



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Магнитогорский государственный технический университет им. Г.И. Носова»



УТВЕРЖДАЮ
Директор ИЭиАС
С.И. Лукьянов

26.02.2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ УПРАВЛЕНИЯ В ТЕХНИЧЕСКИХ СИСТЕМАХ

Направление подготовки (специальность)
27.04.04 УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ

Направленность (профиль/специализация) программы
Автоматизация технологических процессов и производств

Уровень высшего образования - магистратура
Программа подготовки - академическая магистратура

Форма обучения
очная

Институт/ факультет	Институт энергетики и автоматизированных систем
Кафедра	Автоматизированных систем управления
Курс	2
Семестр	3

Магнитогорск
2020 год

Рабочая программа составлена на основе ФГОС ВО по направлению подготовки 27.04.04 УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ (уровень магистратуры) (приказ Минобрнауки России от 30.10.2014 г. № 1414)

Рабочая программа рассмотрена и одобрена на заседании кафедры Автоматизированных систем управления
12.02.2020, протокол № 6

Зав. кафедрой _____ С.М. Андреев

Рабочая программа одобрена методической комиссией ИЭиАС
26.02.2020 г. протокол № 5

Председатель _____ С.И. Лукьянов

Рабочая программа составлена:
доцент кафедры АСУ, канд. техн. наук _____ Е.С. Рябчикова

Рецензент:
зам. директора ЗАО "КонсОМ СКС" , канд. техн. наук
_____ Ю.Н. Волщук



Лист актуализации рабочей программы

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2020 - 2021 учебном году на заседании кафедры Автоматизированных систем управления

Протокол от 02 сентября 2020 г. № 1
Зав. кафедрой  С.М. Андреев

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2021 - 2022 учебном году на заседании кафедры Автоматизированных систем управления

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____ С.М. Андреев

1 Цели освоения дисциплины (модуля)

Целями освоения дисциплины «Компьютерные технологии управления в технических системах» являются:

- формирование у обучающегося способности использовать на практике умения и навыки в организации исследовательских и проектных работ с помощью применения компьютерных технологий;
- формирование у обучающегося способности применять современные методы разработки технического, информационного и алгоритмического обеспечения систем автоматизации и управления с помощью применения компьютерных технологий;
- формирование у обучающегося целостного представления о специфике и возможностях современных компьютерных технологий управления в технических системах.

Для достижения поставленной цели в дисциплине «Компьютерные технологии управления в технических системах» решаются задачи:

- определение места компьютерных технологий в системах автоматического управления производственными процессами;
- знакомство с основными компьютерными технологиями управления, используемыми в технических системах;
- получение навыков использования компьютерных технологий в процессе технического творчества и научного эксперимента.

2 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина Компьютерные технологии управления в технических системах входит в базовую часть учебного плана образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Аппаратные средства и программное обеспечение микропроцессорных технологических контроллеров

Агрегатные комплексы технических средств

Аппаратно- программные комплексы систем автоматизации

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения дисциплин/практик:

Научно-исследовательская работа

Производственная-преддипломная практика

Подготовка к защите и защита выпускной квалификационной работы

3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения

В результате освоения дисциплины (модуля) «Компьютерные технологии управления в технических системах» обучающийся должен обладать следующими компетенциями:

Структурный элемент компетенции	Планируемые результаты обучения
ОК-2	способностью использовать на практике умения и навыки в организации исследовательских и проектных работ, в управлении коллективом

Знать	<ul style="list-style-type: none"> - основные этапы выполнения научно- исследовательской работы; - компьютерные технологии поддержки принятия решений при проведении НИР; - основные этапы разработки САиУ; - виды обеспечения САиУ; - основные виды архитектуры САиУ;
Уметь	<ul style="list-style-type: none"> - использовать типовые средства компьютерных технологий для представления индивидуальных научных результатов;
Владеть	<ul style="list-style-type: none"> - навыками декомпозиции исследовательских и проектных работ; - методами структурирования задач НИР; - навыками представления индивидуальных научных результатов;
ПК-3 способностью применять современные методы разработки технического, информационного и алгоритмического обеспечения систем автоматизации и управления	
Знать	<ul style="list-style-type: none"> - основные методы разработки технического, информационного и алгоритмического обеспечения систем автоматизации и управления (САиУ); - основные типы современных компьютерных технологий разработки САиУ; - основы применения компьютерных технологий при создании SCADA-систем;
Уметь	<ul style="list-style-type: none"> - эффективно использовать современные методы разработки систем автоматизации и управления с применением компьютерных технологий; - эффективно использовать современные средства моделирования САиУ;
Владеть	<ul style="list-style-type: none"> - навыками работы с типовыми компьютерными средствами разработки и моделирования систем автоматизации и управления; - способами эффективного применения широкого круга современных методов разработки и моделирования САиУ, основанных на компьютерных технологиях.

4. Структура, объём и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 3 зачетных единиц 108 акад. часов, в том числе:

- контактная работа – 62,9 акад. часов;
- аудиторная – 60 акад. часов;
- внеаудиторная – 2,9 акад. Часов
- в форме практической подготовки – 24 акад. часа;
- самостоятельная работа – 9,4 акад. часов;
- подготовка к экзамену – 35,7 акад. часа

Форма аттестации - экзамен

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в акад. часах)			Самостоятельная работа студента	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код компетенции
		Лек.	лаб. зан.	практ. зан.				
1. Основы управления в технических системах с использованием компьютерных технологий								
1.1 Понятие компьютерных технологий. Классификация компьютерных технологий	3	2					ОК-2, ПК-3	
1.2 Основные методы решения задач управления в технических системах с использованием компьютерных технологий. Основные этапы разработки		1					ОК-2, ПК-3	
1.3 Виды обеспечения САиУ. Наука как объект		1					ОК-2, ПК-3	
1.4 Анализ и выбор архитектуры САиУ		2					ОК-2, ПК-3	
1.5 Компьютерные технологии при создании		2					ОК-2, ПК-3	
Итого по разделу		8						
2. Современные методы разработки систем автоматизации и управления с применением компьютерных технологий								
2.1 Основы программирования ПЛК ОВЕН	3	2					ОК-2, ПК-3	
2.2 Изучение языков программирования CoDeSys		2		6			ОК-2, ПК-3	

2.3	Реализация системы управления простыми объектами в CoDeSys			14	9	Выполнение творческого задания. Подготовка к семинару.	Отчет по творческому заданию. Выступление на семинаре	ОК-2, ПК-3
2.4	Реализация системы управления технологическим процессом в CoDeSys			10				ОК-2, ПК-3
2.5	Визуализация системы управления технологическим процессом в CoDeSys			10				ОК-2, ПК-3
Итого по разделу		4		40	9			
3.	Компьютерное моделирование при разработке и отладке программного обеспечения систем автоматизации и управления							
3.1	Основы моделирования систем управления в MATLAB Simulink	3			0,4	Самостоятельное изучение учебной и научной литературы. Подготовка к практическим занятиям	Собеседование	ПК-3
3.2	Моделирование САР температуры в MATLAB Simulink				4			ПК-3
3.3	Моделирование САР уровня воды в парогенераторе в MATLAB Simulink				4			ПК-3, ОК-2
Итого по разделу				8	0,4			
Итого за семестр		12		48	9,4		экзамен	
Итого по дисциплине		12		48	9,4		экзамен	ОК-2,ПК-3

5 Образовательные технологии

Для реализации предусмотренных видов учебной работы в качестве образовательных технологий в преподавании дисциплины «Компьютерные технологии управления в технических системах» используются:

Традиционные образовательные технологии – информационная лекция (вводная лекция, где дает первое представление о предмете и знакомство студентов с назначением и задачами курса); лекции – консультации, изложение нового материала сопровождается постановкой вопросов и дискуссией в поисках ответов на эти вопросы; практические работы.

Технологии проблемного обучения – практическое занятие в форме семинара и творческое домашнее задание, направленное на решение комплексной учебно-познавательной задачи, требующей от студента применения как научно-теоретических знаний, так и практических навыков.

Интерактивные технологии: семинар-дискуссия – коллективное обсуждение какого-либо спорного вопроса, проблемы, выявление мнений в группе. Изложение проблем и их совместное решение.

Информационно-коммуникационные образовательные технологии – в ходе проведения лекционных занятий предусматривается использование электронного демонстрационного материала (лекции-визуализации), использование Интернет ресурсов для промежуточных аттестаций и проверки остаточных знаний.

Практические занятия проводятся в форме практической подготовки в условиях выполнения обучающимися видов работ, связанных с будущей профессиональной деятельностью и направленных на формирование, закрепление, развитие практических навыков и компетенций по профилю образовательной программы.

6 Учебно-методическое обеспечение самостоятельной работы обучающихся

Представлено в приложении 1.

7 Оценочные средства для проведения промежуточной аттестации

Представлены в приложении 2.

8 Учебно-методическое и информационное обеспечение дисциплины (модуля)

а) Основная литература:

1. Борисевич, А. В. Теория автоматического управления: элементарное введение с применением MATLAB : монография / А. В. Борисевич. - Москва : Инфра-М, 2014. - 200 с. - ISBN 978-5-16-101828-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/470329> (дата обращения: 27.10.2020). – Режим доступа: по подписке.

2. Советов, Б. Я. Информационные технологии : учебник для вузов / Б. Я. Советов, В. В. Цехановский. — 7-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 327 с. — (Высшее образование). — ISBN 978-5-534-00048-1. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/449939> (дата обращения: 17.09.2020).

б) Дополнительная литература:

1. Коткин, Г. Л. Компьютерное моделирование физических процессов с использованием Matlab : учебное пособие для вузов / Г. Л. Коткин, Л. К. Попов, В. С. Черкасский. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 202 с. — (Высшее образование). — ISBN 978-5-534-10512-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/455883> (дата обращения: 27.10.2020).

