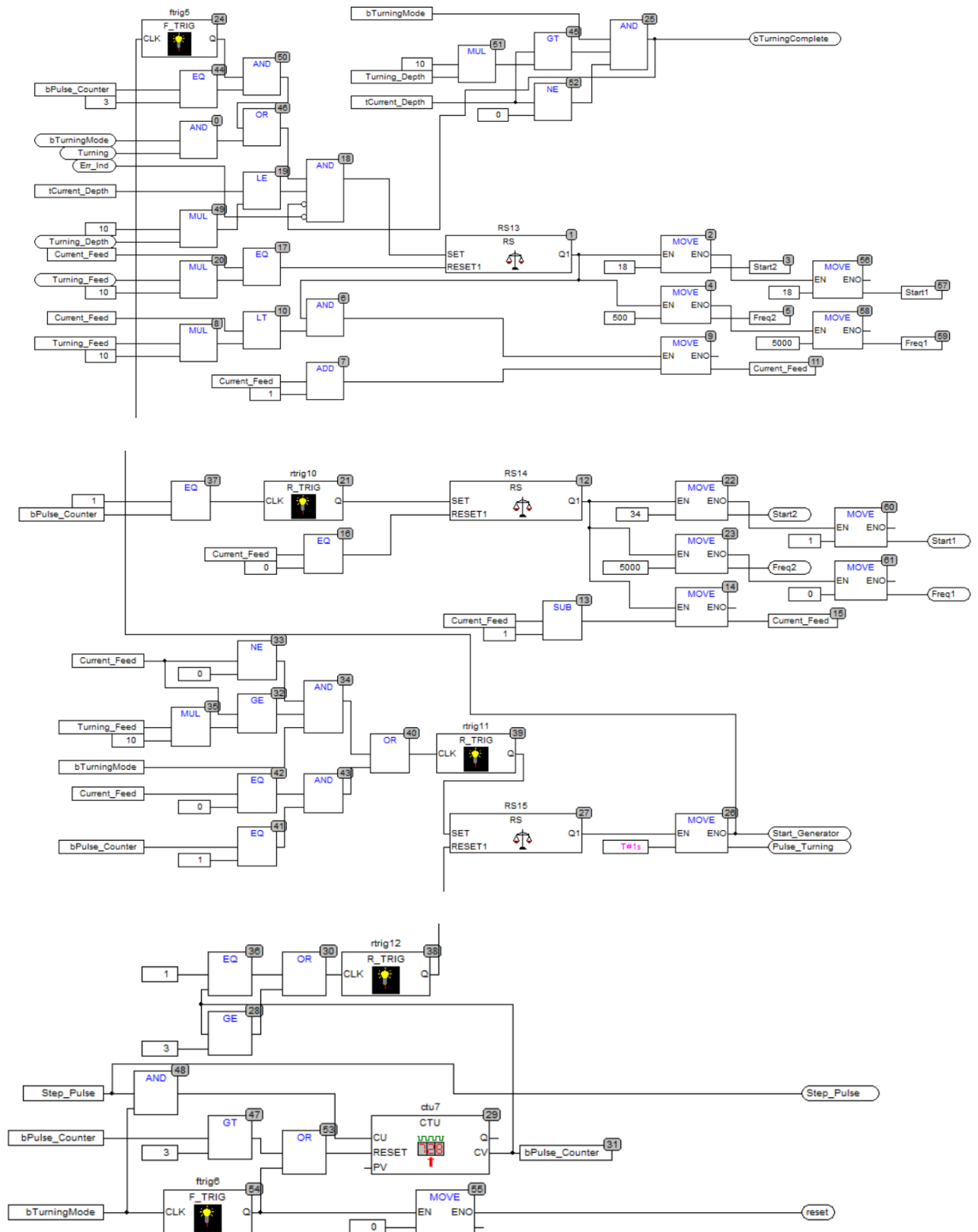


Рисунок 66 – Макрос Turning

Макрос Turning работает в паре с функциональным блоком cross_steps1 (рисунок 67).

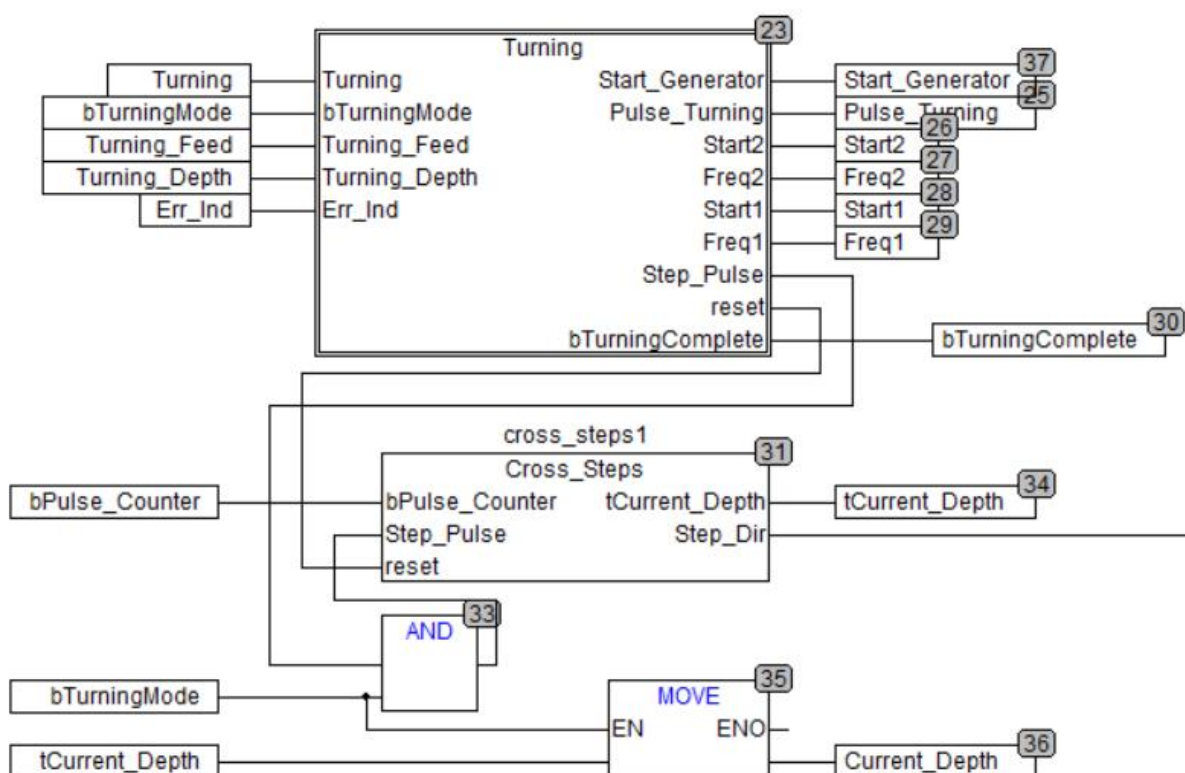


Рисунок 67 – Макрос Turning и функциональный блок Cross_Steps

Функциональный блок Threading_Calculator рассчитывает параметры резьбы, такие как шаг резьбы и ее глубина. На вход подаются значения диаметра заготовки и длина резьбы. Шаг резьб рассчитывается исходя из значения диаметра по ГОСТ 8724-2002. Глубина резьбы вычисляется по формуле $Depth_mm := 0.54127 \times Pitch_mm$ для метрической резьбы и $Depth_mm := 0.6495 \times Pitch_mm$, где $Depth_mm$ — глубина резьбы, $Pitch_mm$ — шаг резьбы. Схема функционального блока Threading_Calculator представлены на рисунках 68-69.

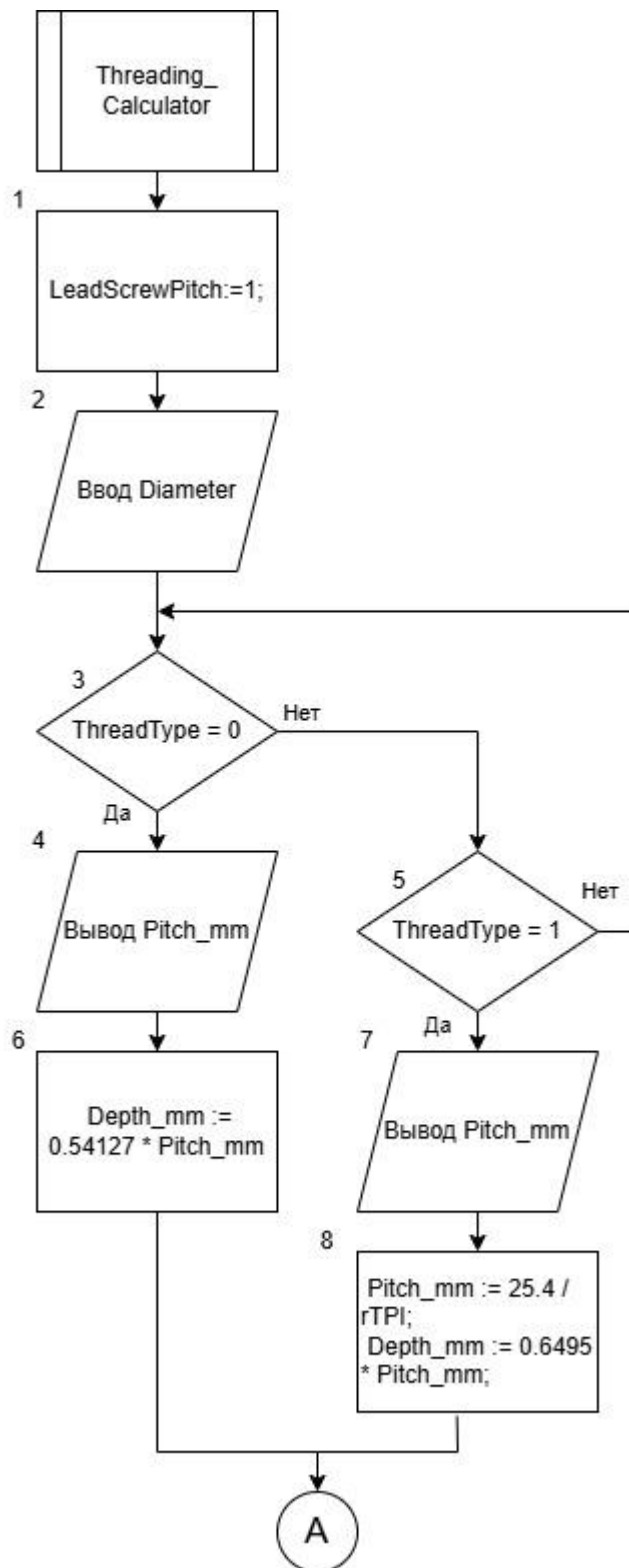


Рисунок 68 – Схема функционального блока Threading_Calculator

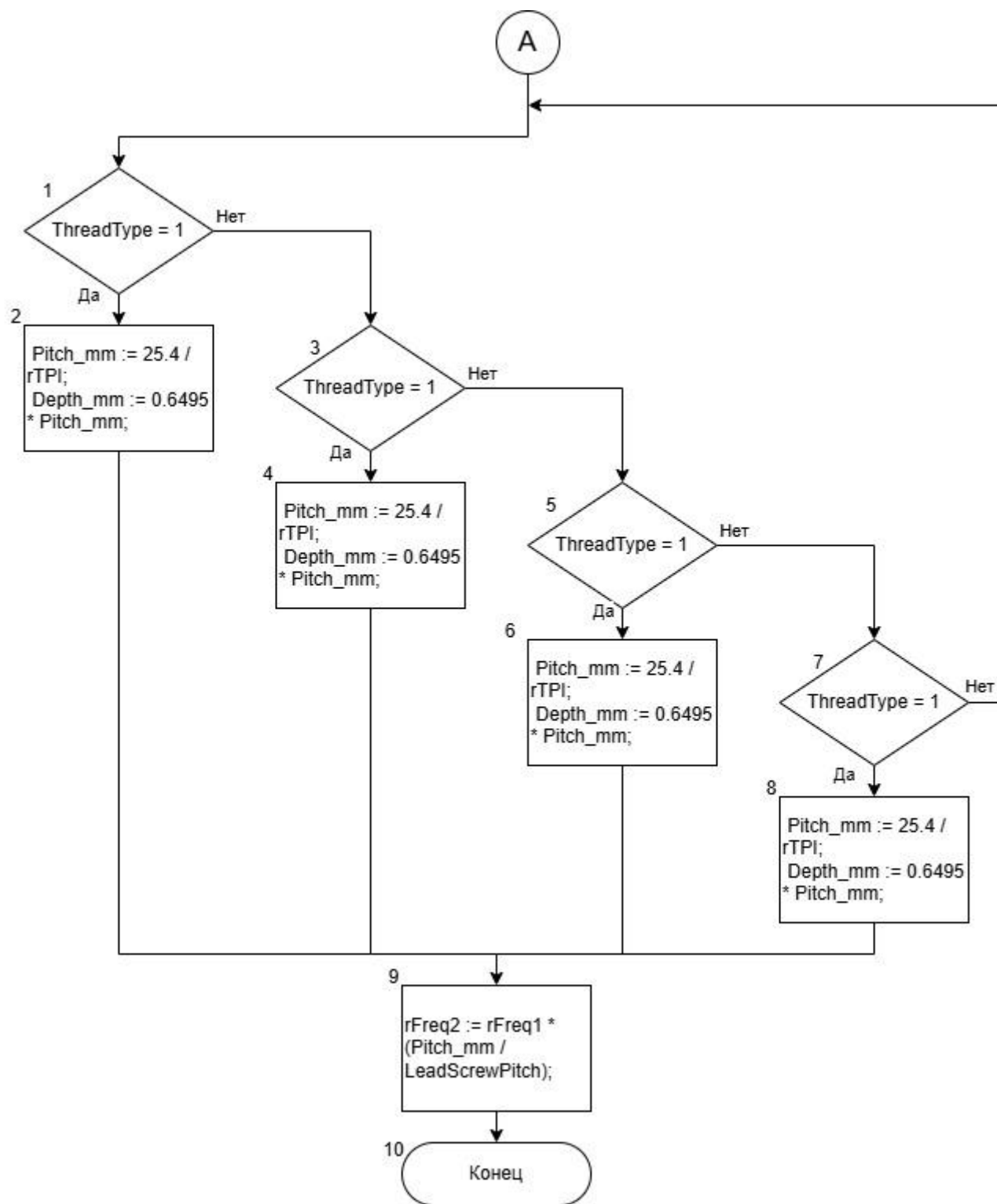


Рисунок 69 - Threading_Calculator. Продолжение

Код функционального блока Threading_Calculator на языке ST:

```

    LeadScrewPitch:=1;
    IF ThreadType = 0 THEN
    IF Diameter >= 1.0 AND Diameter <= 1.2 THEN

```

```

        Pitch_mm := 0.25;
ELSIF Diameter > 1.2 AND Diameter <= 1.6 THEN
        Pitch_mm := 0.35;
ELSIF Diameter > 1.6 AND Diameter <= 2.0 THEN
        Pitch_mm := 0.4;
ELSIF Diameter > 2.0 AND Diameter <= 2.5 THEN
        Pitch_mm := 0.45;
ELSIF Diameter > 2.5 AND Diameter <= 3.0 THEN
        Pitch_mm := 0.5;
ELSIF Diameter > 3.0 AND Diameter <= 4.0 THEN
        Pitch_mm := 0.7;
ELSIF Diameter > 4.0 AND Diameter <= 5.0 THEN
        Pitch_mm := 0.8;
ELSIF Diameter > 5.0 AND Diameter <= 6.0 THEN
        Pitch_mm := 1.0;
ELSIF Diameter > 6.0 AND Diameter <= 8.0 THEN
        Pitch_mm := 1.25;
ELSIF Diameter > 8.0 AND Diameter <= 10.0 THEN
        Pitch_mm := 1.5;
ELSIF Diameter > 10.0 AND Diameter <= 12.0 THEN
        Pitch_mm := 1.75;
ELSIF Diameter > 12.0 AND Diameter <= 16.0 THEN
        Pitch_mm := 2.0;
ELSIF Diameter > 16.0 AND Diameter <= 20.0 THEN
        Pitch_mm := 2.5;
ELSIF Diameter > 20.0 AND Diameter <= 24.0 THEN
        Pitch_mm := 3.0;
ELSIF Diameter > 24.0 AND Diameter <= 30.0 THEN
        Pitch_mm := 3.5;
ELSIF Diameter > 30.0 AND Diameter <= 36.0 THEN
        Pitch_mm := 4.0;
ELSIF Diameter > 36.0 AND Diameter <= 42.0 THEN
        Pitch_mm := 4.5;
ELSIF Diameter > 42.0 AND Diameter <= 48.0 THEN
        Pitch_mm := 5.0;

```

```

        ELSIF Diameter > 48.0 AND Diameter <= 56.0 THEN
            Pitch_mm := 5.5;
        ELSIF Diameter > 56.0 AND Diameter <= 68.0 THEN
            Pitch_mm := 6.0;
        ELSE
            Pitch_mm := 1.5;
        END_IF;
        Depth_mm := 0.54127 * Pitch_mm;
    ELSIF ThreadType = 1 THEN
        IF Diameter >= 0.125 AND Diameter <= 0.25 THEN
            rTPI := 64;    (* 1/8" - 1/4" *)
        ELSIF Diameter > 0.25 AND Diameter <= 0.5 THEN
            rTPI := 56;    (* 1/4" - 1/2" *)
        ELSIF Diameter > 0.5 AND Diameter <= 0.75 THEN
            rTPI := 40;
        ELSIF Diameter > 0.75 AND Diameter <= 1.0 THEN
            rTPI := 32;
        ELSIF Diameter > 1.0 AND Diameter <= 1.5 THEN
            rTPI := 28;
        ELSIF Diameter > 1.5 AND Diameter <= 2.0 THEN
            rTPI := 24;
        ELSIF Diameter > 2.0 AND Diameter <= 3.0 THEN
            rTPI := 20;
        ELSIF Diameter > 3.0 AND Diameter <= 4.0 THEN
            rTPI := 16;
        ELSE
            rTPI := 16;
        END_IF;
        Pitch_mm := 25.4 / rTPI;
        Depth_mm := 0.6495 * Pitch_mm;
    END_IF;
    IF Pitch_mm < 0.5 THEN
        rFreq1 := 2000;
    ELSIF Pitch_mm < 1.0 THEN
        rFreq1 := 1500;
    
```

```

        ELSIF Pitch_mm < 1.5 THEN
            rFreq1 := 1200;
        ELSIF Pitch_mm < 2.5 THEN
            rFreq1 := 1000;
        ELSE
            rFreq1 := 800;
    END_IF;
    rFreq2 := rFreq1 * (Pitch_mm / LeadScrewPitch);

```

Макрос Threading управляет станком в режиме нарезания резьбы. Его логика копирует логику работы макроса Turning, за исключением того, что вместо переменной Turning_Depth на вход подается значение переименованной Depth_mm (Глубина резьбы), рассчитанная во время работы функционального блока Threading_Calculator. Вместо переменной Turning_Feed на вход макроса передается переменная ThreadLength (Длина резьбы). Также вводятся значения частоты вращения двигателей шпинделя и продольной подачи, рассчитанные функциональным блоком Threading_Calculator (переменные rFreq1 и rFreq2). Эти параметры записываются в переменные, передающие во время работы макроса, установленную частоту на преобразователь. Схема макроса Threading представлена на рисунке 70. Код макроса на языке CFC — на рисунке 71.