2. Юсупов, Р. Х. Основы автоматизированных систем управления технологическими процессами: Учебное пособие / Юсупов Р.Х. - Москва : Инфра-Инженерия, 2018. - 132 с. ISBN 978-5-9729-0229-3. - Текст : электронный. - URL: <https://znanium.com/catalog/product/989081> (дата обращения: 17.09.2020). – Режим доступа: по подписке.

3. Хиврин, М. В. Аппаратное и программное обеспечение управления технологическими процессами. Разделы: Автоматизированные системы управления предприятием. Применение сетей во взрывоопасных зонах. Аппаратные и программные средства программируемых контроллеров : учебно-методическое пособие / М. В. Хиврин. — Москва : МИСИС, 2015. — 95 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/116790> (дата обращения: 27.10.2020). — Режим доступа: для авториз. пользователей.

4. Смирнов, Ю. А. Технические средства автоматизации и управления : учебное пособие / Ю. А. Смирнов. — 3-е изд., стер. — Санкт-Петербург : Лань, 2020. — 456 с. — ISBN 978-5-8114-5413-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/140779> (дата обращения: 27.10.2020). — Режим доступа: для авториз. пользователей.

5. Красавин, А. В. Компьютерный практикум в среде matlab : учебное пособие для вузов / А. В. Красавин, Я. В. Жумагулов. — 2-е изд. — Москва : Издательство Юрайт, 2020. — 277 с. — (Высшее образование). — ISBN 978-5-534-08509-9. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/455824> (дата обращения: 27.10.2020).

в) Методические указания:

1. Мамонова, Т. Е. Информационные технологии. Лабораторный практикум : учебное пособие для вузов / Т. Е. Мамонова. — Москва : Издательство Юрайт, 2020. — 176 с. — (Высшее образование). — ISBN 978-5-9916-7060-9. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/451399> (дата обращения: 27.10.2020).

2. Методические указания по выполнению практических заданий представлены в приложении 3.

г) Программное обеспечение и Интернет-ресурсы:

Программное обеспечение

Наименование ПО	№ договора	Срок действия лицензии
FAR Manager	свободно распространяемое ПО	бессрочно
7Zip	свободно распространяемое ПО	бессрочно
MS Windows XP Professional(для классов)	Д-1227-18 от 08.10.2018	11.10.2021
MS Office 2003 Professional	№ 135 от 17.09.2007	бессрочно
CoDeSys	свободно распространяемое ПО	бессрочно
MathWorks MathLab v.2014 Classroom License	К-89-14 от 08.12.2014	бессрочно

Профессиональные базы данных и информационные справочные системы

Название курса	Ссылка
Электронная база периодических изданий East View Information Services, ООО «ИВИС»	https://dlib.eastview.com/
Национальная информационно-аналитическая система – Российский индекс научного цитирования	URL: https://elibrary.ru/project_risc.asp
Поисковая система Академия Google (Google Scholar)	URL: https://scholar.google.ru/
Информационная система - Единое окно доступа к информационным ресурсам	URL: http://window.edu.ru/
Федеральное государственное бюджетное учреждение «Федеральный институт промышленной собственности»	URL: http://www1.fips.ru/
Российская Государственная библиотека. Каталоги	https://www.rsl.ru/ru/4readers/catalogues/
Электронные ресурсы библиотеки МГТУ им. Г.И. Носова	http://magtu.ru:8085/marcweb2/Default.asp
Университетская информационная система РОССИЯ	https://uisrussia.msu.ru
Международная наукометрическая реферативная и полнотекстовая база данных научных изданий «Web of science»	http://webofscience.com
Международная реферативная и полнотекстовая справочная база данных	http://scopus.com
Международная база полнотекстовых журналов Springer Journals	http://link.springer.com/
Международная коллекция научных протоколов по различным отраслям знаний	http://www.springerprotocols.com/
Международная база научных материалов в области физических наук и инжиниринга	http://materials.springer.com/

Международная база справочных изданий по всем отраслям знаний SpringerReference	http://www.springer.com/references
Международная реферативная база данных по чистой и прикладной математике zbMATH	http://zbmath.org/
Международная реферативная и полнотекстовая справочная база данных научных изданий «Springer Nature»	https://www.nature.com/siteindex
Архив научных журналов «Национальный электронно-информационный консорциум» (НИ НЭИКОН)	https://archive.neicon.ru/xmlui/

Интернет-ресурсы:

1. ОВЕН. Оборудование для автоматизации: [сайт]. – Москва, 1991. – URL: <https://owen.ru/> (дата обращения: 29.10.2020). – Текст. Изображение: электронные.
2. Среда программирования CODESYS. Техническая документация и ПО : [сайт]. – Москва, 1991. – URL: https://www.owen.ru/product/codesys_v2 (дата обращения: 29.10.2020). – Текст. Изображение: электронные.
3. Экспонента: MATLAB, Simulink, Центр инженерных технологий и моделирования: [сайт]. – Москва, 2001. – URL: <https://exponenta.ru/> (дата обращения: 29.10.2020). – Текст. Изображение: электронные.

9 Материально-техническое обеспечение дисциплины (модуля)

Материально-техническое обеспечение дисциплины включает:

1. Учебные аудитории для проведения занятий лекционного типа
Мультимедийные средства хранения, передачи и представления информации
2. Учебная аудитория для проведения практических занятий: компьютерный класс
Персональные компьютеры с пакетом MS Office, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета
3. Помещения для самостоятельной работы обучающихся
Персональные компьютеры с пакетом MS Office, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета
4. Учебные аудитории для групповых и индивидуальных консультаций, текущего контроля и промежуточных консультаций
Доска, мультимедийный проектор, экран
5. Помещение для хранения и профилактического обслуживания учебного оборудования
Стеллажи для хранения учебно-методической документации

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Компьютерные технологии управления в технических системах»

По дисциплине «Компьютерные технологии управления в технических системах» предусмотрена аудиторная и внеаудиторная самостоятельная работа обучающихся.

Внеаудиторная самостоятельная работа предполагает выполнение творческого домашнего задания (индивидуально или в составе группы). Аудиторная самостоятельная работа предполагает изучение теоретического материала по теме «Основы моделирования систем управления в MATLAB Simulink» и подготовку к выполнению практических работ по теме «Компьютерное моделирование при разработке и отладке программного обеспечения систем автоматизации и управления», а также выступление на семинаре-дискуссии, где проводится обсуждение выполненного творческого задания.

При выполнении творческого задания на основании заданной технологической схемы и описания технологического процесса обучающийся должен разработать:

- технологические требования к схеме управления;
- таблицу сигналов;
- прикладную программу для ПЛК;
- визуализацию работы системы управления;
- дать описание работы прикладной программы и визуализации.

В таблицу сигналов вносятся:

- порядковый номер переменной;
- имя переменной (не должно содержать пробелов и кириллицы);
- тип переменной (дискретный, аналоговый);
- класс переменной (локальная, глобальная);
- адрес (для внутренних переменных не заполняется).

Пример заполнения таблицы параметров:

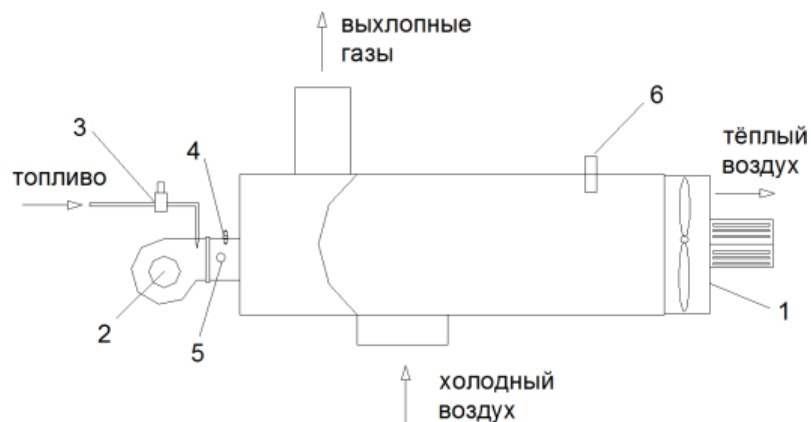
№ п/п	Наименование параметра	Имя	Тип	Класс	Адрес
1.	Температура воздуха	Temp_1	аналог	глобал.	%IВ0
2.	Кнопка «Пуск»	SB1	дискр.	локал.	-

В тексте программы необходимо учесть все технологические требования, предъявляемые к системе управления.

Работоспособность проекта обучающийся должен проверить в режиме эмуляции, а затем продемонстрировать на семинаре-дискуссии.

Примеры творческих заданий

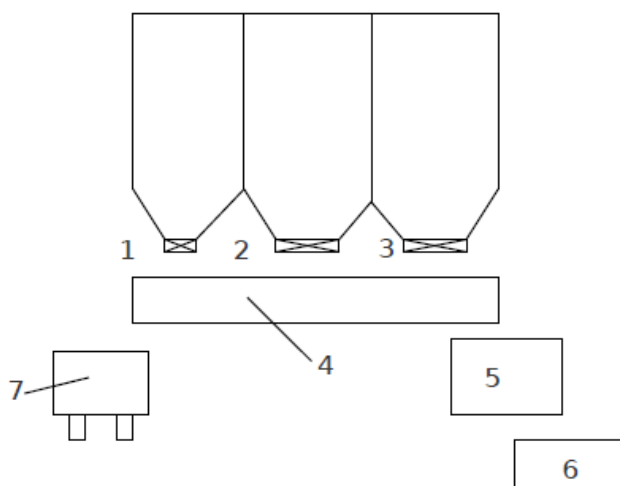
1. Система управления теплогенератором



При нажатии на кнопку ПУСК, звучит предупредительная сигнализация и запускается основной вентилятор теплого воздуха 1. После запуска основного вентилятора, включается топливный вентилятор 2 для продувки (10 с). Затем включается топливный соленоидный клапан 3 и топливная смесь закачивается в камеру сгорания (5 с). Срабатывает запальная свеча 4 (4 с). Реле пламени 5 контролирует наличие пламени.

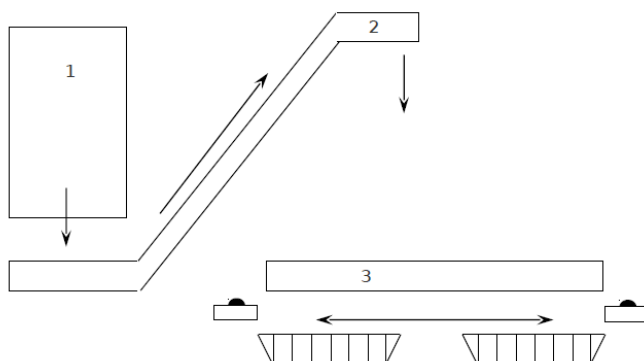
Если пламя не появилось в течение 5 с., процесс розжига выполняется еще раз (с продувки воздухом 15с.). При повторном незапуске агрегата включается продувка 1 мин. и аварийная сигнализация. При нормальном запуске агрегата, система должна контролировать температуру воздуха на выходе термопреобразователем 6 и изменять скорость вращения топливного вентилятора 2. При остановке агрегата, продувка должна осуществляться до тех пор, пока температура не упадет ниже T_{\min} .

2. Система управления бункерами и транспортером



Зерно поступает на транспортер 4 через одну из задвижек 1,2 или 3 или все вместе (выбор задвижки производится оператором) и далее либо в тележку 7 либо на дробилку 5 и далее в бункер 6. Схема должна отключаться при срабатывании датчика уровня в бункере 6 или при срабатывании датчика давления под тележкой.

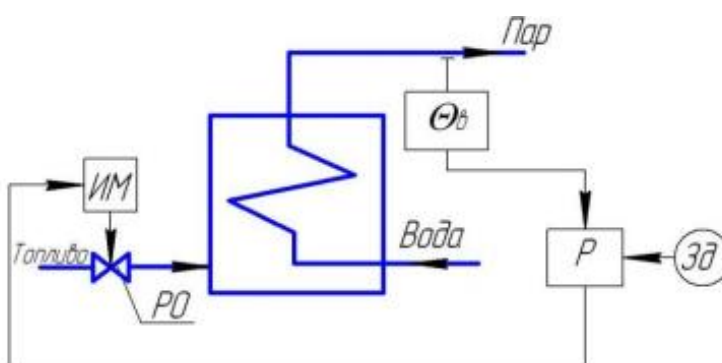
3. Система управления раздатчиком корма



Продукт на платформенный раздатчик корма 3 подается загрузочным транспортером 2 и шнековым дозатором корма из бункера 1. Платформенный раздатчик начинает движение после того, как на него падает первая порция корма. При этом транспортер 3 движется вправо. При наезде на конечный выключатель SQ1 корм сбрасывается в кормушки и транспортер останавливается. Обратное движение платформенного раздатчика начинается через одну-две секунды, при этом происходит заполнение второй половины платформенного раздатчика.

Через выдержку времени должно произойти отключение шнекового дозатора корма, а остатков корма на загрузочном транспортере 2 должно хватить для заполнения оставшейся части фронта кормления. При наезде на конечный выключатель SQ2 происходит сбрасывание корма во вторую половину кормушек и отключение всей схемы. Сброс корма в кормушки производится плужковыми сбрасывателями.

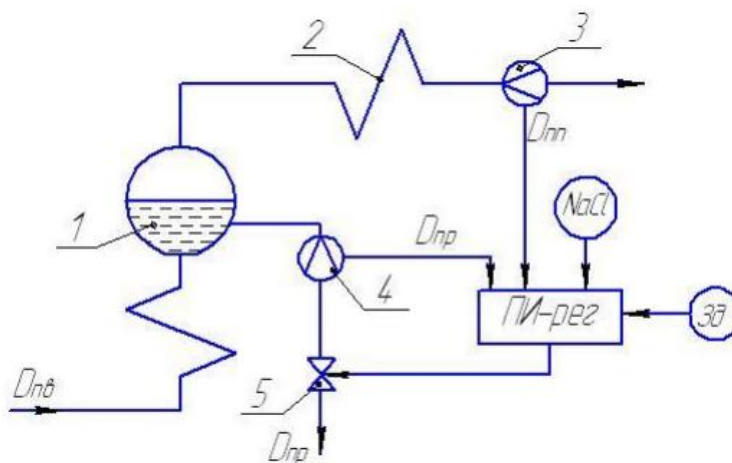
4. Система автоматического регулирования нагрузки водогрейного котла



Регулятор нагрузки котла получает импульс по температуре воды за котлом и воздействует на изменение подачи топлива к котлу. Закон управления можно выбрать любой, например, ПИ-закон управления.

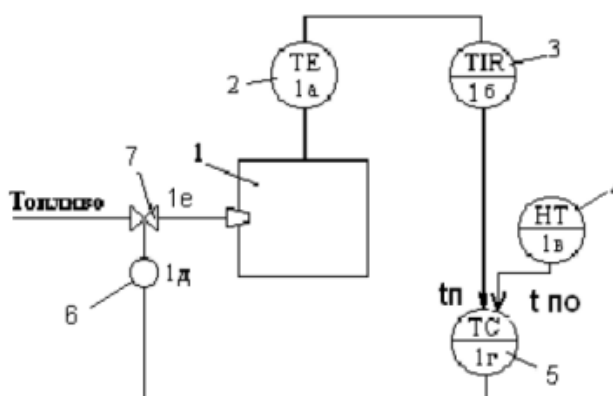
5. Система автоматического регулирования непрерывной продувки барабанного парового котла

Непрерывная продувка котла служит для удаления избытка соли NaCl и оксидов кремния SiO₂, скапливающихся в котловой воде в процессе парообразования. Регулирование непрерывной продувки осуществляют воздействием регулятора продувки на регулировочный клапан на линии продувки. На вход ПИ-регулятора поступают сигналы по расходу пара $D_{пп}$ и расходу продувочной воды $D_{пр}$, а также корректирующий сигнал по солесодержанию солей NaCl.



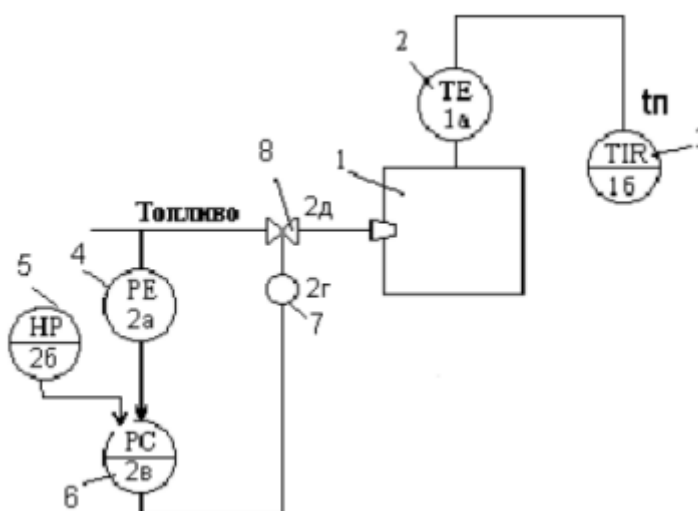
1 – барабан к/а, 2 – пароперегреватель, 3 – датчик расхода свежего пара $D_{пп}$, 4 – датчик расхода продувочной воды $D_{пр}$, 5 – регулирующий клапан продувки, ПИ-рег – регулятор продувки, 3д – задатчик ручного управления

6. Замкнутая САУ температуры в печи



Чувствительным элементом – датчиком температуры служит термопара 2 (поз. обозн. 1а). Информация о значении температуры в печи поступает на показывающий и регистрирующий прибор 3 (поз. обозн. 1б), а с него в регулятор 5 (поз. обозн. 1г). В регулятор с задатчика 4 (поз. обозн. 1в) поступает сигнал о заданном значении температуры $t_{но}$, в состав которого входит сравнивающий элемент. Сравнивающий элемент вырабатывает отклонение $\varepsilon = t_{но} - t_n$, и в соответствии с алгоритмом управления, регулятор формирует управляющее воздействие. Это воздействие в виде управляющего сигнала передаётся на исполнительный механизм 6 (поз. обозн. 1д), обеспечивающий перемещение регулирующего органа 7 (поз. обозн. 1е). В качестве регулирующего органа используется поворотная заслонка в трубопроводе. Если температура в печи меньше заданной, то расход топлива увеличивается, а если больше – то уменьшается. Предусмотреть ручной и автоматический режим работы системы управления.