					ПГУ 1.090501.05.001 ПЗ	Лист
						80
Изм.	Лист	№ докум.	Подпись	Дата		

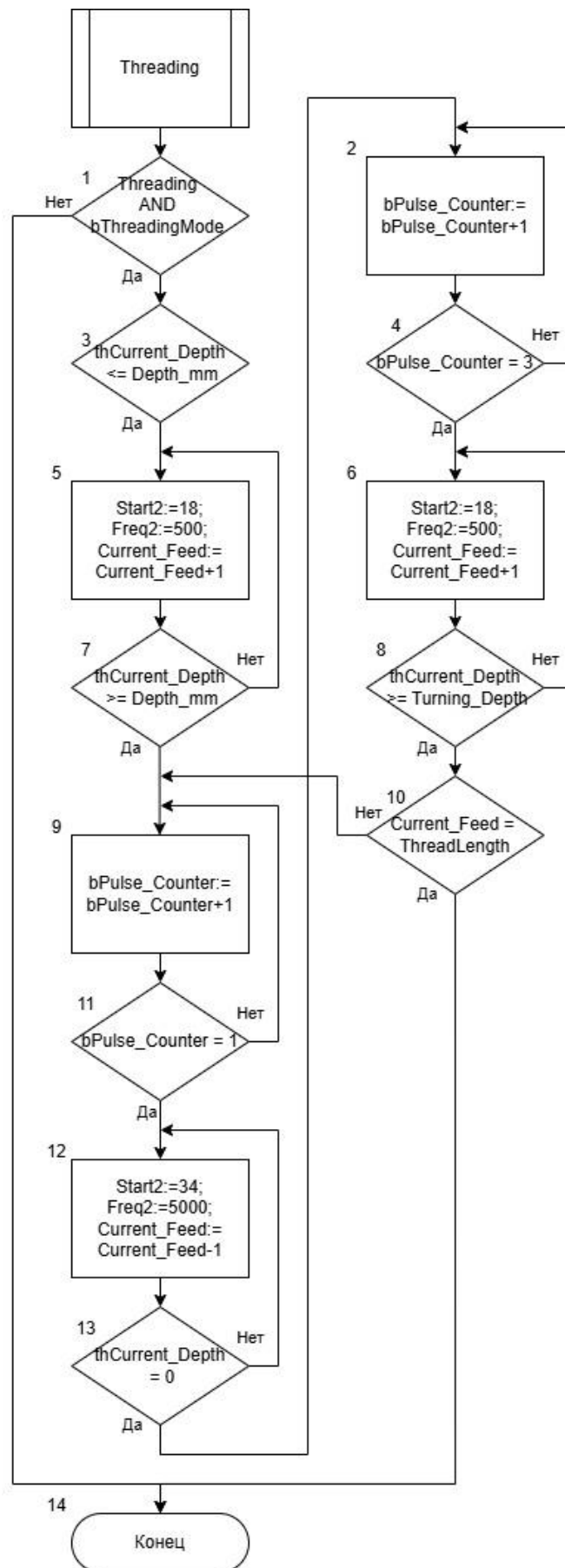


Рисунок 70 – Схема макроса Threading

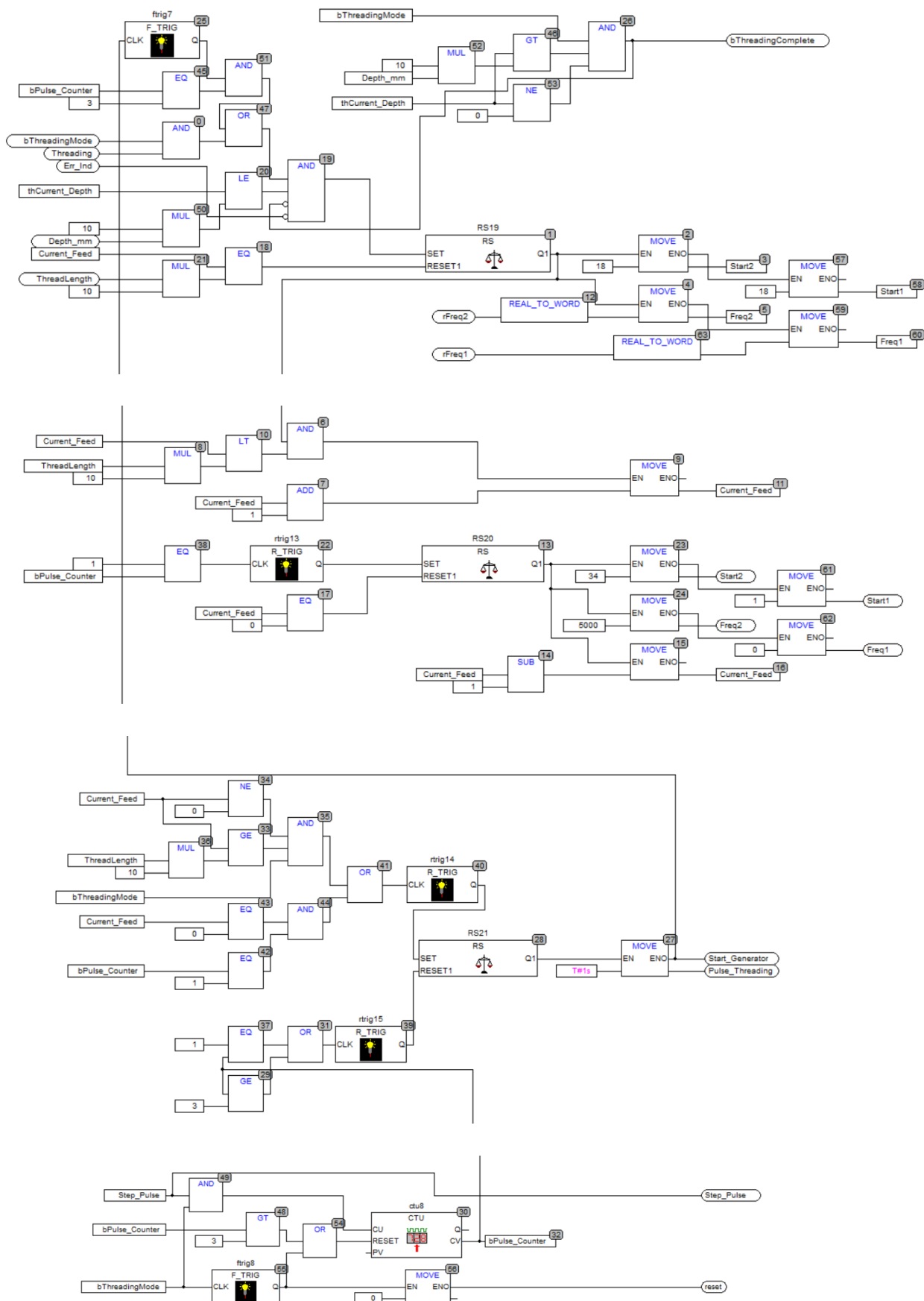


Рисунок 71 – Макрос Threading

Макрос Threading взаимодействует с функциональными блоками Threading_Calculator и cross_steps2 (рисунок 72).

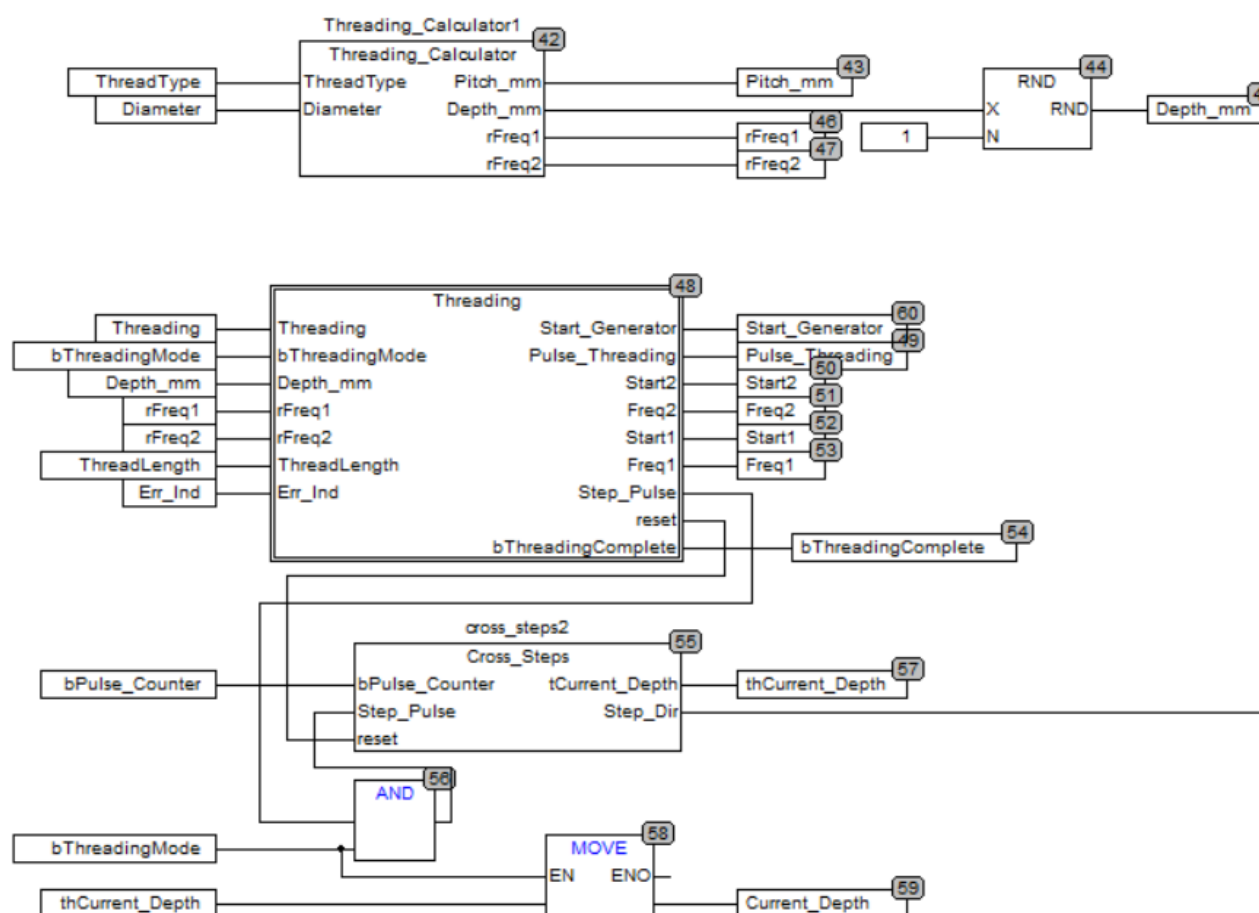


Рисунок 72 –Threading_Calculator, Threading, Cross_steps

1.2.3 Конфигурирование панели оператора

Исходя из заложенного в управляющую программу функционала были описаны UML-диаграммы деятельности и сценариев использования, в которых описаны шаблоны взаимодействия пользователя и разрабатываемой системы.

На рисунке 73 представлена UML-диаграмма деятельности, описывающая взаимодействие, пользователя, соответствующей панели оператора и ПЛК.

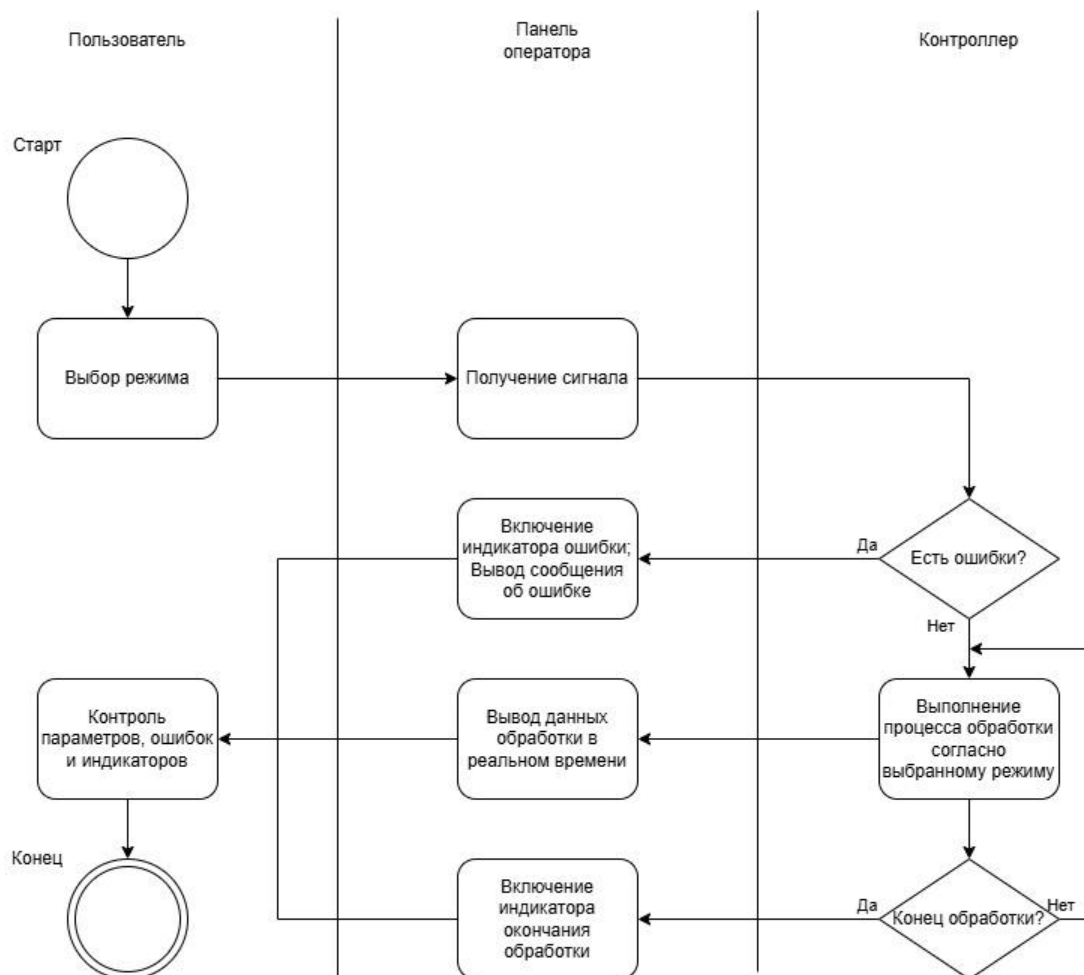


Рисунок 73 – UML-диаграмма деятельности для пользователя

На рисунке 74 представлена UML-диаграмма сценариев использования, описывающая шаблон использования системы. Оператор может выбрать один из режимов, ввести параметры обработки и контролировать процесс, который выполняется автоматически.



Рисунок 74 – UML-диаграмма сценариев использования для пользователя

После анализа вариантов взаимодействия оператора с системой был начат процесс создания интерфейса панели оператора

Во вкладке «Визуализации» был создан объект, названный «Vis1» (рисунок 75).

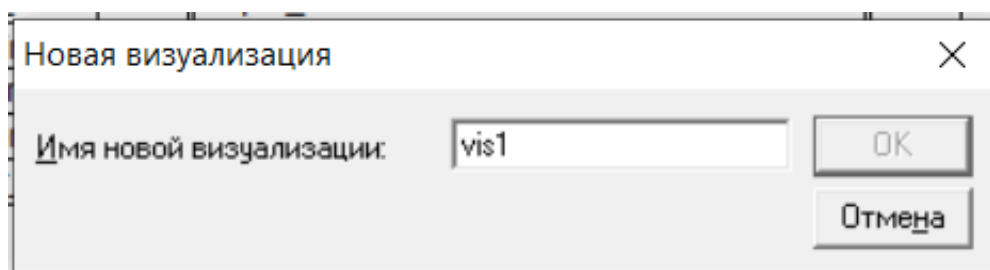


Рисунок 75 – Создание новой визуализации

После добавления нового объекта открывается окно визуализации, в котором будет вестись разработка интерфейса (рисунок 76).

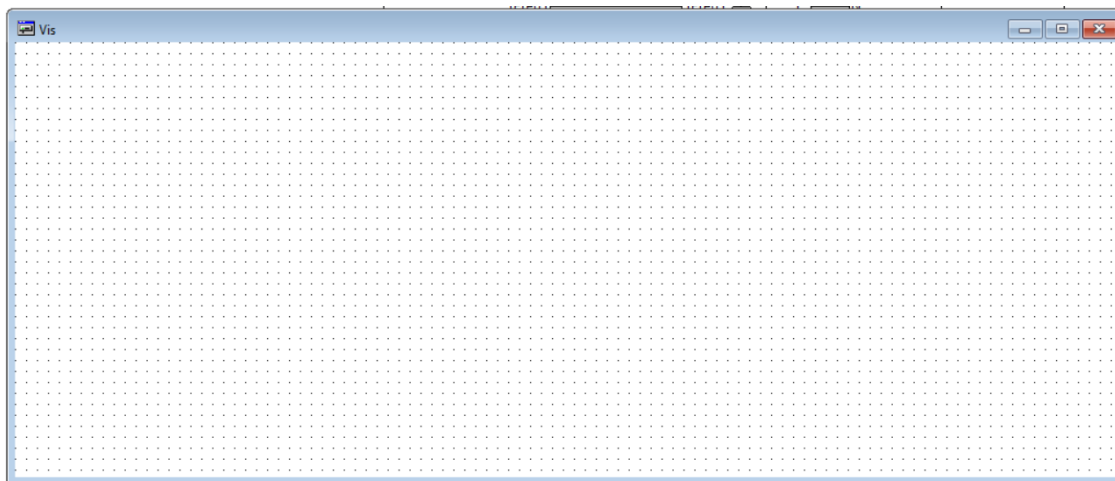


Рисунок 76 – Окно визуализации

Для создания объектов интерфейса, таких как кнопки, индикаторы, поля ввода и вывода доступны различные графические примитивы — прямоугольники, скругленные прямоугольники, круги, многоугольники. Также при построении интерфейса можно использовать таблицы, тренды, ползунки, стрелочные индикаторы, столбчатые диаграммы, гистограммы. Все это доступно в панели инструментов среды разработки визуализации (рисунок 77).



Рисунок 77 – Панель инструментов

На рисунке 78 показан фрагмент визуализации, при помощи которого оператор управляет функцией торцевания. Он состоит из информационных текстовых полей, полей ввода данных, кнопок и индикатора.

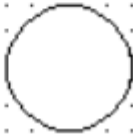
Длина торцевания. мм		%s
Торцевание		
Длина обработки	%s	
Глубина обработки	%s	
Начать торцевание		

Рисунок 78 – Фрагмент визуализации

Для создания информационного поля «Длина торцевания. мм» был добавлен элемент Прямоугольник, был задан его размер. Двойным нажатием по элементу открывается окно «Конфигурирование элемента». Во вкладке «Текст» в настройке «Строка» указано «Длина торцевания. мм», горизонтальное и вертикальное выравнивание выбрано «По центру» (рисунок 79). Остальные настройки оставлены по умолчанию. Другие информационные текстовые поля созданы аналогично.

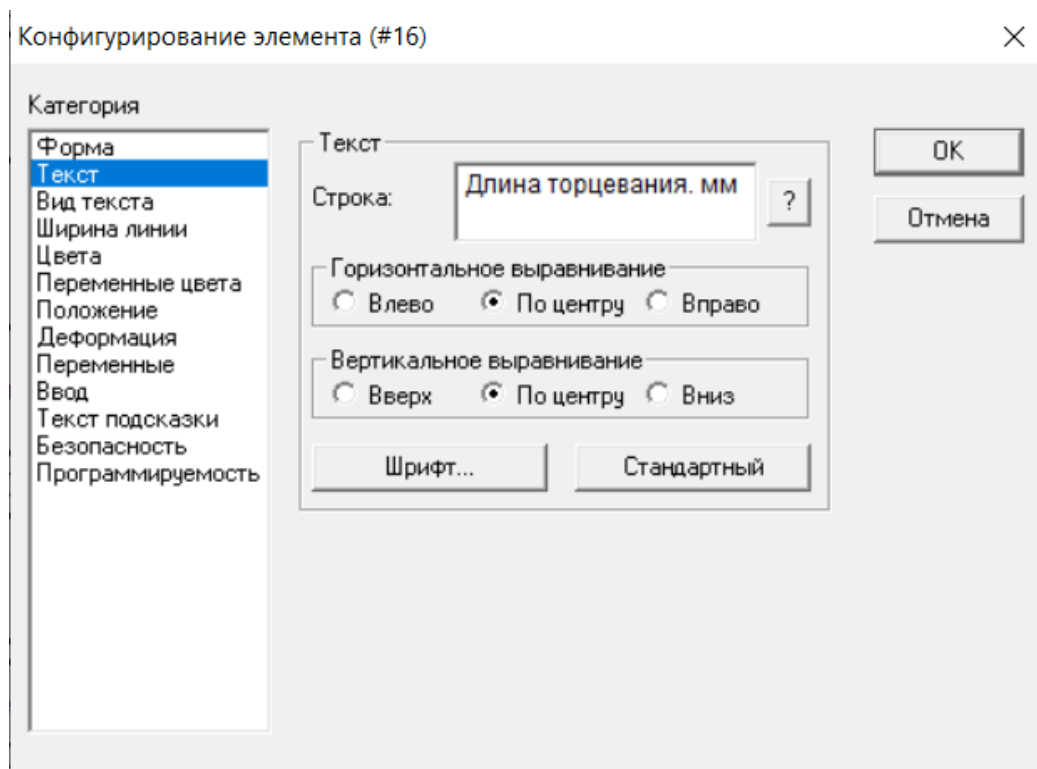


Рисунок 79 – Конфигурирование элемента информационной строки

Для создания кнопки «Торцевание» был выбран элемент Скругленный прямоугольник. На вкладке «Текст» окна «Конфигурирование элемента» в поле «Строка» указано «Торцевание». Горизонтальное и вертикальное выравнивание выбрано «По центру» (рисунок 80). На вкладке «Ширина линии» Выбрана ширина линии 3 (рисунок 81). На вкладке «Цвета» в настройках «Тревожный цвет» выбран зеленый (рисунок 82). На вкладке «Переменные» в строке «Запрет ввода» указано «.Facing_Dis», в строке «Изм. цвета» — «.bFacingMode» (рисунок 83). На вкладке «Ввод» галочкой выбрана настройка «Пер-я переключения» и в строке указана переменная «.bFacingMode» (рисунок 84). Таким образом, по нажатию кнопки значение переменной будет из нуля обращаться в единицу, а по повторному нажатию — из единицы в ноль. Остальные кнопки созданы и настроены по такому же алгоритму.