7. Разомкнутая САР температуры в печи с регулированием по возмущению

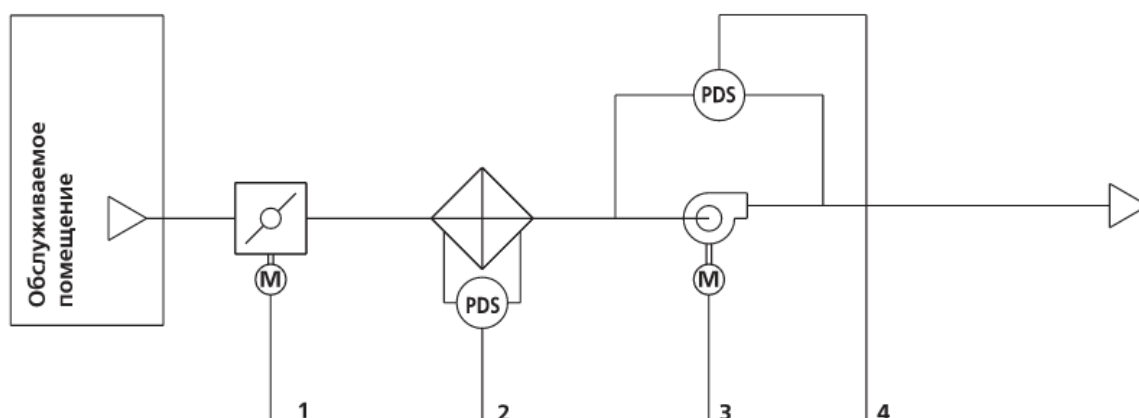


Регулируемой величиной является температура t_n в печи 1. Основным возмущением является изменение давления газа в газопроводе, которое вызывает изменение расхода топлива и изменение температуры в печи, т.е. изменение регулируемой величины. Для компенсации влияния возмущения на значение выходной величины применяют регулятор 6 (поз. обозн. 2в), называемый компенсатором возмущений. Регулятор получает информацию о значении давления газа от датчика давления 4 (поз. обозн. 2а) и заданном значении давления от ручного задатчика 5 (поз. обозн. 2б). Затем по заранее заданной программе с помощью исполнительного механизма 7 (поз. обозн. 2г) регулятор изменяет положение регулирующего органа 8 (поз. обозн. 2д). Давление перед горелкой при правильно выбранной структуре и законе действия компенсатора не будет зависеть от давления в газопроводе и, следовательно, не будет сказываться на расходе топлива и

значении температуры в печи. В этом заключается принцип компенсации возмущений. В рассмотренном примере регулируемая величина – температура в печи измеряется термопарой 2 (поз. обозн. 1а) и регистрируется прибором 3 (поз. обозн. 1а). Но эта текущая информация не используется системой регулирования, т.е. отсутствует обратная связь по результатам работы системы.

Контур компенсации возмущения разомкнут, т.е. выходная величина контура не оказывает влияния на входную величину – изменение давления в газопроводе.

8. Система управления вентиляцией (вытяжкой)



Система вентиляции (вытяжки) содержит следующие элементы:

- 1 — привод воздушной заслонки;
- 2 — датчик-реле перепада давления на фильтре (PDS);
- 3 — вентилятор;
- 4 — датчик-реле перепада давления на вентиляторе (PDS).

Система имеет два режима запуска:

- местный
- (с электрического щита);
- дистанционный
- (по команде оператора из диспетчерской с выносного пульта управления).

Режим выбирается переключателем «Вкл / Выкл / ДУ» на лицевой панели щита.

Режимы работы вытяжной вентиляции:

- автономный режим, когда включение системы происходит непосредственно со щита;
- заблокированный режим, когда включение системы происходит от приточной вентиляции.

При срабатывании внешнего датчика сигнализации «Пожар» система выключается.

Система предусматривает управление и контроль следующих параметров:

- 1) контроль засорения фильтра по датчику-реле перепада давления воздуха;
- 2) контроль работоспособности вентилятора по датчику-реле перепада давления воздуха;
- 3) контроль работоспособности вентилятора по токам короткого замыкания;
- 4) управление воздушной заслонкой электроприводом.

Описание работы системы

Запуск производится переключателем «Пуск» в положение «Вкл», загорается индикатор «Пуск».

1) Если система настроена на автономную работу (на сухих контактах установлена перемычка), происходит запуск двигателя вентилятора 3, привод 1открывает воздушную заслонку, при открытии заслонки загорается индикатор «Заслонка», работает датчик-реле 2 перепада давления на фильтре. Через определенный интервал времени включается датчик-реле 4 перепада давления на вентиляторе. При выходе вентилятора на рабочий режим загорается индикатор «Вентилятор».

2) Если система заблокирована с включением приточной вентиляции, то она переходит в режим ожидания. При запуске приточной вентиляции происходит запуск и вытяжной вентиляции. Дальнейшая работа системы аналогична автономному режиму работы.

Воздух из обслуживаемого помещения, проходя через открытую воздушную заслонку, попадает на воздушный фильтр. Если перепад давления на фильтре слишком велик, что определяется по датчику-реле 2, то на щите загорается индикатор «Фильтр». Отключение системы при этом не предусмотрено.

Датчик-реле 4 контролирует перепад давления воздуха на вентиляторе 3.

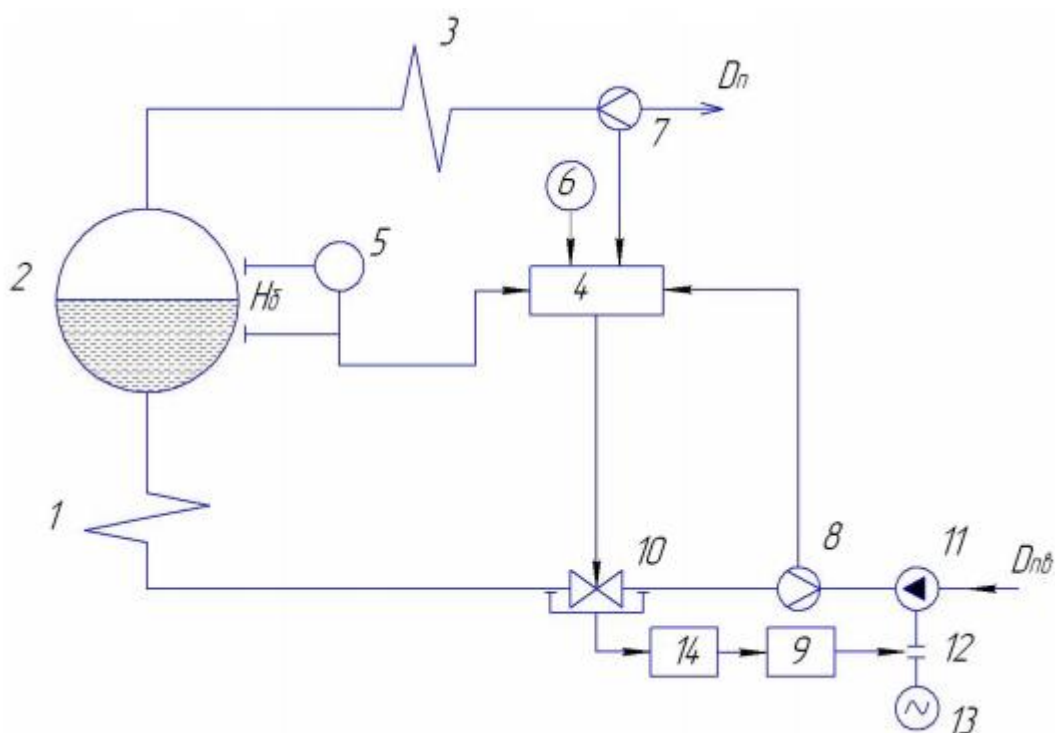
Если при запуске системы через определенный интервал времени заданный перепад давления не появляется, система останавливается. То же происходит, если указанный перепад давления исчезает во время работы системы. При этом загорается индикатор «Авария», индикатор «Вентилятор» гаснет.

9. Система автоматического регулирования питания барабана парового котла

В САР питания котла водой должен быть реализован принцип комбинированного регулирования по возмущению – при изменении расхода пара или питательной воды и отклонению – при изменении уровня воды в барабане котла. Регулятор питания должен обеспечить постоянство среднего уровня воды независимо от нагрузки котла и возмущающих воздействий.

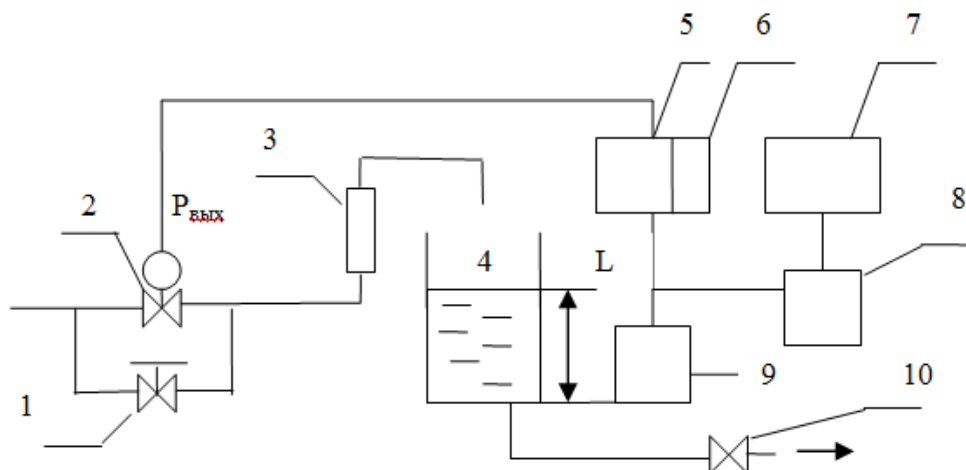
В САР питания используют для этих целей трехимпульсный регулятор питания. Сигналы по возмущению: расход свежего пара D_n , расход питательной воды $D_{пв}$. Сигнал по отклонению: уровень в барабане котельного агрегата $H_б$. Сигнал по расходу питательной воды используется как выключающий для снятия в статике сигнала по расходу пара. Регулятор питания перемещает регулировочный орган на линии питательной воды при появлении сигнала небаланса между расходами питательной воды и перегретого пара. Помимо этого он воздействует на положение клапана при отклонении уровня воды в барабане котельного агрегата от заданного значения. Использование сигналов D_n и $D_{пв}$ обеспечивают быстрое действие САР питания, сигнал $H_б$ – заданную точность поддержания уровня в барабане. В схеме измерительного блока регулятора питания датчики D_n , $D_{пв}$ и $H_б$ включены таким образом, что при понижении уровня воды в барабане котлоагрегата, увеличении расхода пара, уменьшении расхода питательной воды, они действуют в одном направлении – в сторону открытия питательного клапана, а при повышении уровня, уменьшении расхода пара и увеличении расхода питательной воды в сторону закрытия питательного клапана.

В качестве регулировочных органов питания используются шиберные клапаны и клапаны золотникового типа.



- 1-экономайзер, 2-барабан котла, 3-пароперегреватель, 4-регулятор питания, 5-датчик уровня, 6-задатчик, 7-датчик расхода пара, 8-датчик расхода питательной воды, 9-регулятор производительности, 10-питательный клапан, 11-питательный насос, 12-гидромуфта, 13- электродвигатель, 14 – дифференциальный манометр

10. Система автоматического регулирования уровня жидкости в резервуаре



В системе управления должно быть реализовано одноконтурное регулирование уровня жидкости в резервуаре.

Вода поступает в резервуар 4 (объект регулирования) по длинному трубопроводу, на котором установлены параллельно вентиль 1 и пневматический регулирующий клапан 2. Расход воды контролируют по показаниям ротаметра 3. Уровень воды в резервуаре (регулируемую величину) измеряют гидростатическим дифманометром-уровнемером 9. Пневматический сигнал с его выхода поступает по вторичный прибор со станцией управления 5 и работающий с ним в комплекте автоматический регулятор 6, а также по манометр 8 с электрическим выходом, подаваемым на регистрирующий прибор 7. Выход с регулятора направляется на исполнительное устройство - клапан 2. Вентиль 1 предназначен для подачи на объект и в систему регулирования возмущающих воздействий, вентиль 10 — для установки номинального расхода жидкости через объект. Заданное значение уровня жидкости в резервуаре устанавливают задатчиком станции управления прибора 5.

Приложение 2

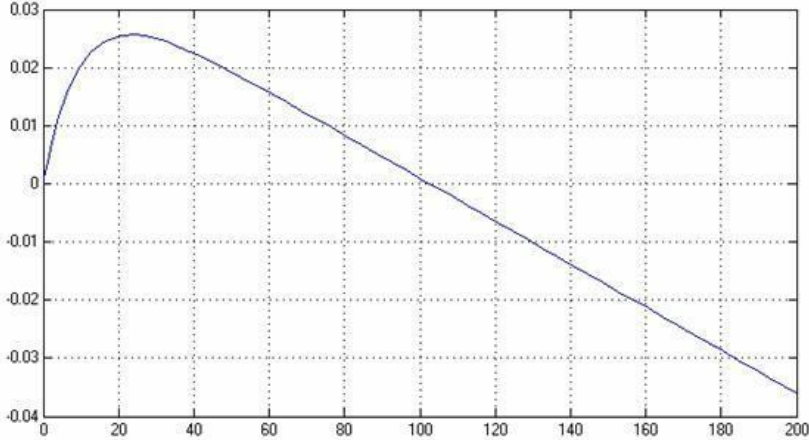
**Оценочные средства для проведения промежуточной аттестации
по дисциплине «Компьютерные технологии управления в технических системах»**

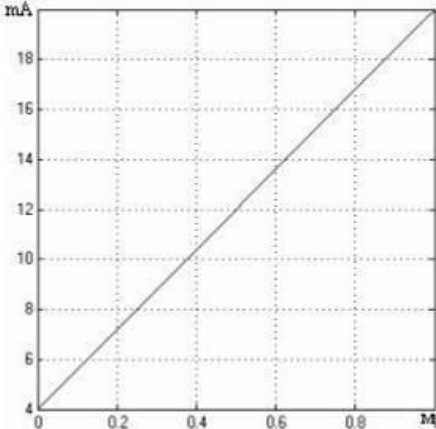
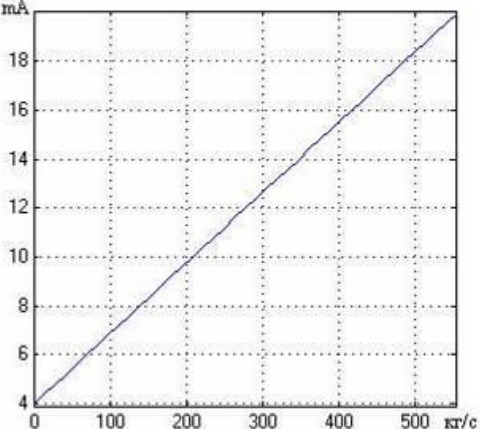
а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации:

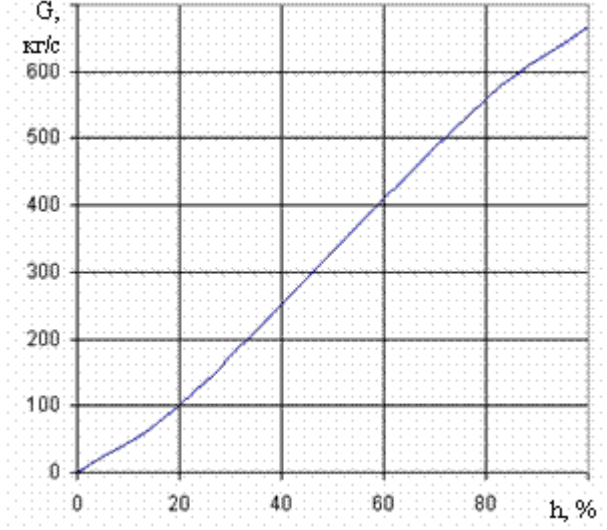
Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
способностью использовать на практике умения и навыки в организации исследовательских и проектных работ, в управлении коллективом (ОК-2)		
Знать	<ul style="list-style-type: none"> – основные этапы выполнения научно-исследовательской работы; – компьютерные технологии поддержки принятия решений при проведении НИР; – основные этапы разработки САиУ; – виды обеспечения САиУ; – основные виды архитектуры САиУ; 	<p>Теоретические вопросы для проведения экзамена:</p> <ol style="list-style-type: none"> 1. Определение понятия «компьютерная технология» 2. Наука как объект компьютеризации. Схема рациональной организации НИР 3. Основные этапы разработки САиУ 4. Виды обеспечения САиУ 5. Архитектуры САиУ. Централизованная архитектура 6. Архитектуры САиУ. Децентрализованная архитектура 7. Архитектуры САиУ. Многоуровневая архитектура
Уметь	<ul style="list-style-type: none"> – использовать типовые средства компьютерных технологий для представления индивидуальных научных результатов; 	<p>Для объекта управления с заданными параметрами:</p> <ol style="list-style-type: none"> 1. Произвести анализ САУ, определив регулируемый параметр (параметры) возмущающие и управляющие воздействия 2. Составить структурную схему САУ 3. Составить математическую модель САУ 4. Реализовать модель системы управления в среде моделирования (например, в пакете MATLAB)
Владеть	<ul style="list-style-type: none"> – навыками декомпозиции исследовательских и проектных работ; – методами структурирования задач НИР; 	<ol style="list-style-type: none"> 1. Реализовать модель САУ температурой парового подогревателя в среде моделирования (например, в пакете MATLAB) Объект управления включает в себя проточную ёмкость, в которую установлен паровой подогреватель воды. Вода в ёмкость подаётся с температурой 20°С и массовым расходом 40кг/мин. В ёмкости поддерживается постоянный уровень;

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
	<p>– <i>навыками индивидуальных результатов;</i> <i>представления научных</i></p>	<p>масса воды, находящейся в ёмкости – 100 кг. Температура воды, выходящей из ёмкости – 80 °С.</p>  <p>2. Реализовать модель САР уровня воды в парогенераторе в среде моделирования (например, в пакете MATLAB) Регулирование питания в парогенераторе (ПГ) сводится к поддержанию материального баланса между отводом пара, продувкой и подачей питательной воды. Параметром, характеризующим материальный баланс, является уровень воды в ПГ. К стабилизации уровня предъявляются довольно жесткие требования. Для ПГ с ВВЭР-1000 номинальный уровень $L_{ном}$ составляет 2450 мм от внутренней образующей корпуса. Точность поддержания уровня в статических режимах составляет ± 50 мм от $L_{ном}$, в динамике ± 150 мм от $L_{ном}$ (с учетом нечувствительности регулятора). Повышение уровня воды от $L_{ном}$ не допускается из-за затопления и нарушения работы сепарационных устройств (заброс воды в турбину), а снижение уровня - из-за оголения поверхности нагрева.</p>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
		<p data-bbox="996 319 2094 422">Экспериментальным путем были получены разгонные кривые парогенератора по уровню при подаче на него возмущений расходом питательной воды и расходом пара.</p>  <p data-bbox="952 917 2083 997">Кривая разгона ПГ по уровню при подаче скачкообразного возмущения расходом питательной воды $DG = 10$ кг/с</p>

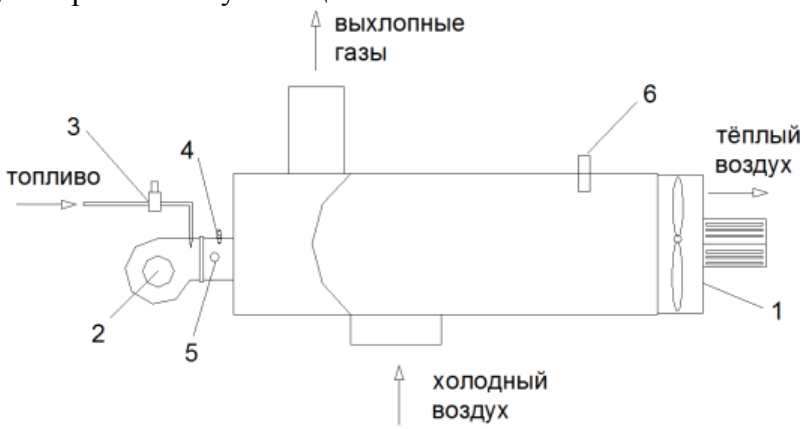
Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
		 <p data-bbox="958 799 2085 874">Кривая разгона ПГ по уровню при подаче скачкообразного возмущения расходом пара $DD = 10 \text{ кг/с}$</p> <p data-bbox="996 911 2085 1011">Уровень в ПГ измеряется датчиком типа Сапфир-22ДД по малому уровнемеру с пределом измерения 0-1000 мм, используя двухкамерный уравнильный сосуд. Статическая характеристика уровнемера приведена на рисунке.</p>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
		<div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;">  <p>а)</p> </div> <div style="text-align: center;">  <p>б)</p> </div> </div> <p>Статические характеристики ИП типа «Сапфир -22ДД» для измерения уровня (а) и расходов питательной воды и пара (б)</p> <p>Расходная характеристика регулирующего питательного клапана приведена на рисунке.</p>