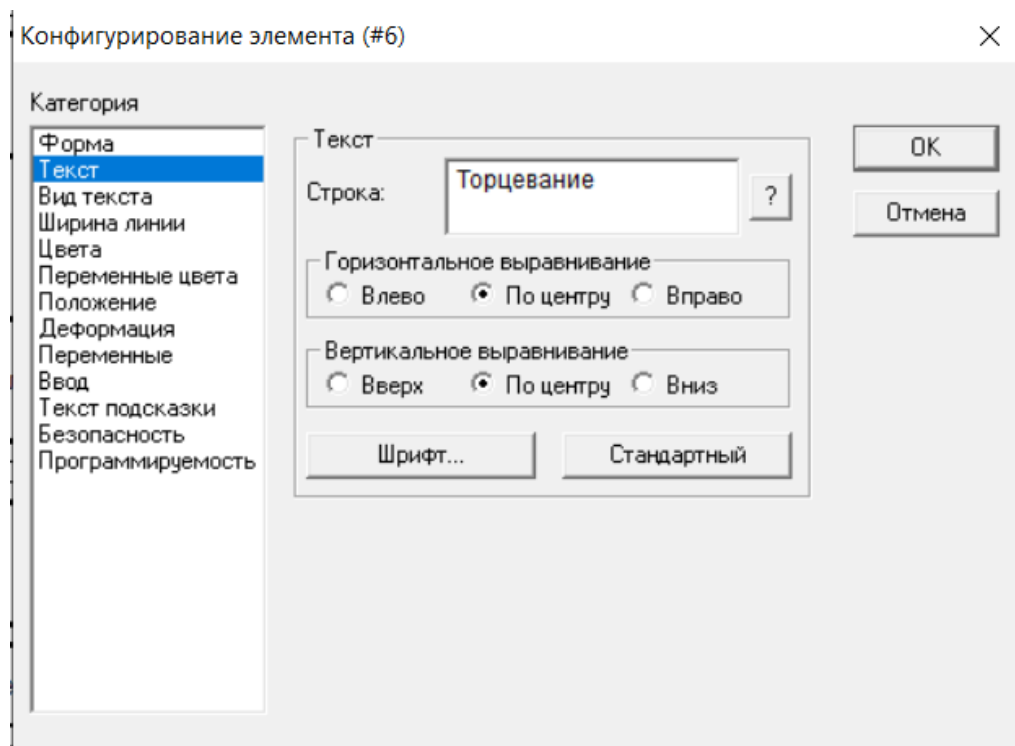


Рисунок 80 – Вкладка «Текст»

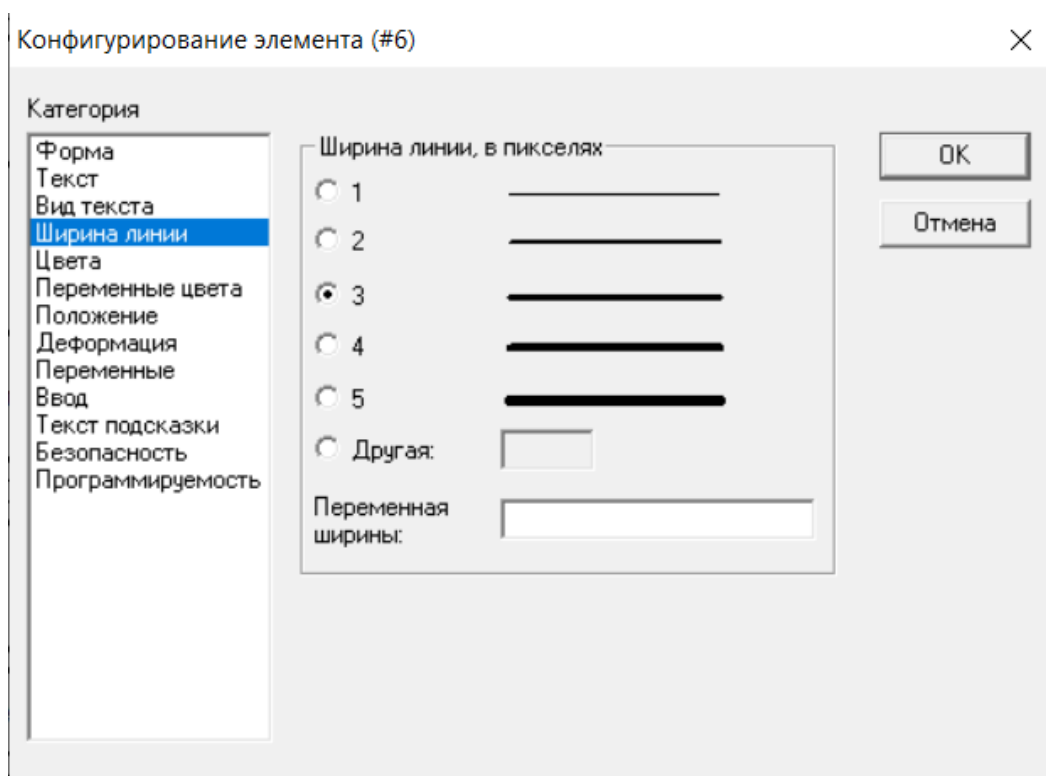


Рисунок 81 – Вкладка «Ширина линии»

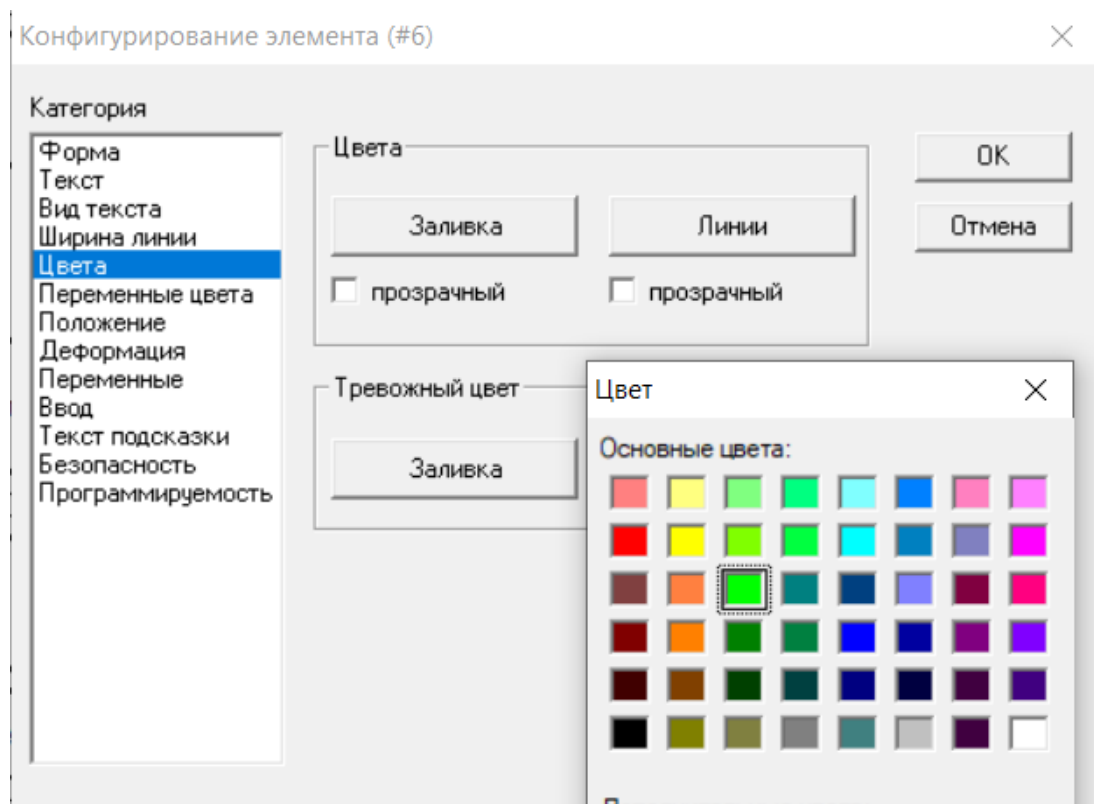


Рисунок 82 – Вкладка «Цвета»

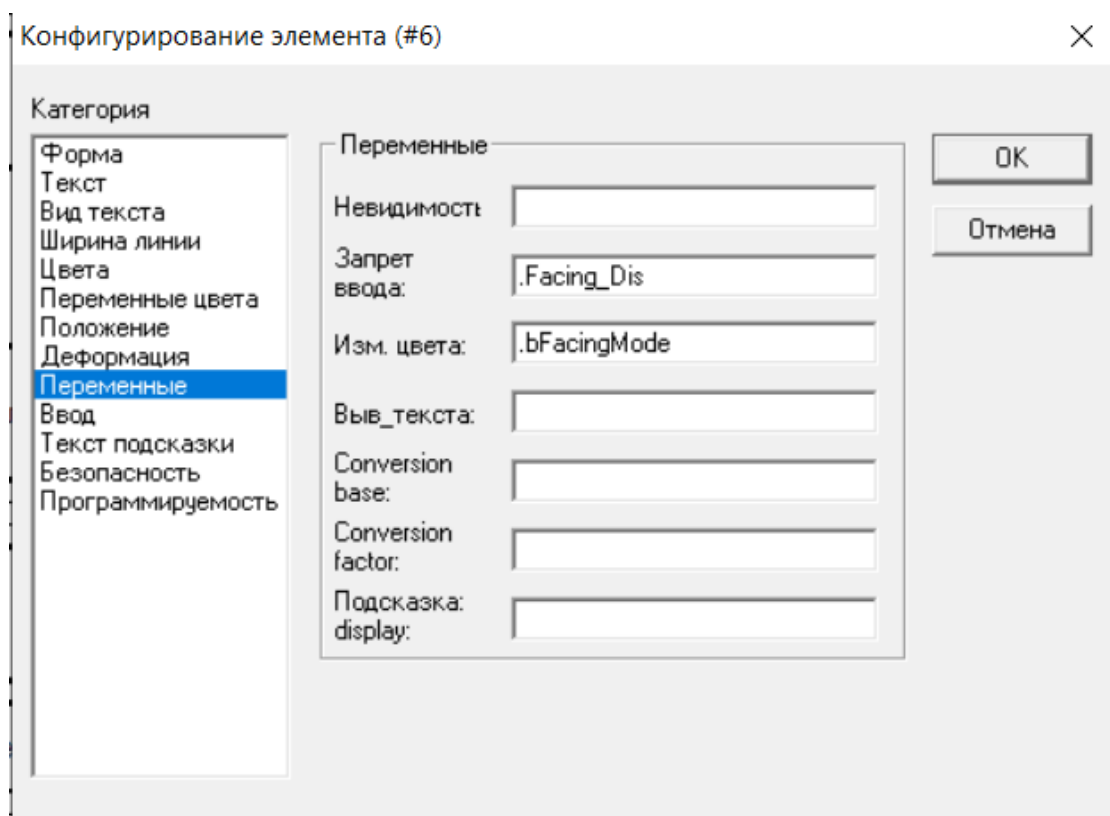


Рисунок 83 – Вкладка «Переменные»

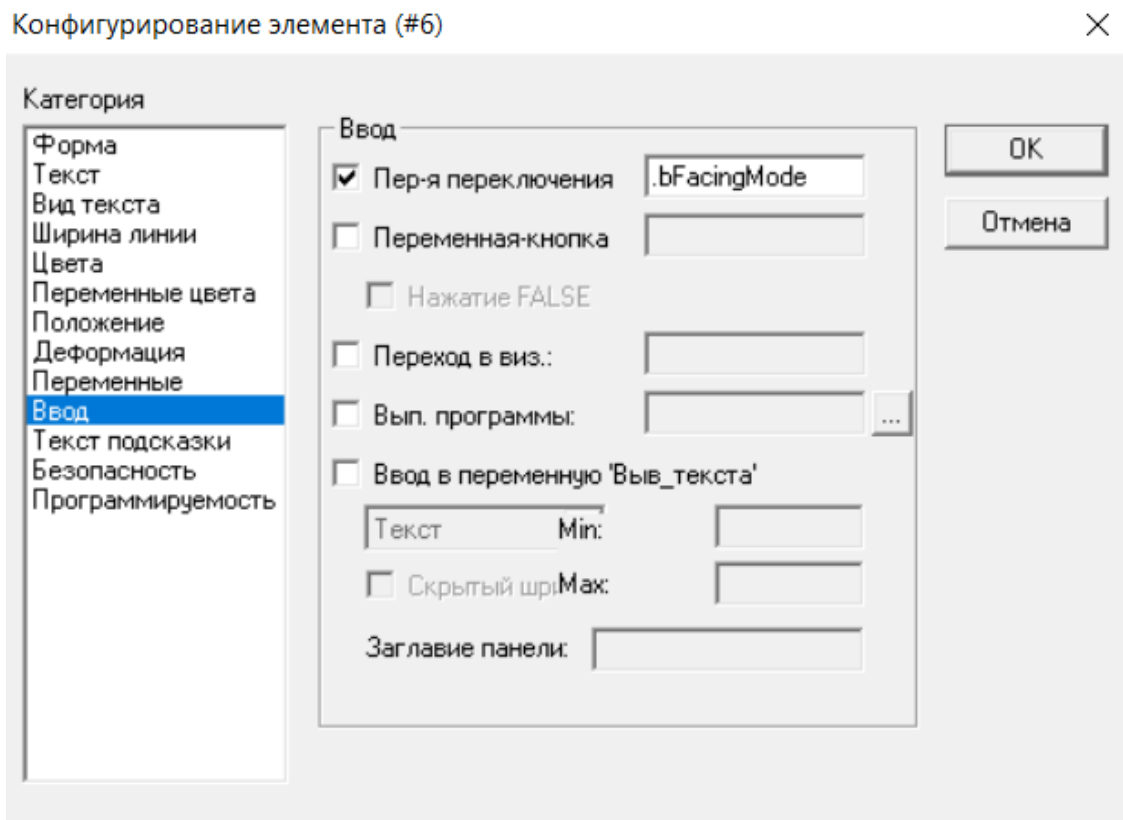


Рисунок 84 – Вкладка «Ввод»

Для создания поля ввода данных выбран элемент Прямоугольник. Во вкладке «Текст» в поле «Строка» указывается «%s» (рисунок 85). Во вкладке «Переменная» в поле «Выв_текста» указывается переменная, значение которой должно вводиться в поле ввода (рисунок 86). Во вкладке «Ввод» галочкой был выбран пункт «Ввод в переменную Выв_текста», указана «Цифр. панель» в выпадающем списке (рисунок 87). При настройке полей вывода галочка напротив пункта «Ввод в переменную Выв_текста» не ставится.

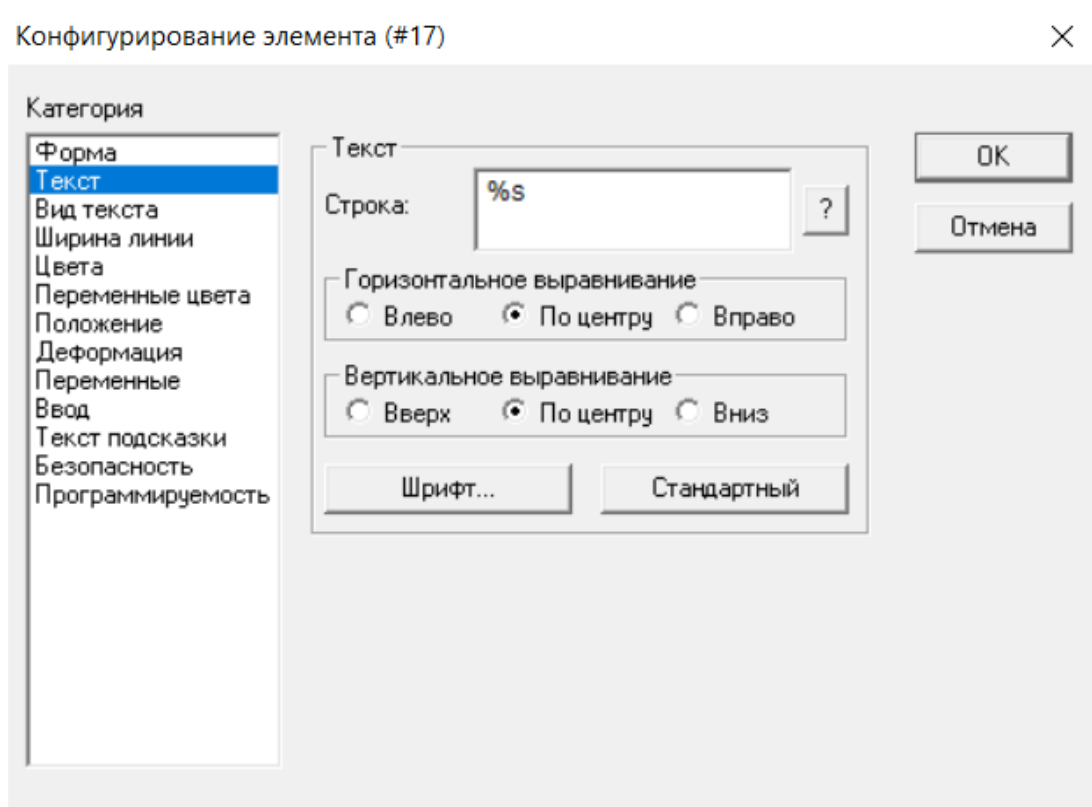


Рисунок 85 – Вкладка «Текст»

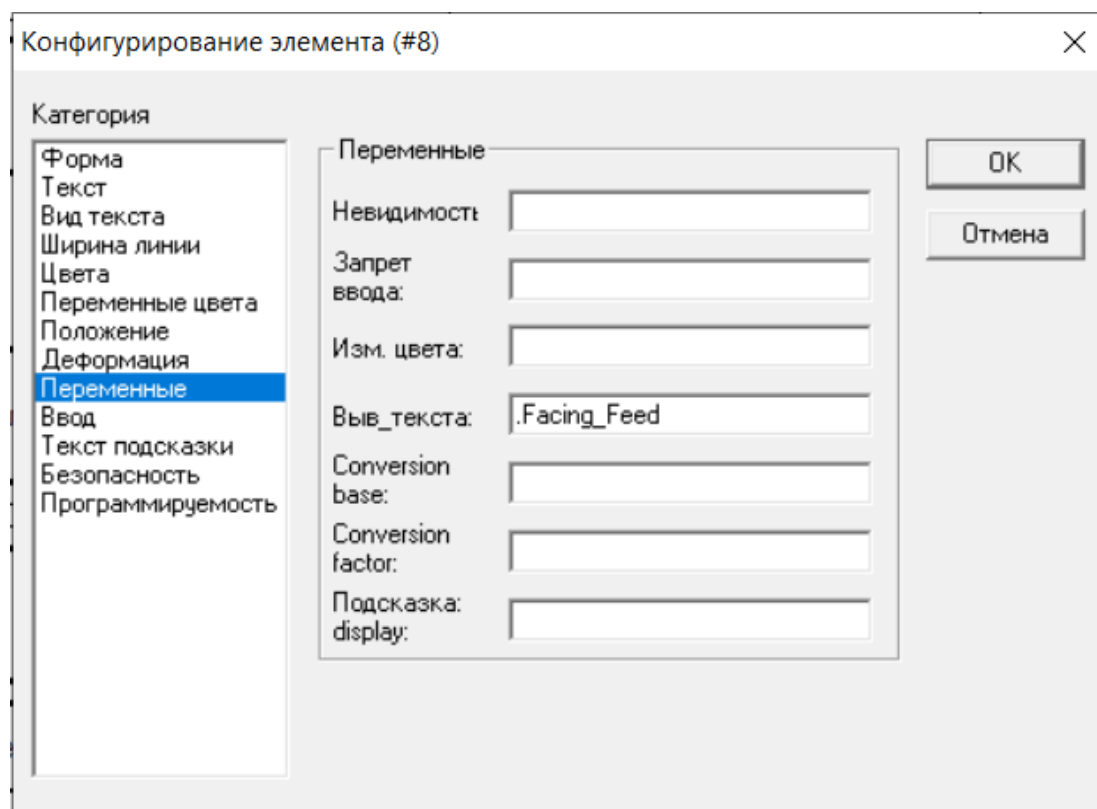


Рисунок 86 – Вкладка «Переменные»

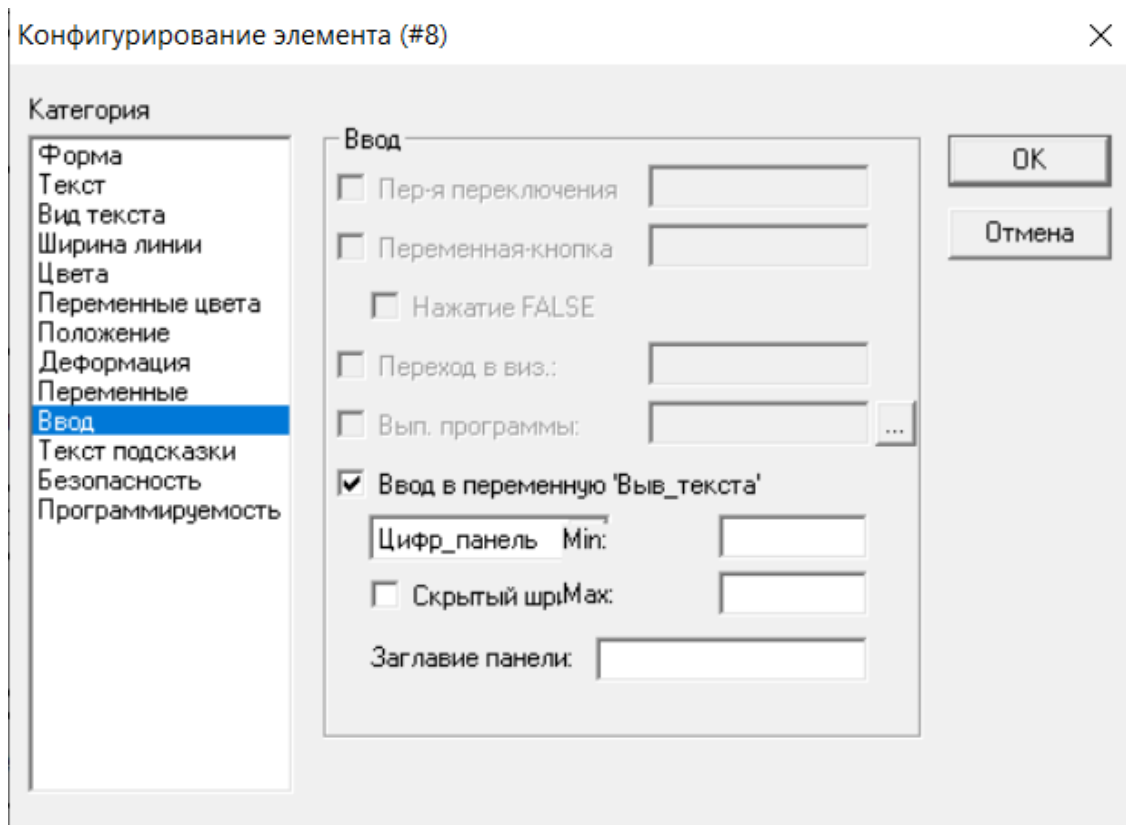


Рисунок 87 – Вкладка «Ввод»

Для создания индикаторов был выбран элемент «Круг». Во вкладке «Цвета», «Тревожный цвет» был выбран красный (рисунок 88). Во вкладке «Переменные» в поле «Изм. цвета» указывается переменная, по сигналу которой индикатор меняет цвет на тревожный (рисунок 89).

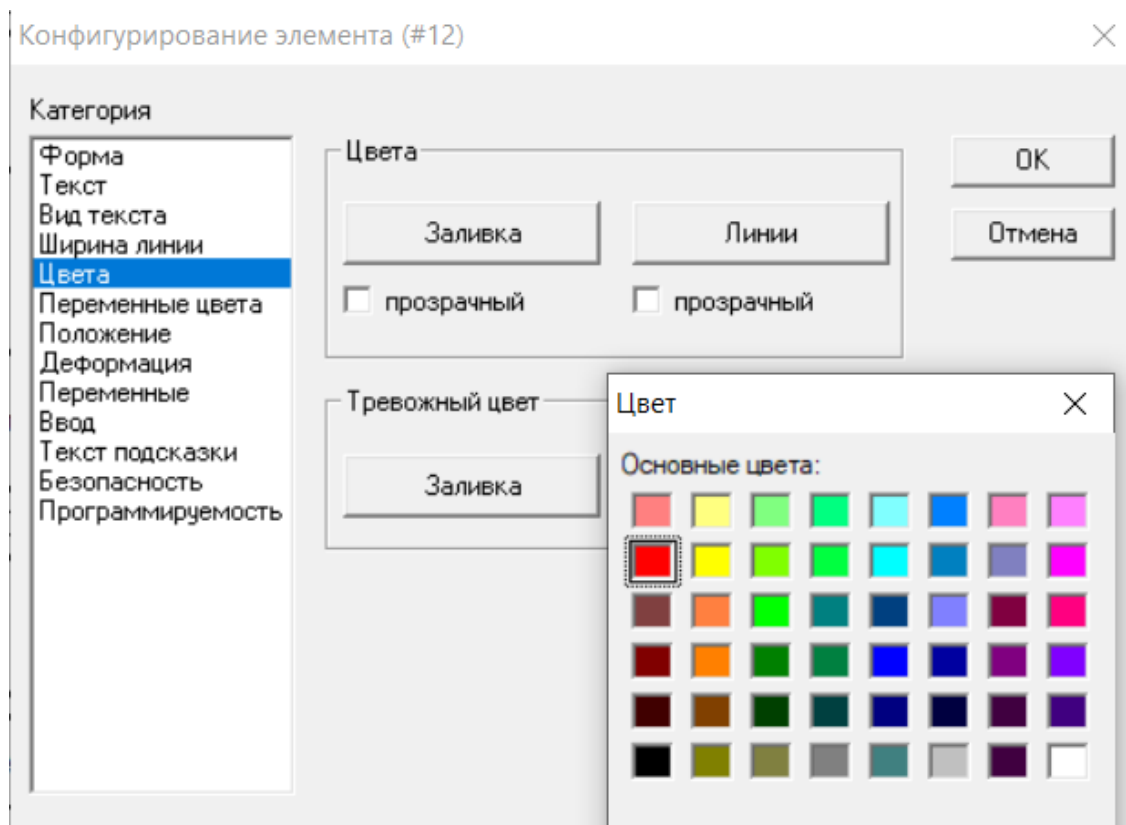


Рисунок 88 – Вкладка «Цвета»

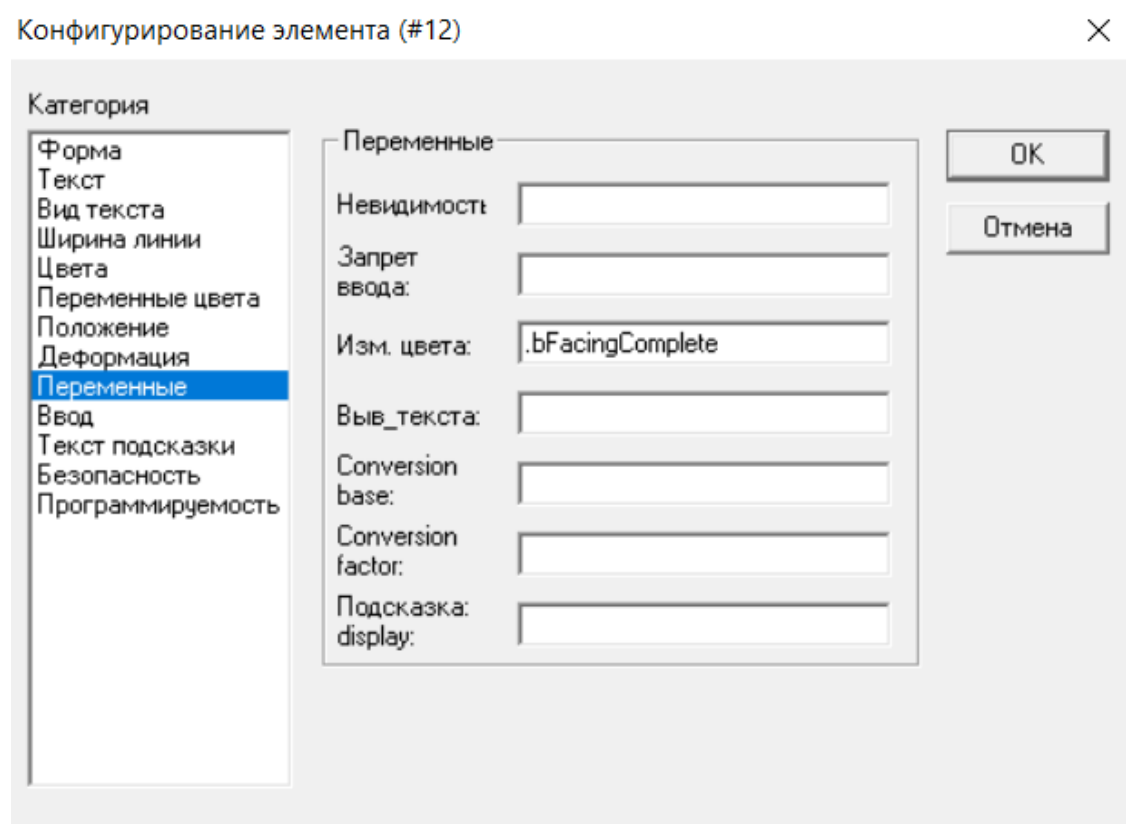


Рисунок 89 – Вкладка «Переменные»

Интерфейс панели оператора разделен на два экрана. Первый, Vis1, представляет собой панель управления. Главной его функцией является вывод сообщений об ошибках, ввод параметров и запуск процессов обработки. Также на него выводятся координаты продольной и поперечной подачи в реальном времени. Второй экран, Vis2, служит для вывода состояний процессов на графиках в реальном времени. Используются тренды для вывода координат и столбчатые диаграммы для вывода частоты вращения двигателей. Экраны представлены на рисунках 90-91.

Рисунок 90 – Экран «Панель управления»

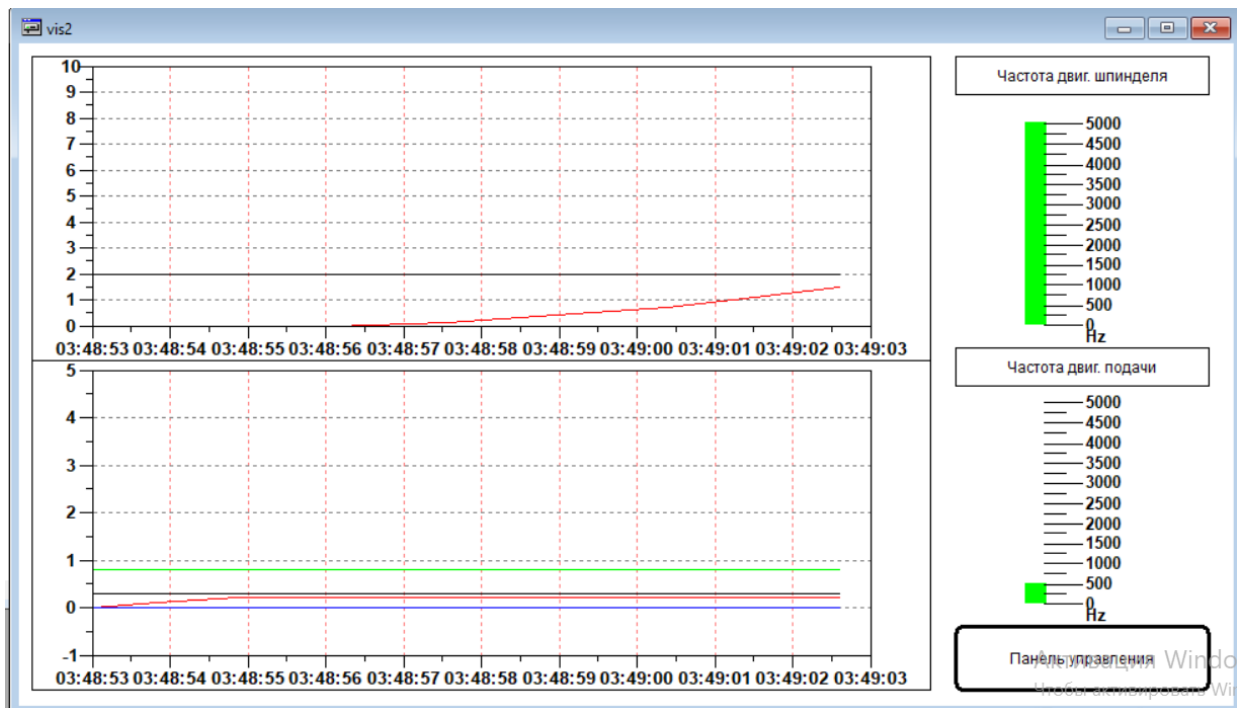


Рисунок 91 – Экран «Графики»

1.3 Анализ результатов

1.3.1 Тестирование и отладка управляющей программы и интерфейса

Для проверки корректности работы созданной управляющей программы в соответствии с исходным заданием было проведено тестирование с использованием режима эмуляции.

При запуске программы выводится сигнал об ошибке «Не закрыты защитные кожухи». Кнопка «Заккрыть защиткные кожухи» имитирует сигнал с концевых выключателей защитных кожухов, по ее нажатию сообщение об ошибке пропадает (рисунок 92-93).

vis1

ОШИБКА

Не закрыты защитные кожухи

Диаметр заготовки. мм	0	Длина торцевания. мм	0	Длина. мм	0	Глубина. мм	0
-----------------------	---	----------------------	---	-----------	---	-------------	---

Установка нуля

Вперед

Влево

Назад

Вправо

Торцевание

Длина обработки

0

Глубина обработки

0

Начать торцевание

Обработка поверхности

Длина обработки

0

Глубина обработки

0

Начать обработку

Нарезание резьбы

Длина резьбы. мм

0

Размер резьбы

0

Шаг резьбы

1.5

Длина обработки

0

Глубина обработки

0

Начать нарезание

Метрическая

Дюймовая

Заккрыть защитные кожухи

Графики

Рисунок 92 – Ошибка «Не закрыты защитные кожухи»

vis1

ОШИБКА

Диаметр заготовки. мм	0	Длина торцевания. мм	0	Длина. мм	0	Глубина. мм	0
-----------------------	---	----------------------	---	-----------	---	-------------	---

Установка нуля

Вперед

Влево

Назад

Вправо

Торцевание

Длина обработки

0

Глубина обработки

0

Начать торцевание

Обработка поверхности

Длина обработки

0

Глубина обработки

0

Начать обработку

Нарезание резьбы

Длина резьбы. мм

0

Размер резьбы

0

Шаг резьбы

1.5

Длина обработки

0

Глубина обработки

0

Начать нарезание

Метрическая

Дюймовая

Заккрыть защитные кожухи

Графики

Рисунок 93 – Ошибка «Не закрыты защитные кожухи»

После этого тестируется функция установки нулевой точки. По нажатию кнопки «Установка нуля» включается режим ручного управления подачей. Для продольной подачи управление осуществляется кнопками «Влево» и «Вправо». По первому нажатию кнопки в переменную Freq2 записывается значение 1000, по второму нажатию — значение 5000 (рисунки 94-95).

Управление поперечной подачей осуществляется нажатием на кнопки «Вперед» и «Назад». По первому нажатию в переменную Pulse_Setting записывается значение 1s0ms, по второму — 100ms (рисунки 96-97).