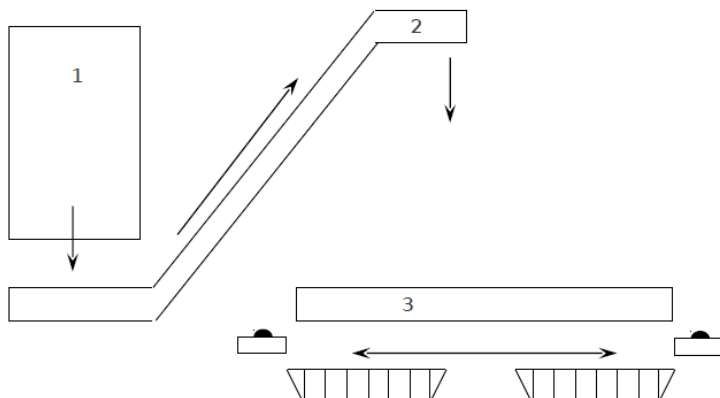
Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
		 <p style="text-align: center;">Расходная характеристика регулирующего питательного клапана</p>
<p>способностью применять современные методы разработки технического, информационного и алгоритмического обеспечения систем автоматизации и управления (ПК-3)</p>		
<p>Знать</p>	<ul style="list-style-type: none"> – основные методы разработки технического, информационного и алгоритмического обеспечения систем автоматизации и управления (САиУ); – основные типы современных компьютерных технологий разработки САиУ; 	<p>Теоретические вопросы для проведения экзамена:</p> <ol style="list-style-type: none"> 1. Классификация компьютерных технологий. Технологии систем реального времени 2. Классификация компьютерных технологий. Сетевые технологии 3. Классификация компьютерных технологий. Компьютерные технологии взаимодействия с человеком-оператором 4. Классификация компьютерных технологий. Технологии программно-технических комплексов

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
	<p>– <i>основы применения компьютерных технологий при создании SCADA-систем;</i></p>	<ol style="list-style-type: none"> 5. Классификация компьютерных технологий. Технологии компьютерного (технического) зрения 6. Технологии интеллектуальных сенсоров. Пассивные и активные сенсоры. 7. Технологии интеллектуальных сенсоров. Сенсорно-компьютерные системы. 8. Технологии интеллектуальных сенсоров. Понятие интеллектуального сенсора. 9. Технологии интеллектуальных сенсоров. Классификация интеллектуальных сенсоров. 10. АСУ ТП и диспетчерское управление 11. Компоненты систем контроля и управления и их назначение 12. Разработка прикладного программного обеспечения АСУ: выбор пути и инструментария 13. Термины «SCADA-система» и «SCADA-пакет» 14. Организация и основные функции современных SCADA-пакетов 15. Контроллер ОВЕН ПЛК110: аппаратные особенности, подключение к ПК. Понятие Target-файла 16. Среда программирования CoDeSys. Основные принципы стандарта МЭК 61131-3. 17. Структура проекта в среде программирования CoDeSys. 18. Языки программирования программного комплекса CoDeSys. Язык IL (Instruction list) 19. Языки программирования программного комплекса CoDeSys. Язык релейно-контактных схем (LD) 20. Языки программирования программного комплекса CoDeSys. Язык функциональных блок-диаграмм (FBD) 21. Языки программирования программного комплекса CoDeSys. Язык последовательных функциональных схем (SFC) 22. Языки программирования программного комплекса CoDeSys. Непрерывные функциональные схемы (CFC) 23. Визуализация в CoDeSys

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
Уметь	<ul style="list-style-type: none"> – <i>эффективно использовать современные методы разработки систем автоматизации и управления с применением компьютерных технологий;</i> – <i>эффективно использовать современные средства моделирования САиУ</i> 	<ol style="list-style-type: none"> 1. Реализовать в среде CoDeSys на языках IL, FBD, LD логическое выражение заданного вида, разработать визуализацию: <ul style="list-style-type: none"> ✓ $Q0 = I0 \& \bar{I1} \vee I2 \& I3$ ✓ $Q1 = I4 \& (I4 \vee I6) \& I7$ ✓ $Q2 = I0 \vee I1 \& I2 \& \bar{I3}$ ✓ $Q3 = I5 \vee I4 \vee I6 \& I7$ ✓ $Q4 = I0 \& I1 \vee I2 \vee I3$ ✓ $Q5 = I4 \& I5 \& (I6 \vee I7)$ ✓ $Q6 = I0 \& \bar{I1} \& I2 \& I3$ ✓ $Q7 = I4 \vee I1 \vee I2 \vee I3$ ✓ $Q0 = I4 \& I5 \vee I2 \& I3$ ✓ $Q1 = I0 \& (\bar{I1} \vee I2) \& I3$ ✓ $Q2 = I5 \& I6 \vee I2 \& \bar{I3}$ ✓ $Q3 = I0 \& I1 \& I2 \& \bar{I3}$ ✓ $Q4 = I0 \& I1 \& I4 \vee I5$ ✓ $Q5 = I0 \& \bar{I1} \vee I2 \& I5$ ✓ $Q6 = I0 \& I1 \vee I2 \vee \bar{I3}$ ✓ $Q0 = I4 \& \bar{I5} \vee I2 \& I3$ ✓ $Q7 = I2 \& I1 \vee \bar{I3} \vee I4$ 2. Реализовать в среде CoDeSys систему управления освещением в комнате, предусмотреть визуализацию. 3. Реализовать в среде CoDeSys систему управления котлом, предусмотреть визуализацию. 4. Реализовать в среде CoDeSys систему управления насосом с задержкой по времени, предусмотреть визуализацию. 5. Реализовать в среде CoDeSys систему пожарной сигнализации в здании, предусмотреть визуализацию.

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
Владеть	<ul style="list-style-type: none"> – <i>навыками работы с типовыми компьютерными средствами разработки и моделирования систем автоматизации и управления;</i> – <i>способами эффективного применения широкого круга современных методов разработки и моделирования САиУ, основанных на компьютерных технологиях.</i> 	<p>1. Разработать программу системы управления теплогенератором, предусмотреть ее визуализацию</p>  <p>При нажатии на кнопку ПУСК, звучит предупредительная сигнализация и запускается основной вентилятор теплого воздуха 1. После запуска основного вентилятора, включается топливный вентилятор 2 для продувки (10 с). Затем включается топливный соленоидный клапан 3 и топливная смесь закачивается в камеру сгорания (5 с). Срабатывает запальная свеча 4 (4 с). Реле пламени 5 контролирует наличие пламени.</p> <p>Если пламя не появилось в течение 5 с., процесс розжига выполняется еще раз (с продувки воздухом 15с.). При повторном незапуске агрегата включается продувка 1мин. и аварийная сигнализация. При нормальном запуске агрегата, система должна контролировать температуру воздуха на выходе термопреобразователем 6 и изменять скорость вращения топливного вентилятора 2. При остановке агрегата, продувка должна осуществляться до тех пор, пока</p>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
		<p>температура не упадет ниже T_{\min}.</p> <p>2. Разработать программу системы управления бункерами и транспортером, предусмотреть ее визуализацию</p>  <p>Зерно поступает на транспортер 4 через одну из задвижек 1,2 или 3 или все вместе (выбор задвижки производится оператором) и далее либо в тележку 7 либо на дробилку 5 и далее в бункер 6. Схема должна отключаться при срабатывании датчика уровня в бункере 6 или при срабатывании датчика давления под тележкой.</p> <p>3. Разработать программу системы управления раздатчиком корма, предусмотреть ее визуализацию</p> <p>Продукт на платформенный раздатчик корма 3 подается загрузочным транспортером 2 и шнековым дозатором корма из бункера 1. Платформенный раздатчик начинает движение после того, как на него падает первая порция корма.</p>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
		<p>При этом транспортер 3 движется вправо. При наезде на конечный выключатель SQ1 корм сбрасывается в кормушки и транспортер останавливается. Обратное движение платформенного раздатчика начинается через одну-две секунды, при этом происходит заполнение второй половины платформенного раздатчика.</p>  <p>Через выдержку времени должно произойти отключение шнекового дозатора корма, а остатков корма на загрузочном транспортере 2 должно хватить для заполнения оставшейся части фронта кормления. При наезде на конечный выключатель SQ2 происходит сбрасывание корма во вторую половину кормушек и отключение всей схемы. Сброс корма в кормушки производится плужковыми сбрасывателями.</p>

б) Порядок проведения промежуточной аттестации, показатели и критерии оценивания:

Промежуточная аттестация по дисциплине «Компьютерные технологии управления в технических системах» включает теоретические вопросы, позволяющие оценить уровень усвоения обучающимися знаний, и практические задания, выявляющие степень сформированности умений и владений, проводится в форме экзамена.

Экзамен по данной дисциплине проводится в устной форме по экзаменационным билетам, каждый из которых включает 2 теоретических вопроса и одно практическое задание.