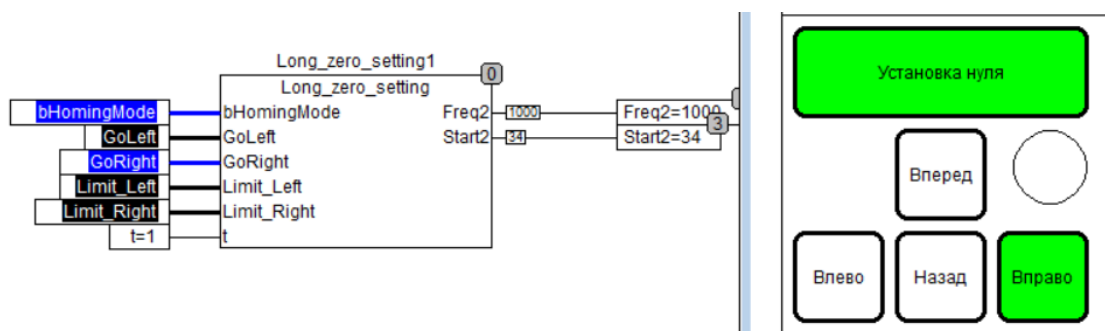


Рисунок 94 – Тестирование продольной подачи

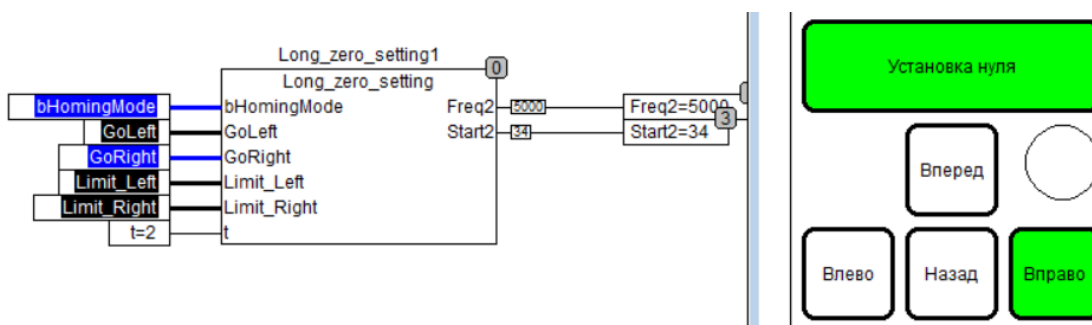


Рисунок 95 – Тестирование продольной подачи

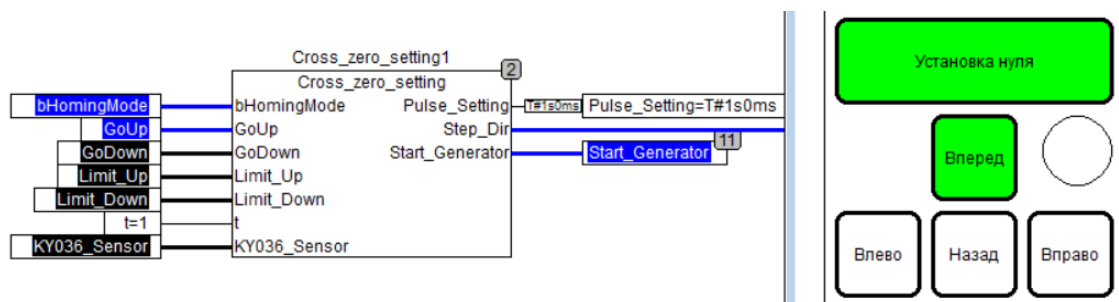


Рисунок 96 – Тестирование поперечной подачи

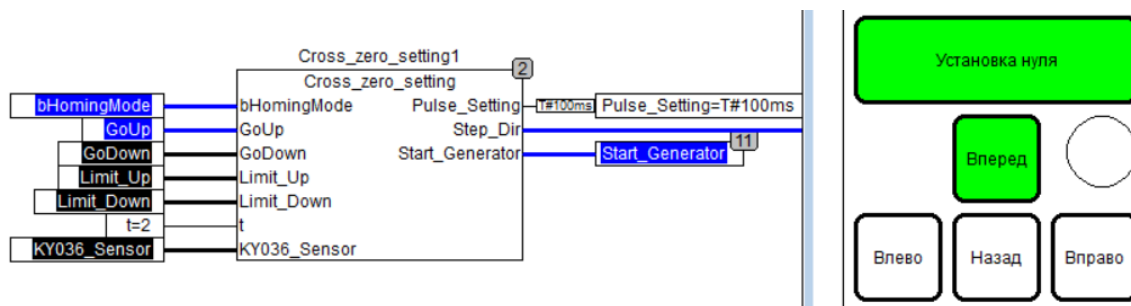


Рисунок 97 – Тестирование поперечной подачи

Далее тестируется функция обработки торца. Режим включается по кнопке «Торцевание». После запуска выводятся сообщения об ошибках «Не указана длина торцевания» и «Не указан диаметр». После ввода этих параметров ошибки пропадают (рисунки 98-99). На первом этапе работы алгоритма Поперечная подача движется вперед, пока значение глубины обработки не станет равным радиусу заготовки (рисунки 100-101). На втором этапе поперечная подача движется назад пока значение глубины обработки не достигнет нуля (рисунки 102-103). На третьем этапе работает поперечная подача, с малой частотой вращения двигателя (1Гц) подача сдвигается на 0,1 миллиметра (рисунки 104-105). Все значения отображаются на графике в реальном времени (рисунок 106). Далее процесс повторяется циклически, и когда значение длины обработки достигнет заданной длины обработки, он завершается, при этом включается красный индикатор (рисунок 107).

vis1

ОШИБКА

Не указана длина торцевания

Диаметр заготовки. мм	0	Длина торцевания. мм	0	Длина. мм	0	Глубина. мм	0
-----------------------	---	----------------------	---	-----------	---	-------------	---

Установка нуля

Вперед

Влево

Назад

Вправо

Торцевание

Длина обработки

0

Глубина обработки

0

Начать торцевание

Обработка поверхности

Длина обработки

0

Глубина обработки

0

Начать обработку

Нарезание резьбы

Длина резьбы. мм

0

Размер резьбы

0

Шаг резьбы

1.5

Длина обработки

0

Глубина обработки

0

Начать нарезание

Метрическая

Дюймовая

Заккрыть защитные кожухи

Графики

Рисунок 98 – Тестирование торцевания

vis1

ОШИБКА

Диаметр заготовки. мм	2	Длина торцевания. мм	0.5	Длина. мм	0	Глубина. мм	0
-----------------------	---	----------------------	-----	-----------	---	-------------	---

Установка нуля

Вперед

Влево

Назад

Вправо

Торцевание

Длина обработки

0

Глубина обработки

0

Начать торцевание

Обработка поверхности

Длина обработки

0

Глубина обработки

0

Начать обработку

Нарезание резьбы

Длина резьбы. мм

0

Размер резьбы

2

Шаг резьбы

0.4

Длина обработки

0

Глубина обработки

0

Начать нарезание

Метрическая

Дюймовая

Заккрыть защитные кожухи

Графики

Рисунок 99 – Тестирование торцевания

Длина торцевания. мм	0.5
<div style="background-color: #00FF00; border: 2px solid black; padding: 10px; display: inline-block;">Торцевание</div>	
Длина обработки	0.1
Глубина обработки	0.7
Начать торцевание	

Рисунок 100 – Тестирование торцевания

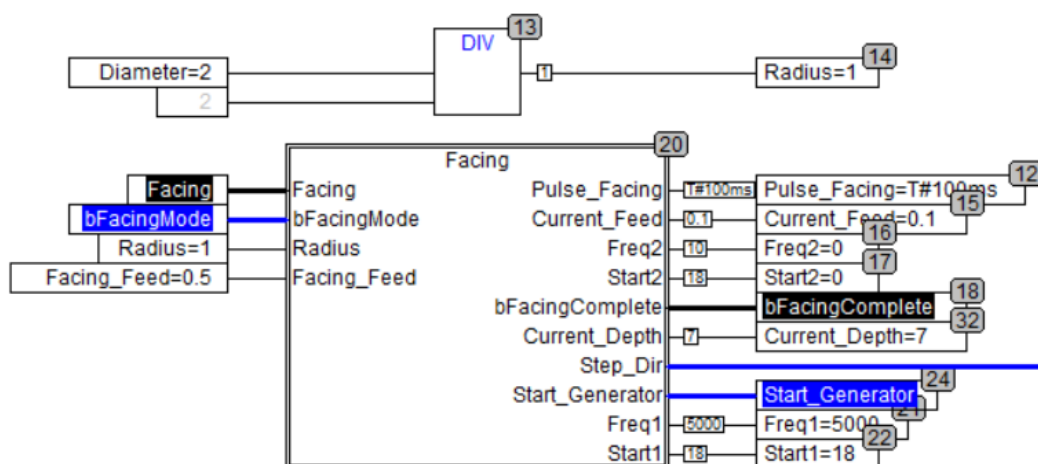


Рисунок 101 – Тестирование торцевания

Длина торцевания. мм	0.5
<div style="background-color: #00FF00; border: 2px solid black; padding: 10px; display: inline-block;">Торцевание</div>	
Длина обработки	0.1
Глубина обработки	0.5
Начать торцевание	

Рисунок 102 – Тестирование торцевания

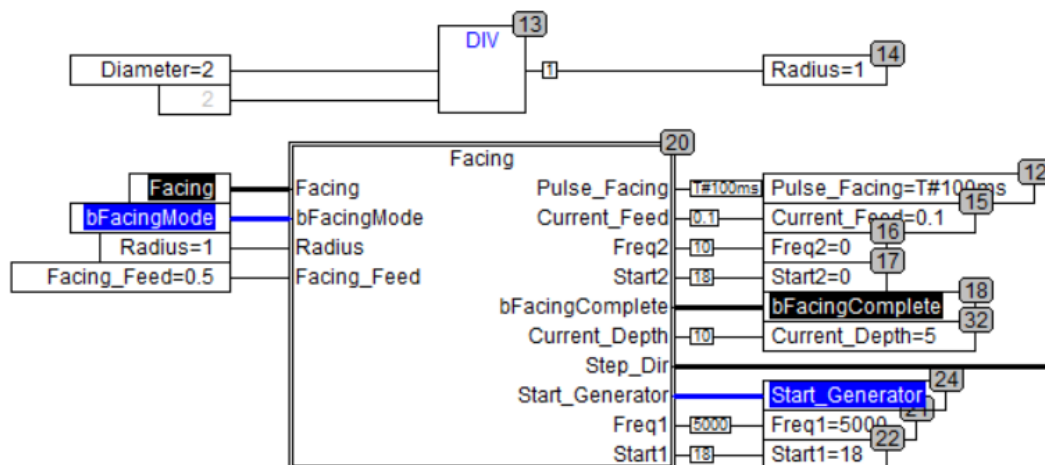


Рисунок 103 – Тестирование торцевания

Длина торцевания. мм	0.5
Торцевание	
Длина обработки	0.2
Глубина обработки	0
Начать торцевание	

Рисунок 104 – Тестирование торцевания

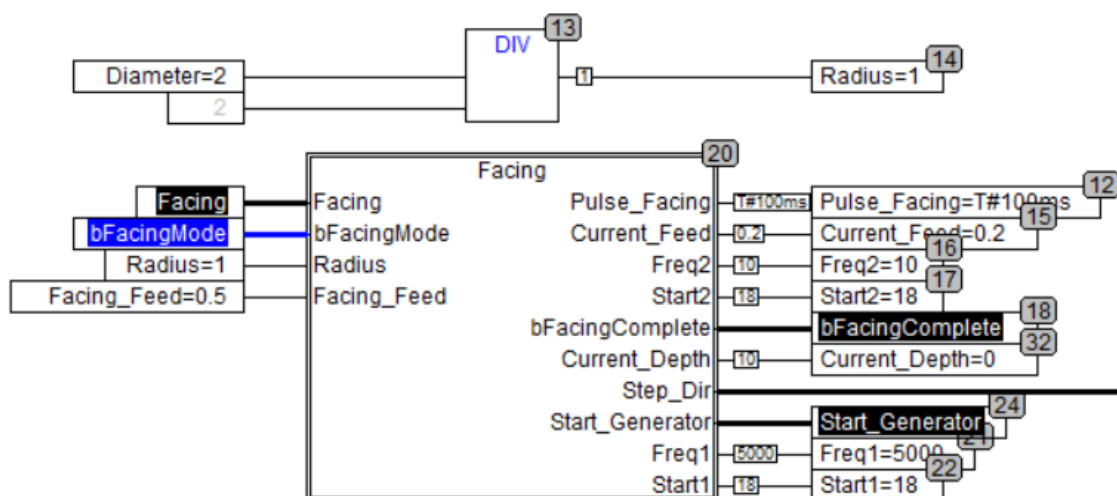


Рисунок 105 – Тестирование торцевания

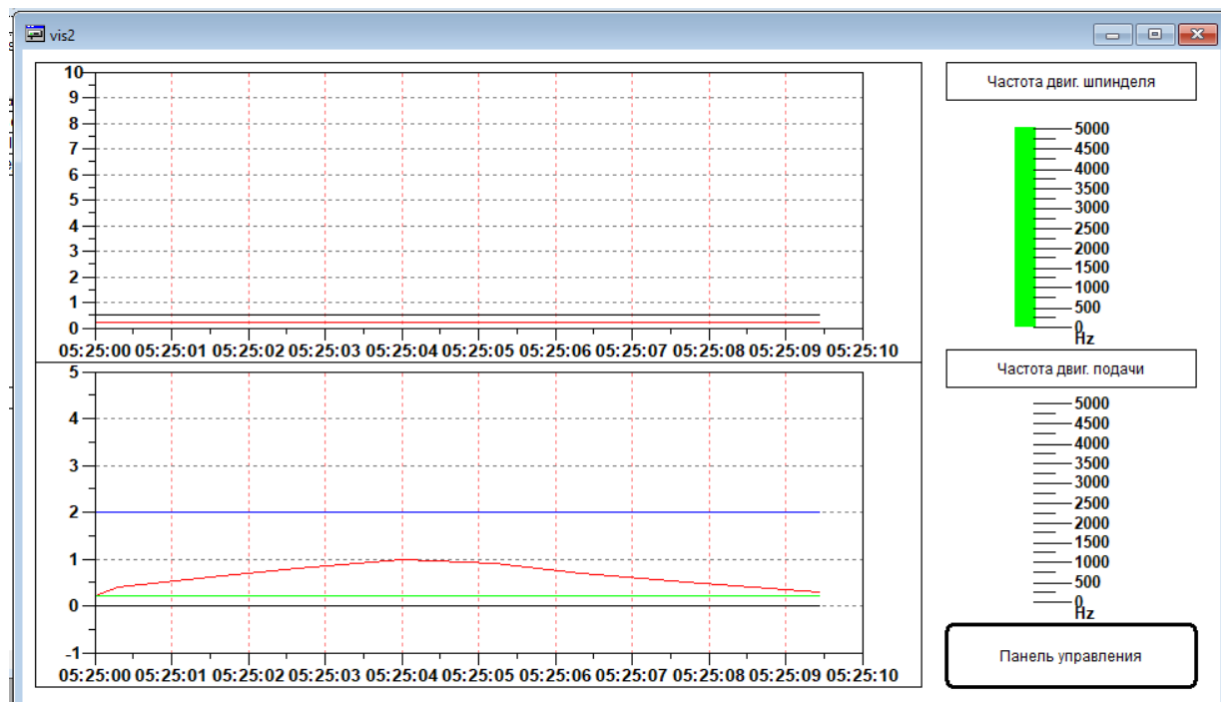


Рисунок 106 – Тестирование торцевания

Длина торцевания, мм	0.5
<div style="background-color: #00FF00; width: 150px; height: 40px; margin: 0 auto; border: 2px solid black; display: flex; align-items: center; justify-content: center;"> Торцевание </div>	
Длина обработки	0.5
Глубина обработки	0
Начать торцевание	<div style="background-color: #FF0000; width: 40px; height: 40px; border-radius: 50%; margin: 0 auto;"></div>

Рисунок 107 – Тестирование торцевания

Далее тестируется функция обработки поверхности. При запуске режима по кнопке «Торцевание», выводятся сообщения об ошибке «Не указана длина обработки» и «Не указана глубина обработки». После ввода данных ошибки пропадают (рисунки 108-109). На первом этапе работы алгоритма работает поперечная подача, пока значение длины обработки не сравняется с заданным

(рисунки 110-111). На втором этапе поперечная подача движется на 0.1 мм назад (рисунки 112-113). На третьем этапе продольная подача движется назад, пока значение длины обработки не станет равным нулю. На четвертом этапе поперечная подача движется на 0.2 мм вперед (рисунки 114-115). Далее процесс повторяется циклически до того момента, как глубина обработки не сравняется с заданной длиной. Тогда процесс останавливается и включается красный индикатор (рисунок 116).

ОШИБКА		Не указана глубина обработки					
Диаметр заготовки. мм	0	Длина торцевания. мм	0	Длина. мм	0	Глубина. мм	0
Установка нуля		Торцевание		Обработка поверхности			
Вперед <input type="button" value="○"/> Влево Вправо		Длина обработки 0 Глубина обработки 0 Начать торцевание <input type="button" value="○"/>		Длина обработки 0 Глубина обработки 0 Начать обработку <input type="button" value="○"/>			
Нарезание резьбы		Длина резьбы. мм	0	Размер резьбы	0	Шаг резьбы	1.5
Метрическая	Дюймовая	Начать нарезание <input type="button" value="○"/>		Длина обработки	0	Глубина обработки	0
Заккрыть защитные кожухи				Графики			

Рисунок 108 – Тестирование обработки поверхности

vis1

ОШИБКА			
Диаметр заготовки. мм	0	Длина торцевания. мм	0
		Длина. мм	1.3
		Глубина. мм	0.3
Установка нуля Вперед ☐ Влево ☐ Назад ☐ Вправо ☐		Торцевание Длина обработки <input type="text" value="0"/> Глубина обработки <input type="text" value="0"/> Начать торцевание ☐	
		Обработка поверхности Длина обработки <input type="text" value="0"/> Глубина обработки <input type="text" value="0"/> Начать обработку ☐	
Нарезание резьбы Метрическая ☐ Дюймовая ☐		Длина резьбы. мм <input type="text" value="0"/> Начать нарезание ☐	Размер резьбы <input type="text" value="0"/> Шаг резьбы <input type="text" value="1.5"/> Длина обработки <input type="text" value="0"/> Глубина обработки <input type="text" value="0"/>
Заккрыть защитные кожухи		Графики	

Рисунок 109 – Тестирование обработки поверхности

Длина. мм	1.3	Глубина. мм	0.3
Обработка поверхности			
Длина обработки	1.1		
Глубина обработки	0		
Начать обработку	☐		

Рисунок 110 – Тестирование обработки поверхности

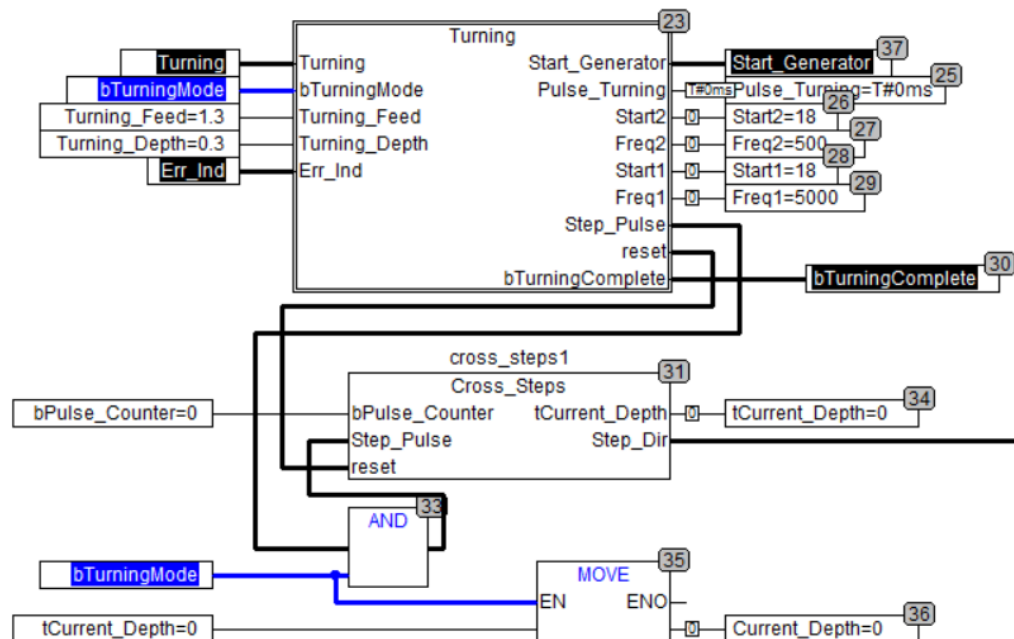


Рисунок 111 – Тестирование обработки поверхности

Длина. мм	1.3	Глубина. мм	0.3
<div style="background-color: #00FF00; border: 2px solid black; padding: 10px; margin: 10px 0;">Обработка поверхности</div>			
Длина обработки	0.5		
Глубина обработки	-0.1		
Начать обработку	<div style="border: 1px solid black; border-radius: 50%; width: 50px; height: 50px; margin: 0 auto;"></div>		

Рисунок 112 – Тестирование обработки поверхности

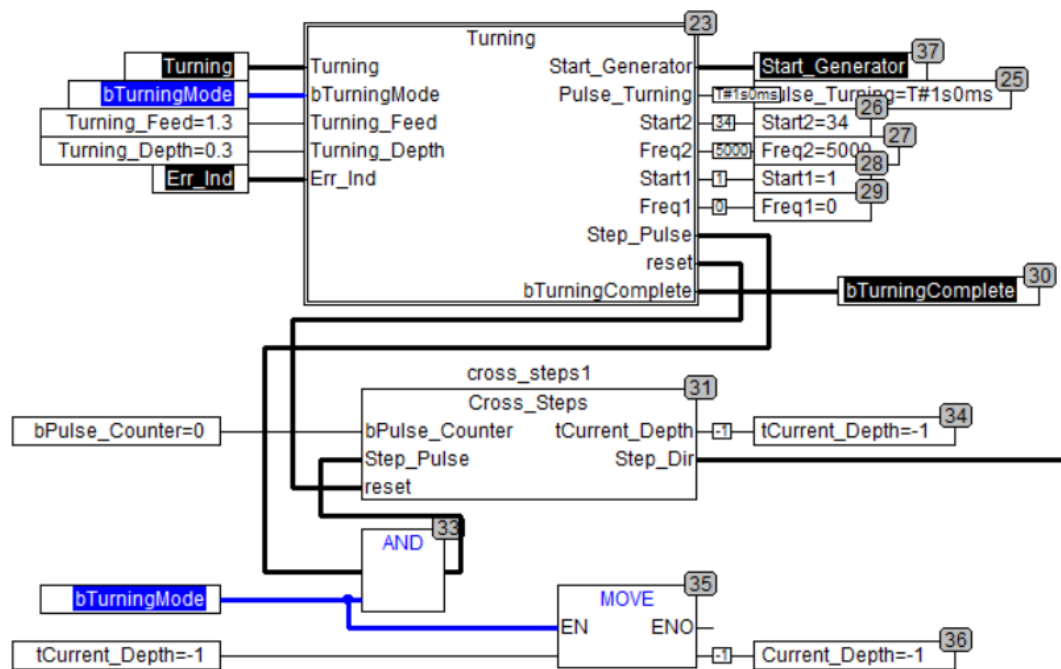


Рисунок 113 – Тестирование обработки поверхности

Длина. мм	1.3	Глубина. мм	0.3
-----------	-----	-------------	-----

Обработка поверхности

Длина обработки	0.8
Глубина обработки	0.1

Начать обработку

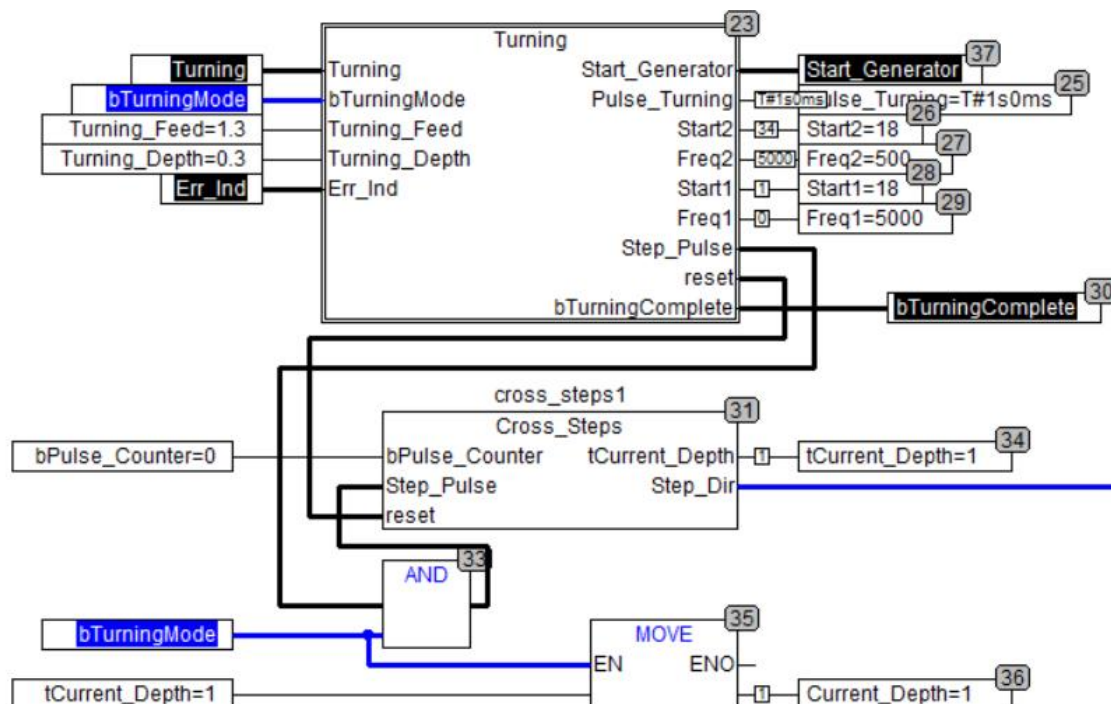


Рисунок 115 – Тестирование обработки поверхности

Длина. мм	1.3	Глубина. мм	0.3
-----------	-----	-------------	-----

Обработка поверхности

Длина обработки	0
Глубина обработки	0.4

Начать обработку

Рисунок 116 – Тестирование обработки поверхности

Далее производится тестирование режима нарезания резьбы. Он включается по кнопке «Нарезание резьбы». Кнопками «Метрическая» и «Дюймовая» выбирается тип резьбы. Алгоритм процесса полностью повторяет алгоритм

обработки поверхности, за исключением того, что глубина обработки не задается оператором, а рассчитывается программно, а процесс завершается когда длина обработки достигнет заданой длины резьбы (рисунки 117-119).

Нарезание резьбы		Длина резьбы. мм	2	Размер резьбы	3
		Метрическая	Дюймовая	Начать нарезание	
				Длина обработки	1.4
				Глубина обработки	0.1

Рисунок 117 – Тестирование нарезания резьбы

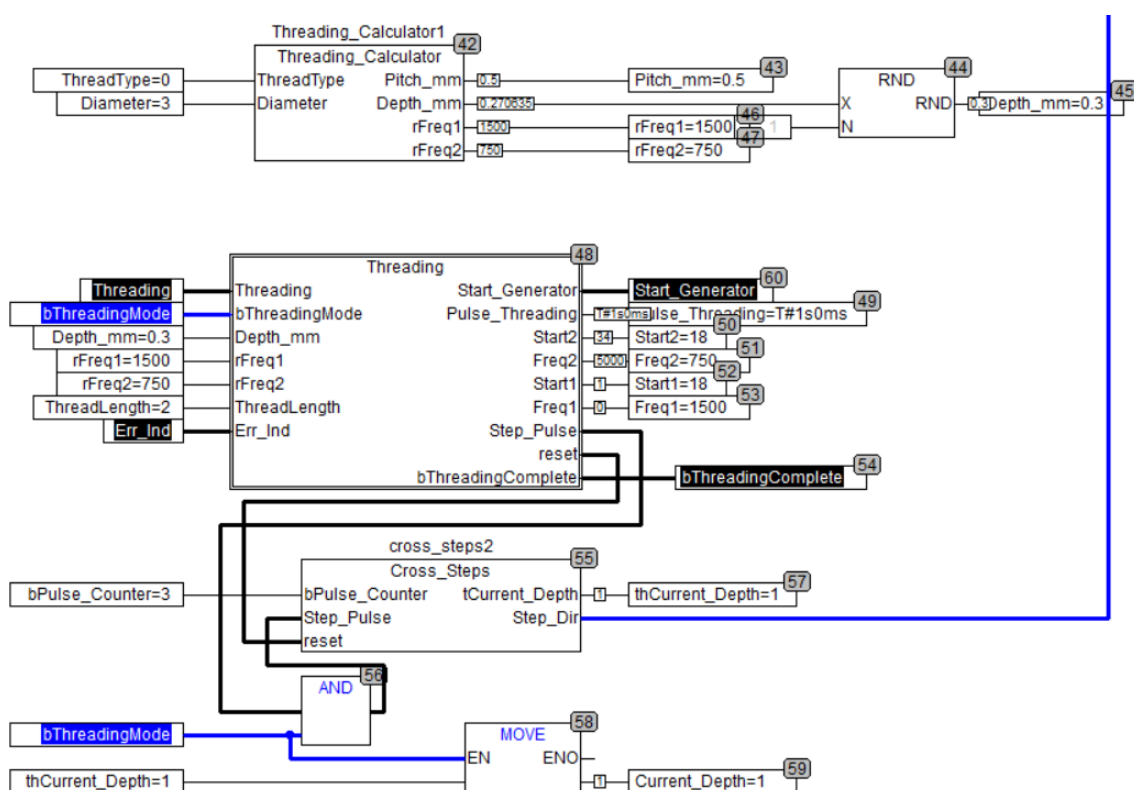


Рисунок 118 – Тестирование нарезания резьбы

Нарезание резьбы		Длина резьбы. мм	2	Размер резьбы	3
		Метрическая	Дюймовая	Начать нарезание	
				Длина обработки	0
				Глубина обработки	0.4

Рисунок 119 – Тестирование нарезания резьбы

1.3.2 Тестирование проекта на стенде целевой платформы

На этапе отладки проекта было произведено тестирование отдельных функций на стенде целевой платформы. На рисунке 120 представлены подключенные ПЛК, преобразователь частоты и электродвигатель. Управляющая программа была загружена в контроллер, осуществлялось управление исполнительным устройством через интерфейс встроенной визуализации среды разработки.

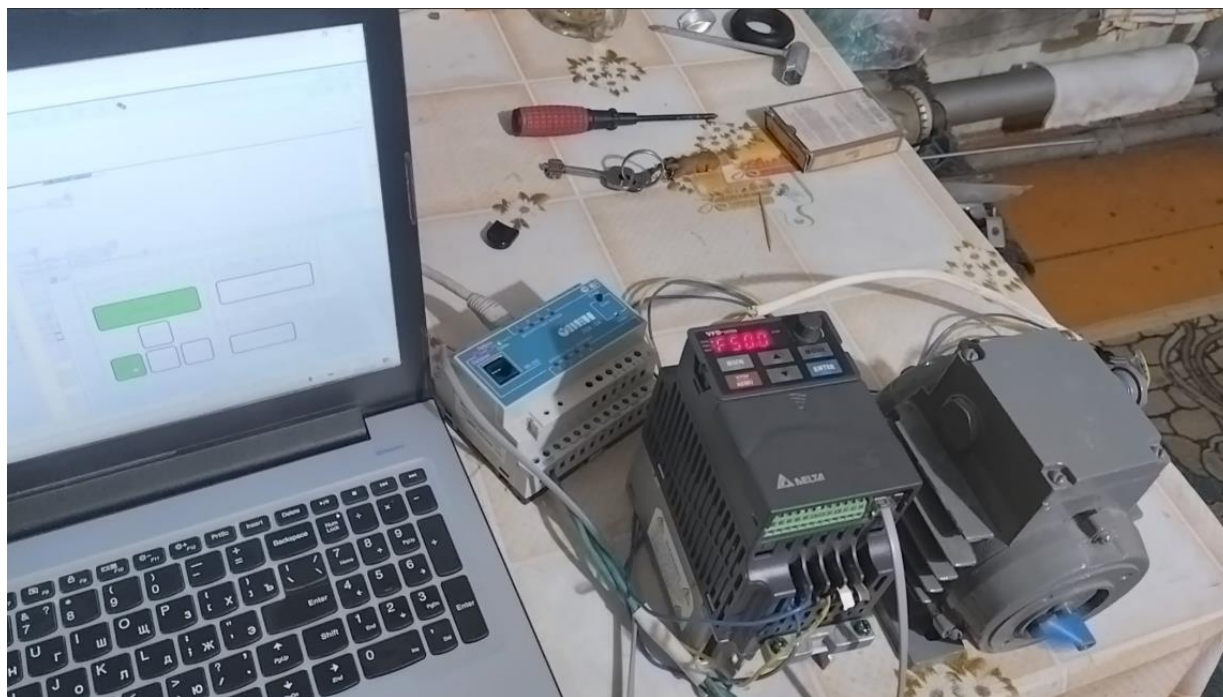


Рисунок 120 – Тестирование на стенде

2 Охрана труда

В данном разделе рассматриваются требования по обеспечению безопасности жизнедеятельности и охраны труда пользователей ПЭВМ при разработке программного обеспечения, установке и эксплуатации ПЛК и панелей оператора, а также способы минимизации пагубных факторов на здоровье.

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

Сидячий образ жизни и напряженная работа за компьютером негативно влияют на здоровье человека. Ограниченная подвижность, статическое напряжение мышц и перегрузка зрения приводят к снижению работоспособности, усталости, раздражительности, физическим недомоганиям.

Чтобы минимизировать вредное влияние компьютера, необходимо следовать следующим правилам:

- отрегулируйте высоту кресла или выберите стул, чтобы при сидении спина была ровной, а предплечья лежали на столе;
- экран должен находиться на уровне глаз, на расстоянии вытянутой руки;
- избегайте бликов и теней, используйте настольную лампу при необходимости;
- каждые 45-60 минут вставайте из-за стола, разминайтесь, делайте разминку для глаз;
- старайтесь не проводить за компьютером более 6-8 часов в день;
- держите спину ровной, не сутультесь;
- не игнорируйте правила электробезопасности – используйте заземленные розетки, не перегружайте сеть;
- соблюдайте правила пожарной безопасности, держите на рабочем месте огнетушитель, не оставляйте без присмотра включенные электроприборы.

2.1.1 Оборудование рабочего места

Правильная организация рабочего места оператора непосредственно влияет на безопасность, качество работы и сохранение здоровья специалиста. При разработке системы управления станком были учтены основные эргономические требования, предъявляемые к рабочим местам с пультами управления и экранами.

Основные параметры организации рабочего места:

- экран панели оператора (НМІ) должен находиться на расстоянии вытянутой руки — примерно 500–700 мм от глаз. Это позволяет чётко видеть все параметры обработки без напряжения зрения;

- дисплей располагается таким образом, чтобы его центр был немного ниже уровня глаз оператора — оптимальный угол взгляда вниз составляет около 15–20°. Такое положение снижает утомляемость шейного отдела позвоночника при длительной работе;

- пульт управления рекомендуется выполнять поворотным или на кронштейне, чтобы оператор мог настроить угол наклона экрана под себя и избежать бликов от цехового освещения;

- рабочая поверхность должна иметь высоту в диапазоне 700–750 мм. Это стандартная высота, позволяющая оператору свободно опираться на стол предплечьями и не сутулиться;

- органы ручного перемещения суппорта (кнопки управления осями) размещаются на уровне локтя согнутой руки — примерно 800–900 мм от пола. Это позволяет управлять перемещениями без лишнего напряжения в плечевом поясе;

- на рабочем столе обязательно выделяется место для технологической документации — не менее формата А4;

- под столешницей должно быть достаточно свободного пространства для ног: высота не менее 600 мм, глубина на уровне колен — от 450 мм, на уровне вытянутых ног — от 650 мм.

2.1.2 Освещенность рабочего места

Для рабочих мест, оборудованных дисплеями и пультами управления, рекомендуется комбинированная система освещения: общий свет плюс местная

					ПГУ 1.090501.05.001 ПЗ	Лист
						112
Изм.	Лист	№ докум.	Подпись	Дата		

подсветка. Суммарная освещённость в зоне расположения пульта должна составлять около 300 – 500 люкс. При этом одного только общего освещения допускается не менее 200 люкс, а рабочая зона станка (место резания) освещается ярче — не менее 750 люкс.

Важно обеспечить равномерное распределение света по всей рабочей поверхности. На столе и пульте не должно быть резких перепадов яркости, бликующих или зеркально отражающих предметов. Чтобы избежать излишней монотонности зрительного восприятия, допускается наличие отдельных цветовых или фактурных акцентов — например, цветной рамки на пульте или контрастной панели с кнопками. Однако такие элементы не должны создавать слепящего эффекта или отвлекать от основного процесса.

Для освещения рабочего зоны используются преимущественно потолочные или встроенные светильники с люминесцентными или светодиодными лампами.

2.1.3 Электробезопасность

Работа любой системы управления, включая разработанную, связана с использованием электрического оборудования. Вопросы электробезопасности регламентируются ГОСТ 12.1.019-2017 «Электробезопасность. Общие требования и номенклатура видов защиты».

Пульт управления, контроллер ПЛК, частотные преобразователи, двигатели и вспомогательное оборудование питаются от сети переменного тока частотой 50 Гц и напряжением 220 В. При таком напряжении поражение электрическим током представляет серьёзную угрозу для жизни человека. Поэтому при эксплуатации системы необходимо строго соблюдать комплекс мер защиты.

Основные требования безопасности:

- всё оборудование, входящее в систему управления станком, должно подключаться к одной фазе электрической сети. Это позволяет избежать разности потенциалов между корпусами устройств;
- для подключения к сети должны использоваться только розетки евростандарта с боковыми заземляющими контактами. Подключение к старым розеткам без заземления не допускается;

					ПГУ 1.090501.05.001 ПЗ	Лист
						113
Изм.	Лист	№ докум.	Подпись	Дата		

– каждая розетка, к которой подключаются элементы системы (ПЛК, ЧРП, панель оператора, насос СОЖ, освещение), должна быть промаркирована с указанием рабочего напряжения. Маркировка наносится на стену или распределительный щит рядом с розеткой яркой краской, высота символов — не менее 50 мм.

Каждый оператор и обслуживающий персонал, работающий с системой управления станком, обязан знать и строго соблюдать правила электробезопасности. Пренебрежение этими правилами может привести к тяжёлым травмам или гибели людей.

В процессе эксплуатации необходимо регулярно проверять исправность электропроводки, выключателей, розеток и соединительных кабелей. Особое внимание уделяется участкам с подвижными соединениями.

При обнаружении любых неисправностей — повреждённой изоляции, искрения, запаха гари, нагрева вилок или розеток — необходимо немедленно отключить питание оборудования и сообщить о проблеме ответственному лицу. Возобновление работы допускается только после устранения неисправности квалифицированным специалистом (электриком или инженером по обслуживанию). Самостоятельный ремонт персоналом, не имеющим соответствующей квалификации, запрещён.

2.1.4 Пожарная безопасность

Эксплуатация любого электротехнического оборудования, включая системы управления станками, сопряжена с потенциальной пожарной опасностью. Вопросы пожарной безопасности регулируются Федеральным законом от 21.12.1994 № 69-ФЗ «О пожарной безопасности».

При работе электроустановок, частотных преобразователей, ПЛК и вспомогательного оборудования наиболее часто пожар возникает по следующим причинам:

– короткое замыкание в электропроводке или внутри электрических устройств, сопровождающееся резким возрастанием тока и выделением большого количества тепла;

– нагрев и последующее воспламенение горючих материалов (пыль, масла, изоляция кабелей, пластиковые корпуса), находящихся в непосредственной близости от работающих электроустройств;

– образование большого переходного сопротивления в местах соединения проводов (ослабленные клеммы, окисленные контакты, неплотные соединения), что приводит к локальному перегреву и возгоранию.

Для обеспечения пожарной безопасности необходимо выполнить комплекс организационных мероприятий, направленных на предотвращение пожара, защиту людей, ограничение распространения огня и создание условий для его успешного тушения.

В помещении, где установлен станок и пульт управления, должны быть реализованы следующие требования:

– проходы между оборудованием, входные двери и запасные выходы должны постоянно содержаться свободными. Загромождение проходов посторонними предметами, мебелью или материалами не допускается — это препятствует быстрой эвакуации людей и доступу пожарных расчётов.

– К средствам пожаротушения (огнетушителям, пожарным кранам, ящикам с песком) должен быть обеспечен свободный и беспрепятственный доступ в любое время. Запрещается закрывать, загромождать или опечатывать доступ к ним без возможности быстрого вскрытия;

– система вентиляции помещения должна быть оборудована устройствами, обеспечивающими её автоматическое отключение при срабатывании пожарной сигнализации. Дополнительно в воздуховодах устанавливаются огнезадерживающие клапаны, предотвращающие распространение дыма и пламени по вентиляционным каналам;

– автоматическая пожарная сигнализация (датчики дыма, тепла или пламени) и стационарные средства пожаротушения должны постоянно содержаться в исправном состоянии. Их техническое обслуживание выполняется в соответствии с типовыми правилами эксплуатации установок пожарной автоматики — с

периодической проверкой работоспособности, очисткой датчиков от загрязнений и своевременной заменой элементов питания.

Применительно к разработанной системе управления токарно-винторезным станком рекомендуется также:

- размещать частотные преобразователи и блоки питания в металлическом шкафу управления с перфорацией для охлаждения, исключив попадание масляной эмульсии и металлической пыли;
- использовать кабели с негорючей обшивкой для подключения силовых цепей;
- регулярно проводить осмотр и чистку шкафа управления от пыли и стружки;
- иметь в доступном месте углекислотный огнетушитель (класса Е) для тушения электрооборудования под напряжением.

2.1.5 Режим труда и отдыха

Оператор, работающий за пультом управления станком с экраным интерфейсом (панелью оператора или ПЭВМ), должен строго соблюдать режим труда и отдыха. Правильное чередование работы и перерывов позволяет снизить утомляемость, сохранить работоспособность на протяжении всей смены, предотвратить развитие профессиональных заболеваний и избежать ошибок, связанных с потерей концентрации внимания.