Показатели и критерии оценивания экзамена:

– на оценку **«отлично»** (5 баллов) – обучающийся демонстрирует высокий уровень сформированности компетенций, всестороннее, систематическое и глубокое знание учебного материала, свободно выполняет практические задания, свободно оперирует знаниями, умениями, применяет их в ситуациях повышенной сложности.

– на оценку **«хорошо»** (4 балла) – обучающийся демонстрирует средний уровень сформированности компетенций: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.

– на оценку **«удовлетворительно»** (3 балла) – обучающийся демонстрирует пороговый уровень сформированности компетенций: в ходе контрольных мероприятий допускаются ошибки, проявляется отсутствие отдельных знаний, умений, навыков, обучающийся испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

– на оценку **«неудовлетворительно»** (2 балла) – обучающийся демонстрирует знания не более 20% теоретического материала, допускает существенные ошибки, не может показать интеллектуальные навыки решения простых задач.

– на оценку **«неудовлетворительно»** (1 балл) – обучающийся не может показать знания на уровне воспроизведения и объяснения информации, не может показать интеллектуальные навыки решения простых задач.

Методические указания к выполнению практических работ по дисциплине
«Компьютерные технологии управления в технических системах»

Практическая работа №1

Изучение языков программирования CoDeSys

Цель работы: приобретение и закрепление навыков программирования в среде программного комплекса CoDeSys на трех языках программирования (Список Инструкций (IL), Диаграммы Функциональных блоков (FBD) и Релейно-контактные схемы (LD)), а также обучение основам визуализации.

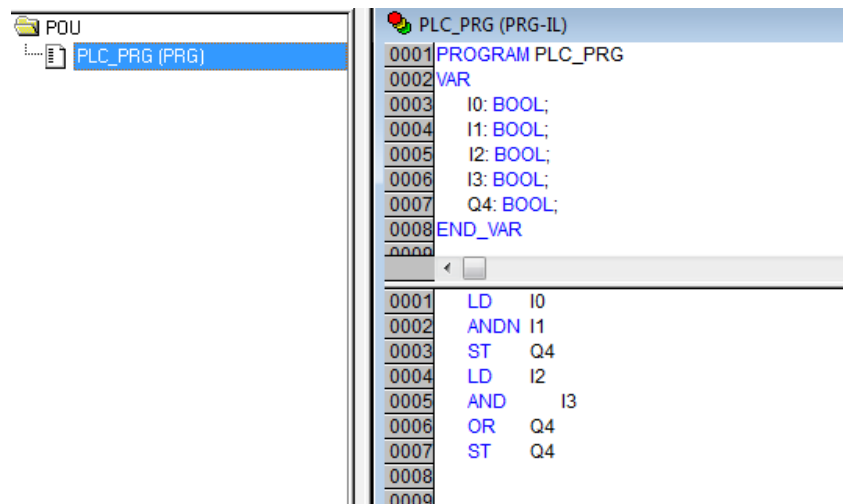
Порядок выполнения работы

1. Исследование языка IL

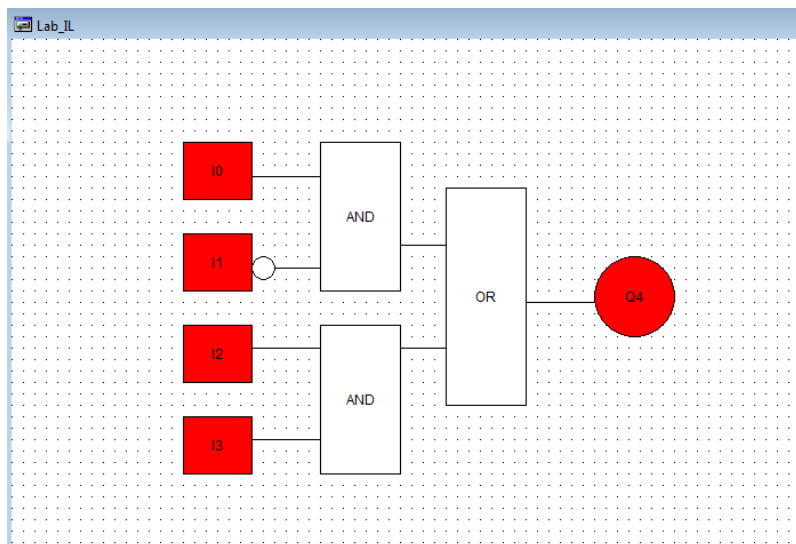
1. Запустить CoDeSys V2.3.
2. Создать новый проект. Выполнить команду: Файл, Создать.
3. Выбрать целевую платформу (Target Settings) 3S CoDeSys SP PLCWinNT V2.4.
4. Определить тип первого программного компонента (New POU) – PLC_PRG. Выбрать язык программирования – IL и тип компонента – программа. В однозадачных проектах система исполнения циклически вызывает программу PLC_PRG.
5. Объявить переменные I0, I1, I2, I3, Q4. В диалоге определения переменных определить тип переменных – BOOL и класс переменных – VAR GLOBAL.

В редакторе набрать текст программы согласно своему варианту. Например, для логических условий $Q4 = (I0 \& \bar{I1}) \vee (I2 \& I3)$, получаем следующий текст:

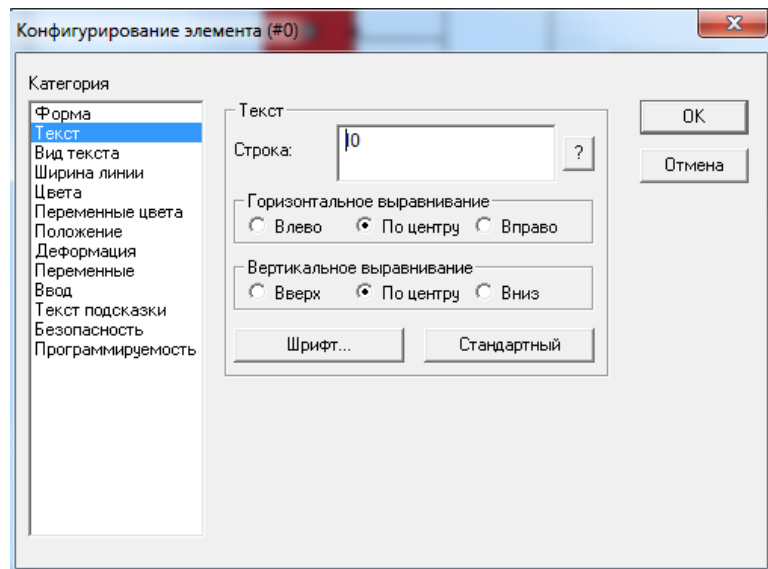
LD	I0
ANDN	I1
ST	Q4
LD	I2
AND	I3
OR	Q4
ST	Q4



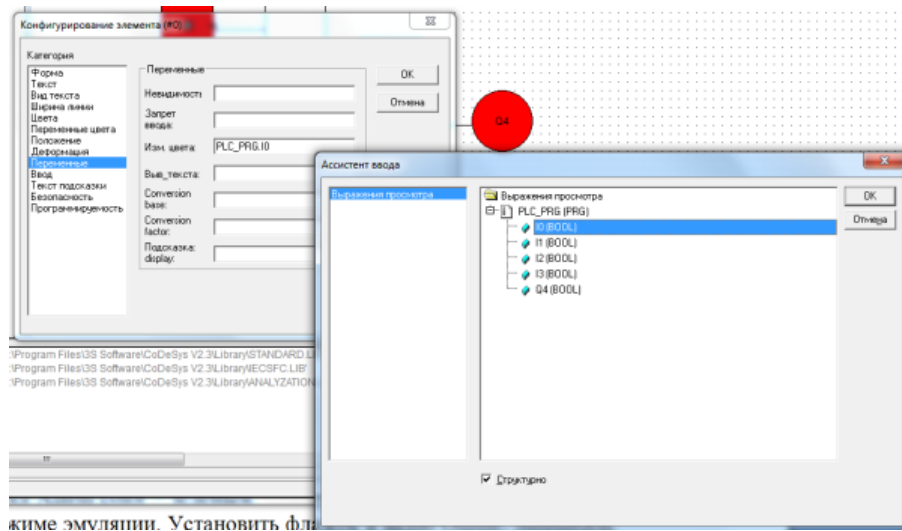
6. Откомпилировать проект. Выполнить команду: Проект, Компилировать или нажать «F11».
7. Исправить ошибки, если таковые имеются.
8. Создать визуализацию проекта. В левой части окна CoDeSys внизу выбрать страницу визуализации (Visualization). На Visualization щелкнуть правой клавишей мыши. В контекстном меню ввода задать команду *Добавить объект...* Присвоить новому объекту имя Lab_IL.
9. Нарисовать элементы визуализации в виде прямоугольников для переменных I0, I1, I2, I3, логических элементов AND, OR. Нарисовать элемент визуализации в виде окружности для переменной Q4.