Вся работа, связанная с использованием экранных устройств (дисплеев, панелей НМІ), делится на три группы в зависимости от характера выполняемых задач:

- группа А — работа, связанная преимущественно со считыванием информации с экрана. Это может быть контроль параметров обработки, просмотр графиков или чтение технологических карт, отображаемых на экране;
- группа Б — работа, связанная с вводом информации. Сюда относится задание параметров обработки, ввод корректирующих значений, заполнение журналов или протоколов;

– группа В — творческая или диалоговая работа с системой. Например, разработка нового цикла обработки, подбор оптимальных режимов резания, анализ ошибок или нештатных ситуаций.

Если в течение рабочей смены оператор выполняет несколько видов деятельности, основным считается тот, который занимает не менее 50% всего времени работы с экраном. Именно под него подбираются нормативы нагрузки и перерывов.

В зависимости от суммарного объёма обработанной информации за смену устанавливаются категории тяжести и напряжённости работы:

– для группы А (считывание информации) предельный объём составляет не более 60 000 знаков за смену. Превышение этого показателя ведёт к перенапряжению зрительного аппарата и требует дополнительных перерывов или снижения нагрузки;

– для группы Б (ввод информации) предельный объём составляет не более 40 000 знаков за смену. Сюда входят как вводимые с клавиатуры символы, так и считываемые с экрана при контроле ввода.

Наиболее эффективным является следующий режим:

– через каждый час работы рекомендуется делать короткий перерыв на 5–10 минут: отвлечься от экрана, посмотреть вдаль, сменить позу;

– после двух часов непрерывной работы — перерыв 10–15 минут с обязательным вставанием из-за пульта;

– в середине смены — обеденный перерыв продолжительностью не менее 30–40 минут.

3 Экономика

Исходные данные для расчетов представлены в таблице 4.

Таблица 4 – исходные данные для расчета

Параметр	Значение
Должность исполнителя	Инженер-программист ПЛК (full-stack: Codesys 2.3 + SCADA/HMI)
Часовая тарифная ставка	1 200 руб./час (с учётом страховых взносов и накладных расходов)
Продолжительность рабочего дня	8 часов
Общий срок выполнения работ	3 календарных месяца
Режим работы	Полный рабочий день (21 рабочий день в месяц)

В таблице 5 представлены трудозатраты.

Таблица 5 – Трудозатраты

Наименование этапа	Трудозатраты, чел.-час	Стоимость, руб.
Анализ требований к системе управления станком (технологические операции, режимы резания, требования к резьбам)	24	28 800
Проектирование архитектуры ПЛК-программы, разработка схемы входов/выходов (I/O mapping)	32	38 400
Разработка глобальных переменных и конфигурации адресного пространства (VAR CONFIG)	16	19 200
Разработка функциональных блоков (FBD/ST): управление шпинделем, подачей, шаговым двигателем	56	67 200
Разработка модуля нарезания резьбы (расчётный блок на ST + CFC-программа)	40	48 000
Разработка модуля ручного управления и привязки к нулю (State Machine)	32	38 400

Наименование этапа	Трудозатраты, чел.-час	Стоимость, руб.
Реализация системы безопасности (защитные кожухи, концевики, тепловые реле, аварийный стоп)	24	28 800
Интеграция с частотными преобразователями (настройка аналоговых выходов 0–10В)	16	19 200
Разработка визуализации (HMI/SCADA) для задания параметров обработки и контроля процесса	48	57 600
Комплексное тестирование, отладка, пуско-наладка на станке	40	48 000
Разработка эксплуатационной документации (руководство оператора, схемы подключения)	24	28 800
Итого	352	422 400

В таблице 6 представлена стоимость аппаратной части.

Таблица 6 – Стоимость аппаратной части

Наименование	Кол-во	Цена за ед., руб.	Сумма, руб.
ПЛК 154И М (контроллер)	1	35 000	35 000
Частотный преобразователь шпинделя (3 кВт)	1	18 000	18 000
Частотный преобразователь подачи (1.5 кВт)	1	12 000	12 000
Драйвер шагового двигателя	1	5 000	5 000
Шаговый двигатель (поперечная подача)	1	6 500	6 500
Блок питания 24В (5А)	1	3 500	3 500
Промежуточные реле (8 шт)	8	300	2 400
Контакты (шпиндель, подача)	4	1 200	4 800
Тепловые реле	2	1 500	3 000
Концевые выключатели (NC)	4	400	1 600
Датчики нуля (индуктивные)	2	1 200	2 400
Кнопки и переключатели (пульт)	1 компл.	4 000	4 000
Защитный кожух с датчиком	1	5 000	5 000
Шкаф управления (IP54)	1	8 000	8 000
Кабельная продукция, клеммники	1 компл.	7 000	7 000
Итого материалы			118 200

В таблице 7 представлен годовой экономический эффект.

Таблица 7 – Годовой экономический эффект

Показатель	До модернизации	После модернизации	Экономия
Оплата труда оператора	666 500 руб.	183 500 руб.	483 000 руб.
Показатель	До модернизации	После модернизации	Экономия
Потери от брака	90 000 руб.	15 000 руб.	75 000 руб.
Итого	756 500 руб.	198 500 руб.	558 000 руб./год

Процент снижения затрат: 73,8%

При общих затратах на разработку и внедрение технологии 564200 руб./год и экономическом эффекте 558000 руб./год срок окупаемости составит один год.

Заключение

В результате выполнения данной выпускной квалификационной работы разработана и реализована система числового программного управления токарно-винторезным станком на базе промышленного контроллера ПЛК 154И-М и среды программирования CODESYS 2.3.

В ходе исследования были детально проанализированы технологические требования к токарной обработке, включая операции торцевания, продольного точения и нарезания резьб. Выявлены ключевые ограничения традиционного ручного управления: высокая зависимость качества обработки от квалификации оператора, сложность получения точных резьб с заданным шагом, отсутствие активной защиты от перегрузок и недостаточная повторяемость результатов. На основе этого анализа сформулированы технические требования к разрабатываемой системе.

В процессе проектирования были рассмотрены два подхода к управлению приводами станка: с использованием частотных преобразователей и прямое контакторное управление. Выбран вариант с двумя частотными преобразователями для главного двигателя шпинделя и двигателя продольной подачи, а также шаговым двигателем для поперечной подачи. Такое решение обеспечило плавное регулирование скорости резания, возможность реализации постоянной скорости резания (CSS) и высокую точность позиционирования.

В рамках работы спроектирована и реализована следующая функциональность системы:

- режим ручного поиска нулевой точки с возможностью перемещения суппорта по осям на малой и большой скорости, фиксацией нулевых координат по нажатию кнопки оператором и последующей привязкой к ним всех технологических циклов;
- автоматическое торцевание заготовки с заданной глубиной и подачей, рассчитываемой на основе оборотов шпинделя;

- автоматическая продольная обработка поверхности с поддержанием постоянной скорости резания и контролем достижения заданного диаметра и длины;
- нарезание метрических и дюймовых резьб по ГОСТ 8724–2002 [5] и стандартам UNC/UNF с автоматическим расчётом шага, глубины, количества проходов и глубины каждого прохода методом уменьшающейся стружки;
- синхронизация скоростей шпинделя и продольной подачи по формуле, обеспечивающая точное соответствие перемещения суппорта шагу резьбы без использования энкодеров — только за счёт согласования частотных преобразователей;
- система безопасности, включающая контроль закрытия защитных кожухов, опрос конечных выключателей продольной подачи, мониторинг тепловых реле шпинделя и подачи, а также аппаратный аварийный стоп с мигающей индикацией;
- функции включения насоса СОЖ и подсветки рабочей зоны по команде оператора.

Программная реализация выполнена с использованием следующих языков стандарта МЭК 61131–3: CFC (Continuous Function Chart) для организации конечного автомата состояний станка и параллельных процессов регулирования, Structured Text (ST) для реализации расчётных алгоритмов резьбы и функциональных блоков управления частотными преобразователями. Глобальные переменные сконфигурированы через раздел VAR_CONFIG, что обеспечило разделение логики программы и аппаратной привязки адресов ввода-вывода.

Разработанная система позволяет оператору через интерфейс визуализации задавать тип резьбы (метрическая или дюймовая), диаметр, длину, а также параметры торцевания и продольной обработки. Все рассчитанные режимы резания (обороты шпинделя, скорость подачи, количество проходов, глубина каждого прохода) отображаются оператору перед началом цикла.

Экономический раздел работы показал, что внедрение разработанной системы управления позволяет сократить годовые затраты на оплату труда

					ПГУ 1.090501.05.001 ПЗ	Лист
						122
Изм.	Лист	№ докум.	Подпись	Дата		

оператора и потери от брака на 558 000 рублей, а срок окупаемости проекта составляет 12 месяцев при общих инвестициях 564 240 рублей.

Созданная система автоматизации токарно-винторезного станка призвана повысить производительность труда, обеспечить стабильное качество обработки, особенно при нарезании резьб, и снизить требования к квалификации персонала. Разработанное программное обеспечение полностью готово к эксплуатации на производственных участках машиностроительных и ремонтных предприятий. Перспективными направлениями развития системы являются добавление библиотеки типовых резьб всех стандартов, реализация функции адаптивного управления подачей по току двигателя шпинделя, а также организация передачи протоколов обработки в верхний уровень АСУ предприятия.

					ПГУ 1.090501.05.001 ПЗ	Лист
						123
Изм.	Лист	№ докум.	Подпись	Дата		

Список использованных источников

1. ГОСТ ИЕС 61131-2-2012. Контроллеры программируемые. Часть 2. Требования к оборудованию и испытания. – Введ. 2014-01-01. – М.: Стандартиформ, 2014. – 198 с.
2. ГОСТ 18097-93. СТАНКИ ТОКАРНО-ВИНТОРЕЗНЫЕ И ТОКАРНЫЕ Основные размеры. Нормы точности – Введ. 1996-07-01. – М.: ИПК Издательство стандартов, 1996. – 19 с.
3. ГОСТ 8724-2002. Основные нормы взаимозаменяемости. РЕЗЬБА МЕТРИЧЕСКАЯ Диаметры и шаги. – Введ. 2004-01-01. – М.: ИПК Издательство стандартов, 2003. – 7 с.
4. Микропроцессорный контроллер ПЛК154. Руководство по эксплуатации. – М.: ОВЕН, 2022. – 60 с. – URL: http://www.owen.ru/uploads/rie_plk154_983.pdf (дата обращения: 20.05.2026).
5. Программируемые логические контроллеры: практический подход к стандарту ИЕС 61131-3 с использованием CODESYS / Д. Х. Ханссен ; пер. с англ. – М.: ДМК Пресс, 2019. – 416 с.
6. CoDeSys V2.3. Руководство пользователя. Программирование в среде CoDeSys 2.3 [Электронный ресурс] // 3S Software. – URL: <https://help.codesys.com/> (дата обращения: 15.05.2026).
7. CoDeSys V2.3. Визуализация. Дополнение к руководству пользователя [Электронный ресурс] // 3S Software. – URL: <https://help.codesys.com/> (дата обращения: 15.05.2026).
8. Конфигурирование периферийного оборудования ПЛК ОВЕН в среде CoDeSys. Руководство пользователя [Электронный ресурс] // ОВЕН. – URL: <http://www.owen.ru/> (дата обращения: 18.05.2026).
9. ГОСТ 12.1.019-2017. Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты. – Введ. 2019-01-01. – М.: Стандартиформ, 2018. – 12 с.

10. ГОСТ 12.2.007.0-75. Система стандартов безопасности труда (ССБТ). Изделия электротехнические. Общие требования безопасности. – Введ. 1977–01–01. – М.: ИПК Издательство стандартов, 2001. – 12 с.

11. Правила технической эксплуатации электроустановок потребителей [Электронный ресурс] // URL: <https://www.consultant.ru/> (дата обращения 15.05.2026).

12. Правила по охране труда при эксплуатации электроустановок [Электронный ресурс] // URL: <https://www.consultant.ru/> (дата обращения 16.05.2026).

13. Федеральный закон от 21.12.1994 № 69-ФЗ "О пожарной безопасности" (ред. от 28.09.2010) [Электронный ресурс] // URL: <https://www.consultant.ru/> (дата обращения 17.05.2026).

					ПГУ 1.090501.05.001 ПЗ	Лист
						125
Изм.	Лист	№ докум.	Подпись	Дата		

Приложение А

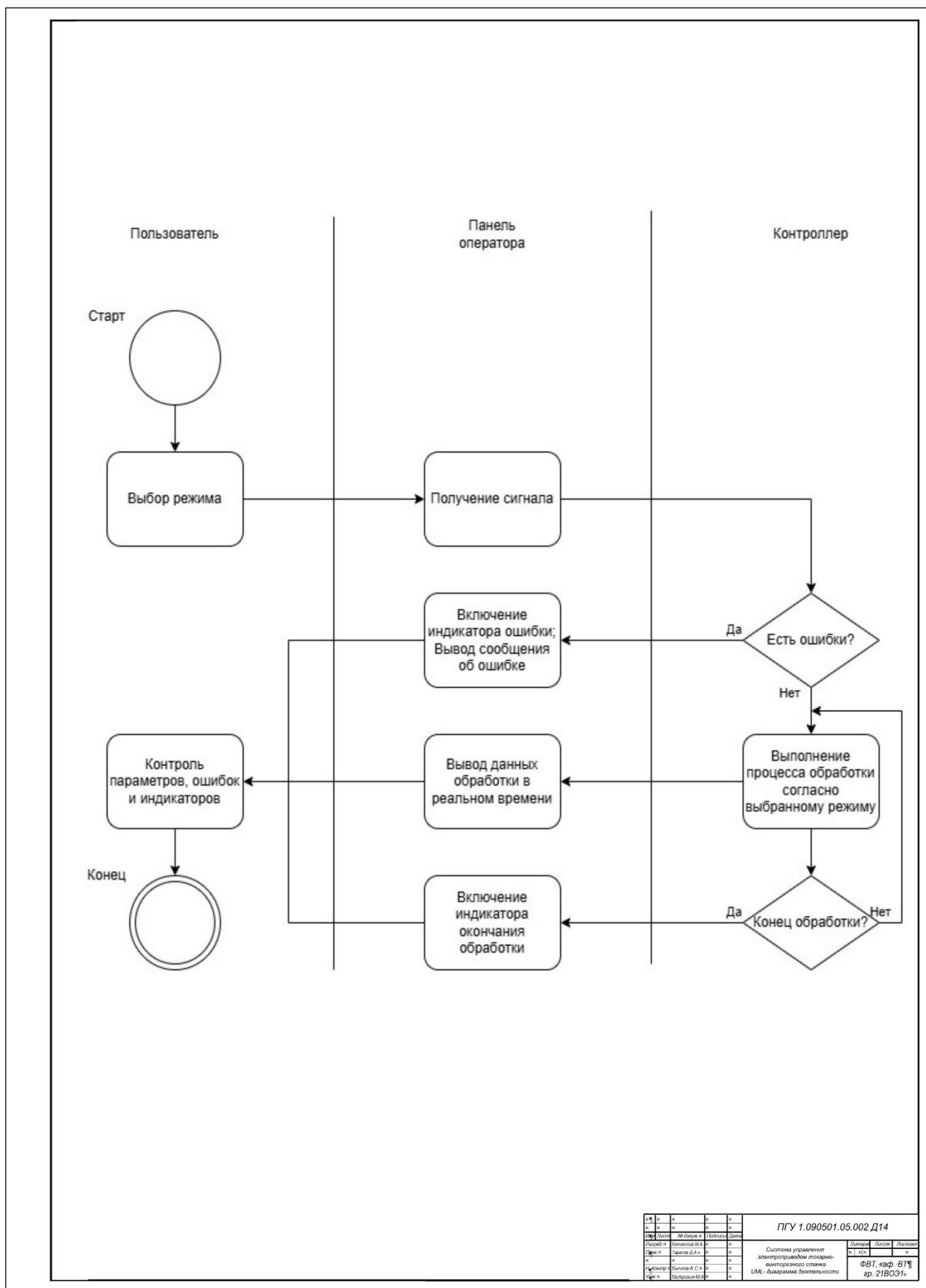
(обязательное)

Система управления электроприводом токарно-винторезного станка

UML-диаграммы

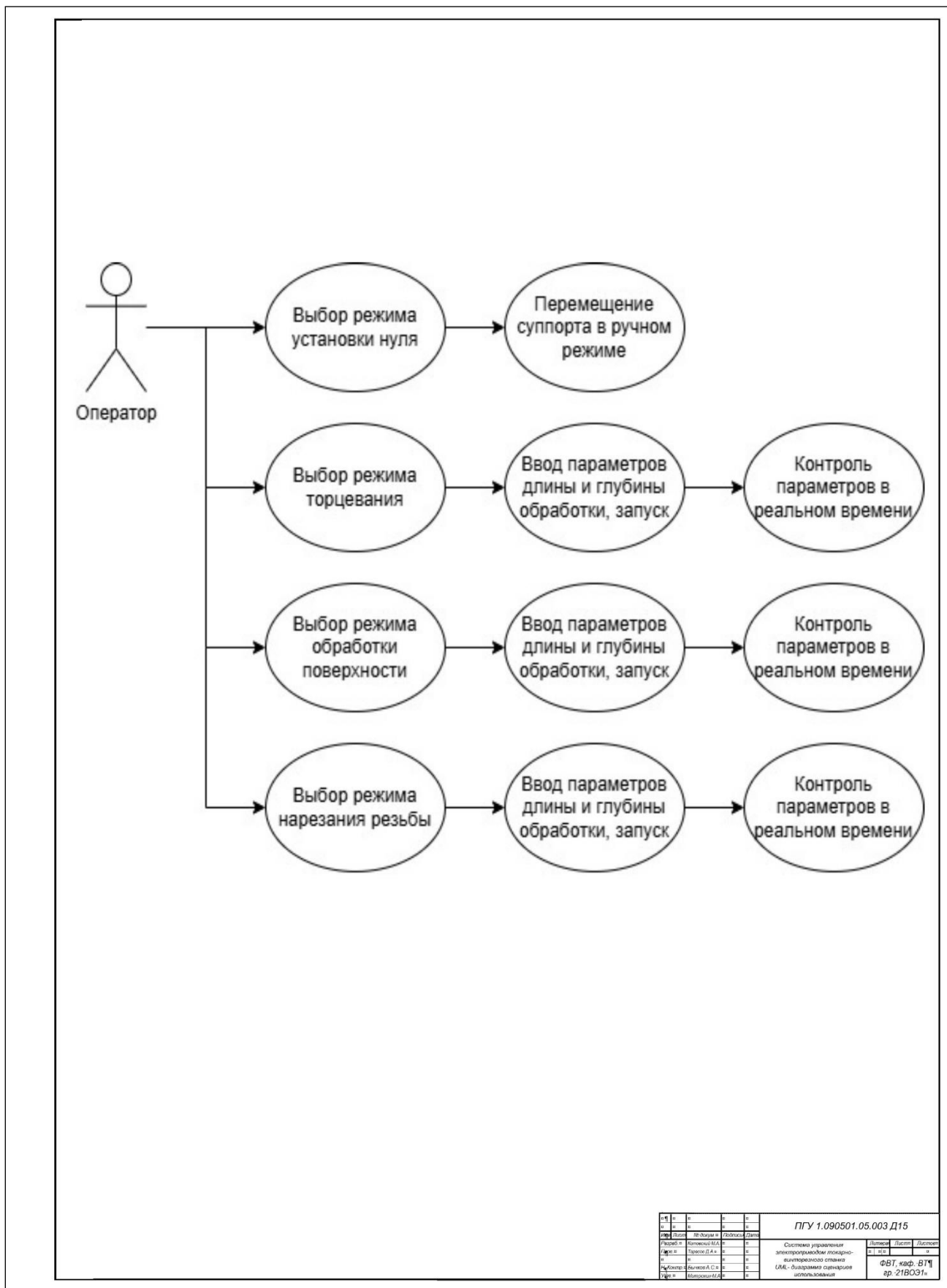
					ПГУ 1.090501.05.001 ПЗ	Лист
						126
Изм.	Лист	№ докум.	Подпись	Дата		

UML – диаграмма деятельности



Изм.	Лист	№ докум.	Подпись	Дата

UML – диаграмма вариантов использования



Приложение Б

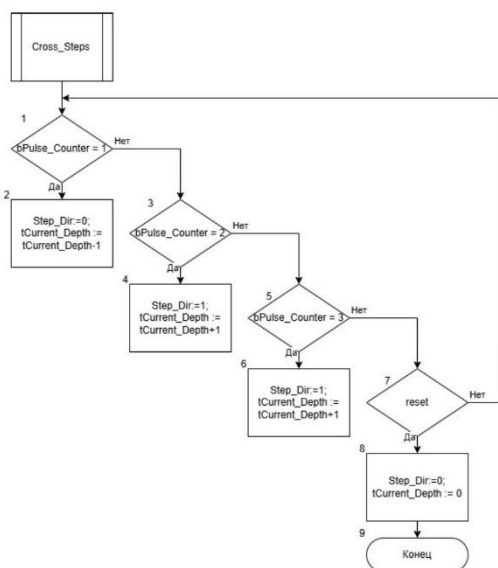
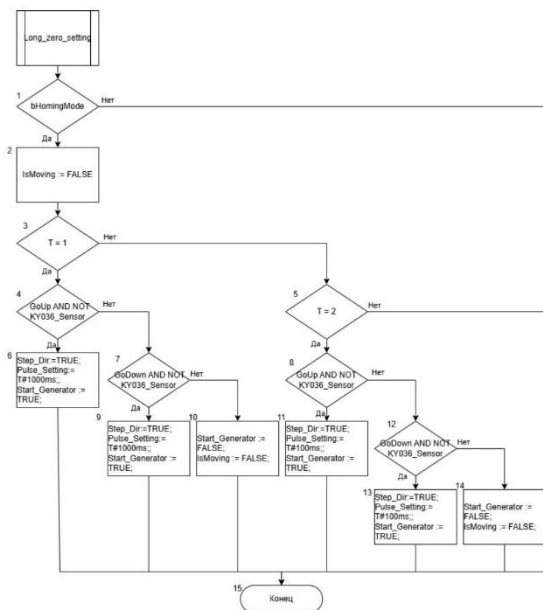
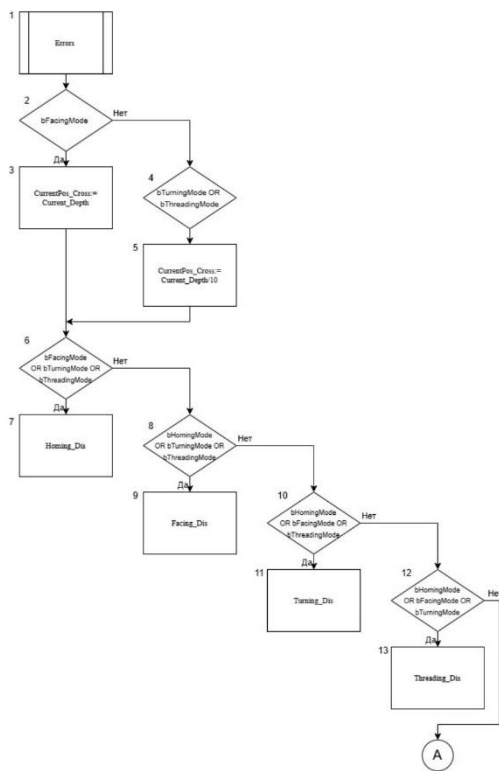
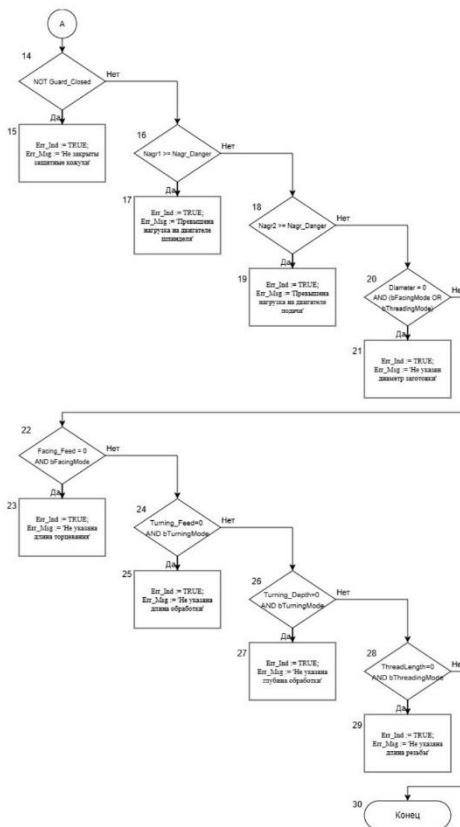
(обязательное)

Система управления электроприводом токарно-винторезного станка

Схемы программы

					ПГУ 1.090501.05.001 ПЗ	Лист
						129
Изм.	Лист	№ докум.	Подпись	Дата		

Схемы программы



№ п/п	Ф.И.О.	Подпись	Дата	Подпись	Дата
1	Иванов И.И.				
2	Петров П.П.				
3	Сидоров С.С.				
4	Кузнецов К.К.				
5	Лебедев Л.Л.				
6	Зайцев З.З.				
7	Васильев В.В.				
8	Попов П.П.				
9	Морозов М.М.				
10	Михайлов М.М.				

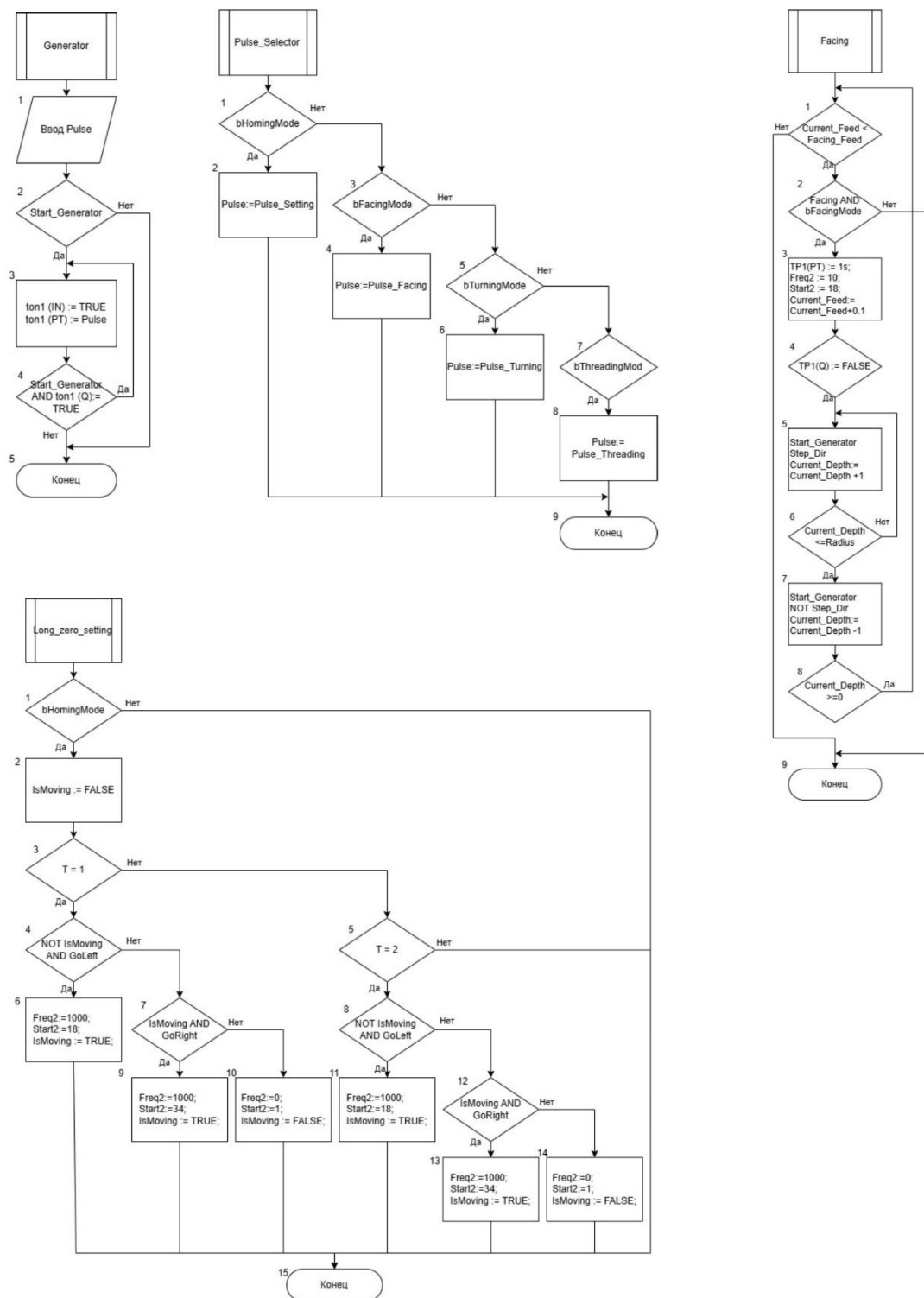
ПГУ 1.090501.05.004 Д16

Система управления
электронным документооборотом
электронного стиля
Схемы программ

Листов 1
Всего 1

ФВТ, каф. ВТ
ар. 218031

Схемы программы



№ п/п	И.О. Фамилия	Подпись	Дата	ПГУ 1.090501.05.005 Д17
Разработчик	Киселев М.А.			Система управления электроприводами токарно-винторезного станка
Проверен	Белых А.С.			Схемы программы
Утвержден	Михайлов И.В.			ФБТ, каф. БТИ пр. 21В031

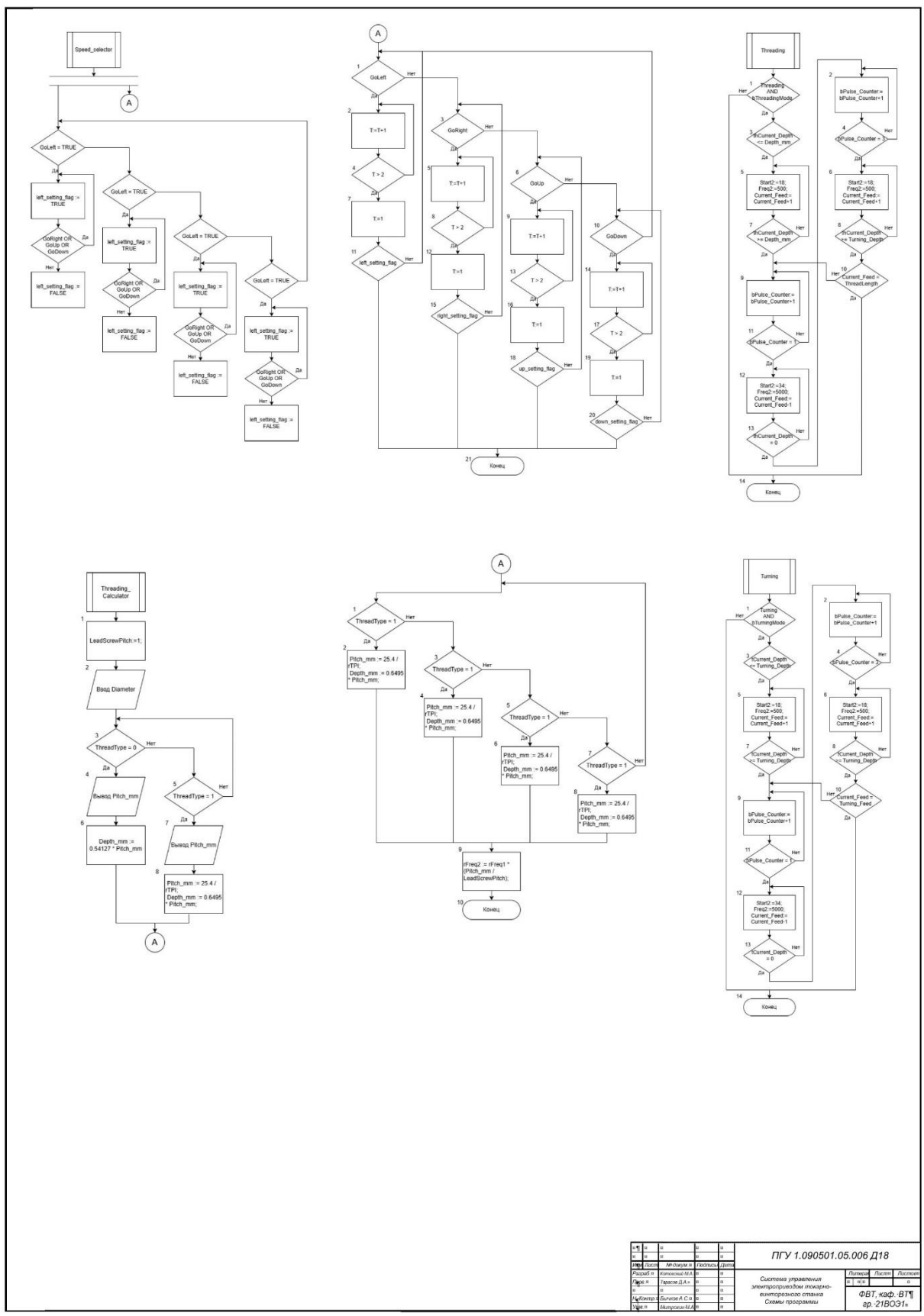
Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.05.001 ПЗ

Лист

131

Схемы программы



Приложение В

(обязательное)

Система управления электроприводом токарно-винторезного станка

Презентация

					ПГУ 1.090501.05.001 ПЗ	Лист
						133
Изм.	Лист	№ докум.	Подпись	Дата		

СИСТЕМА УПРАВЛЕНИЯ ЭЛЕКТРОПРИВОДОМ ТОКАРНО- ВИНТОРЕЗНОГО СТАНКА

Котовский Максим Александрович
Студент 21В0Э1

Цель:

- Основная цель исследования заключается в разработке системы автоматизированного управления электроприводом токарно-винторезного станка на базе программируемого логического контроллера. Разрабатываемая система обеспечивающего автоматическое выполнение процессов обработки, отслеживание аварий и ошибок и контроль параметров в реальном времени.

Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.05.001 ПЗ

Лист

134

В процессе разработки должны быть решены следующие задачи:

- Основная задача разработки заключается в создании автоматизированной системы управления, которое обеспечивает надежное и точное управление электроприводом токарно-винторезного станка, автоматическое выполнение основных функций обработки по заданным параметрам, защищенность от ошибок во время работы, удобный графический интерфейс панели оператора и возможность дальнейшего развития. Система должна повысить эффективность процессов металлообработки, выполняемых на токарно-винторезных станках и повысить их производительность.

Аналоги:

- Ручные станки с релейно-контакторной системой управления и коробкой подач
- Станки с системой управления «ПЛК+ПЧ»
- Станки ЧПУ

Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.05.001 ПЗ

Лист

135

Станок 16К20 с релейно-контакторной системой управления и коробкой подач



Система «ПЛК + ПЧ»



Система «ПЛК + ПЧ»



ЧПУ токарно-винторезный станок



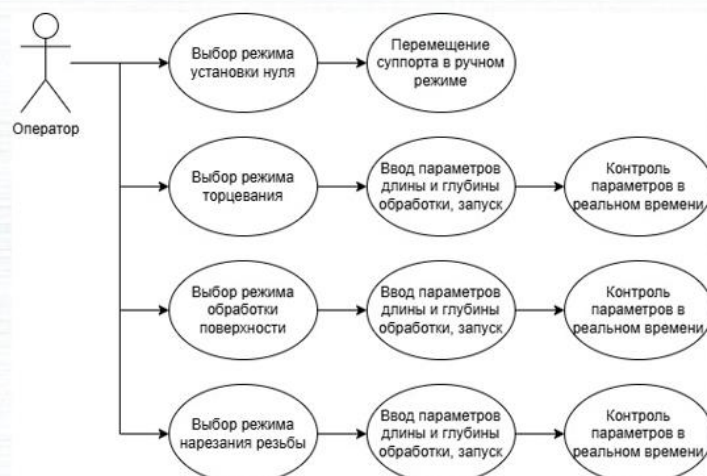
Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.05.001 ПЗ

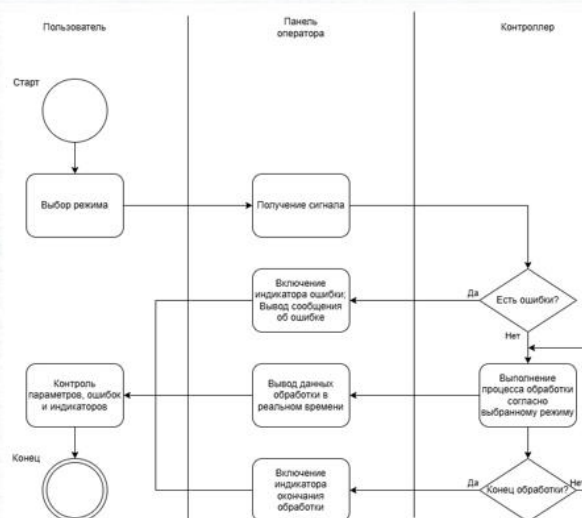
Лист

137

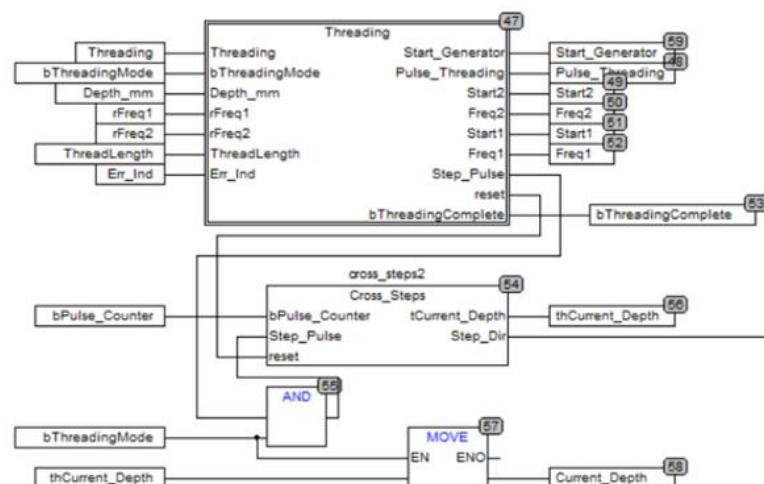
Варианты использования системы для оператора



Деятельность оператора

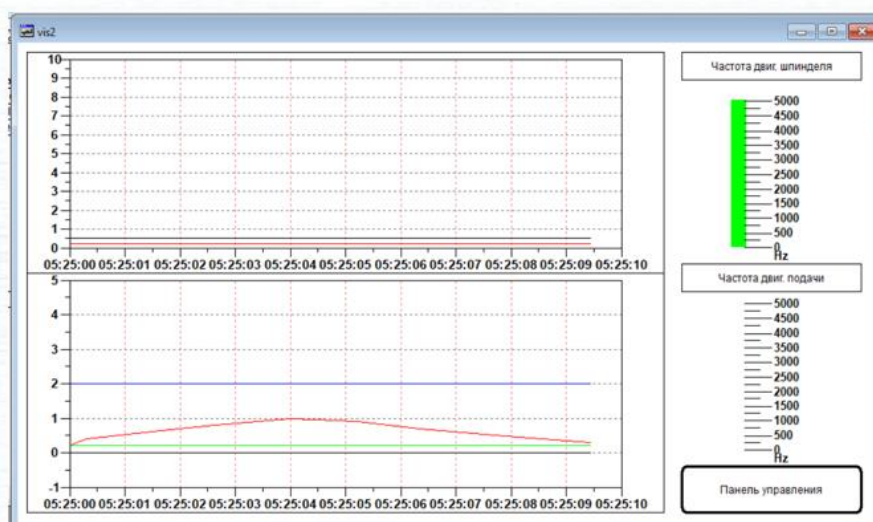


Управляющая программа



Интерфейс панели оператора

Интерфейс панели оператора



Заключение

- В процессе разработки реализованы основные функции системы: ручное управление подачей, автоматическое выполнение процесса обработки торца, поверхности и нарезания резьбы, также функция контроля ошибок и вывод данных в реальном времени.
- Систему управления можно улучшить, добавляя различные функции для выполнения узкоспециализированных задач.

Изм.	Лист	№ докум.	Подпись	Дата

ПГУ 1.090501.05.001 ПЗ

Лист

140