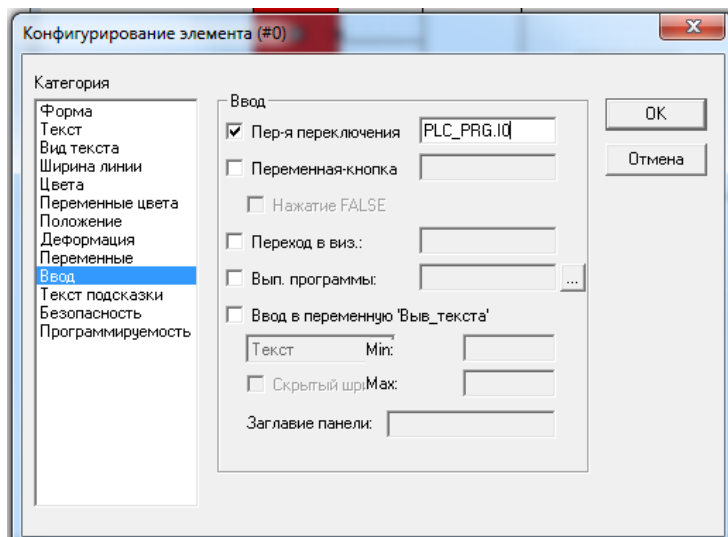
10. Настроить элементы визуализации – переменные I0, I1, I2, I3, Q4. Для них задать следующие настройки:
 - Текст, строка – ввести текст I0 (I1, I2, I3, Q4);



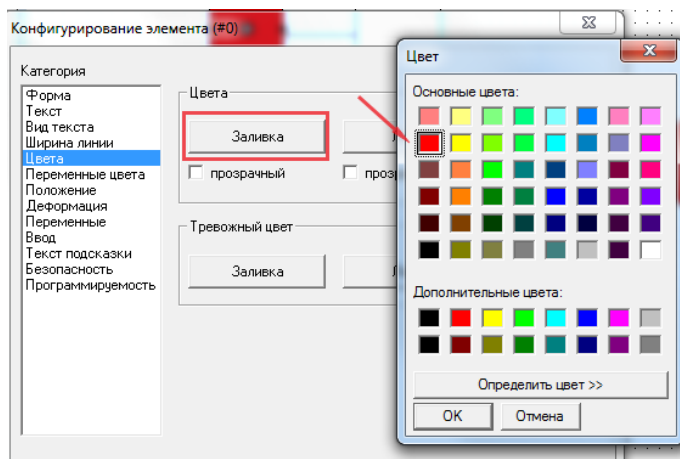
- Переменные, Изм.цвета – выбрать переменную из списка, нажав кнопку F2, например PLC_PRG.I0 (PLC_PRG.I1, PLC_PRG.I2, PLC_PRG.I3, PLC_PRG.Q4);



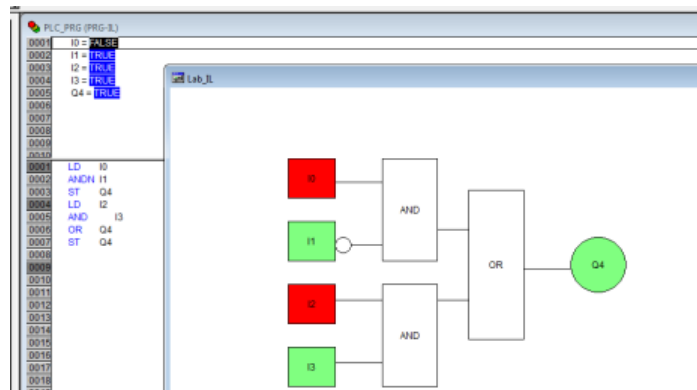
- Ввод, флажок Переменная переключения включить, выбрать переменную из списка, нажав кнопку F2, например PLC_PRG.I0 (PLC_PRG.I1, PLC_PRG.I2, PLC_PRG.I3);



- Цвета, верхняя панель Цвета – заливка красным (выключенное состояние переменной), нижняя панель Тревожный цвет – заливка зеленым (включенное состояние переменной).

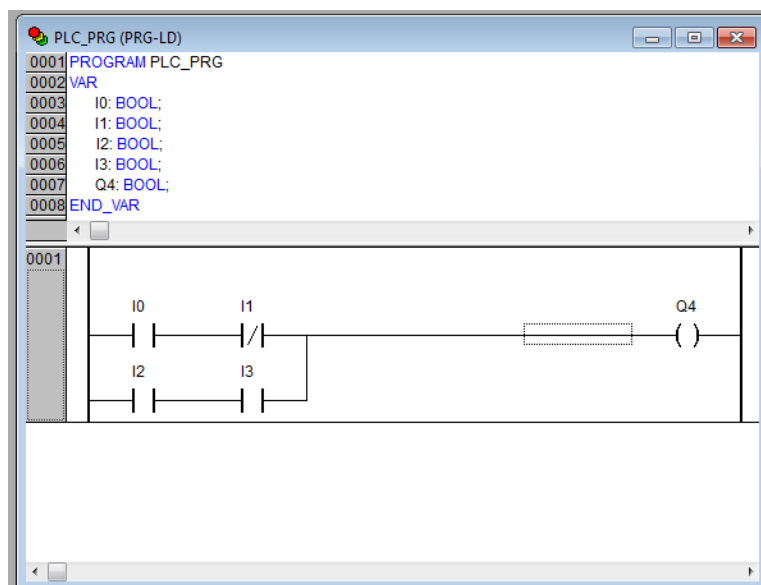


11. Перейти в режиме эмуляции. Установить флажок в меню Онлайн, Режим эмуляции.
12. Запустить проект. Выполнить команду Онлайн, Подключение. Выполнить команду Онлайн, Пуск (или кнопкой F5).
13. Изменя значения входных переменных, нажимая на элементы визуализации, наблюдать за изменением выходной переменной. Зафиксировать результаты вычислительного эксперимента в виде таблицы, где отобразить состояние входов и состояние выхода.
14. Остановить проект. Выполнить команду Онлайн, Стоп.



2. Исследование языка LD

1. Создать новый проект. Выполнить команду: Файл, Создать.
2. Выбрать целевую платформу (Target Settings) 3S CoDeSys SP PLCWinNT V2.4.
3. Определить тип первого программного компонента (New POU) – PLC_PRG. Выбрать язык программирования – LD и тип компонента – программа. В однозадачных проектах система исполнения циклически вызывает программу PLC_PRG.
4. Объявить переменные I0, I1, I2, I3, Q4. В диалоге определения переменных определить тип переменных – BOOL и класс переменных – VAR GLOBAL.
5. В графическом редакторе нарисовать схему из параллельных и последовательных контактов I0, I1, I2, I3 и обмотки Q4 согласно своему варианту.

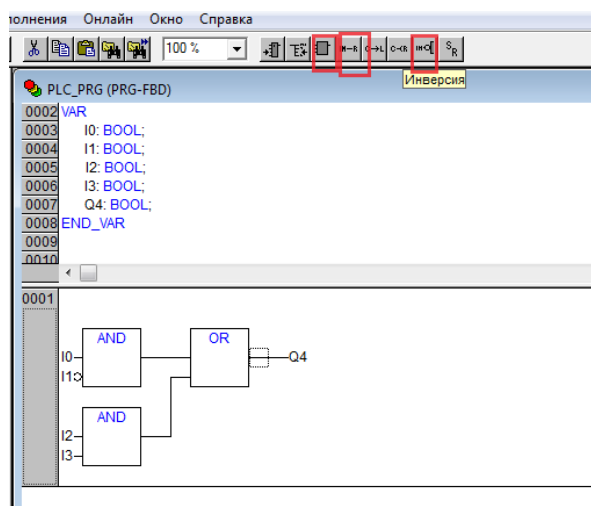


6. Откомпилировать проект. Выполнить команду: Проект, Компилировать или нажать «F11».
7. Исправить ошибки, если таковые имеются.
8. Создать визуализацию проекта. В левой части окна CoDeSys внизу выбрать страницу визуализации (Visualization). На Visualization щелкнуть правой клавишей мыши. В контекстном меню ввода задать команду *Добавить объект...* Присвоить новому объекту имя Lab_IL.

9. Нарисовать элементы визуализации в виде прямоугольников для переменных I0, I1, I2, I3, логических элементов AND, OR. Нарисовать элемент визуализации в виде окружности для переменной Q4.
10. Настроить элементы визуализации – переменные I0, I1, I2, I3, Q4 – аналогично п.п. 11 предыдущего задания.
11. Перейти в режиме эмуляции. Установить флажок в меню Онлайн, Режим эмуляции.
12. Запустить проект. Выполнить команду Онлайн, Подключение. Выполнить команду Онлайн, Пуск (или кнопкой F5).
13. Изменяя значения входных переменных, нажимая на элементы визуализации, наблюдать за изменением выходной переменной. Убедиться, что реализованная схема работает также, как в предыдущем задании.
14. Остановить проект. Выполнить команду Онлайн, Стоп.

3. Исследование языка FBD

1. Создать новый проект. Выполнить команду: Файл, Создать.
2. Выбрать целевую платформу (Target Settings) 3S CoDeSys SP PLCWinNT V2.4.
3. Определить тип первого программного компонента (New POU) – PLC_PRG. Выбрать язык программирования – FBD и тип компонента – программа. В однозадачных проектах система исполнения циклически вызывает программу PLC_PRG.
4. Объявить переменные I0, I1, I2, I3, Q4. В диалоге определения переменных определить тип переменных – BOOL и класс переменных – VAR GLOBAL.
5. В графическом редакторе нарисовать схему согласно своему варианту, используя команды Элемент, Присваивание. При этом переменные и команды блоков присваивать из списка (кнопка F2).



6. Откомпилировать проект. Выполнить команду: Проект, Компилировать или нажать «F11».
7. Исправить ошибки, если таковые имеются.
8. Создать визуализацию проекта. В левой части окна CoDeSys внизу выбрать страницу визуализации (Visualization). На Visualization щелкнуть правой клавишей мыши. В

- контекстном меню ввода задать команду *Добавить объект...* Присвоить новому объекту имя Lab_II.
9. Нарисовать элементы визуализации в виде прямоугольников для переменных I0, I1, I2, I3, логических элементов AND, OR. Нарисовать элемент визуализации в виде окружности для переменной Q4.
 10. Настроить элементы визуализации – переменные I0, I1, I2, I3, Q4 – аналогично п.п. 11 предыдущего задания.
 11. Перейти в режиме эмуляции. Установить флажок в меню Онлайн, Режим эмуляции.
 12. Запустить проект. Выполнить команду Онлайн, Подключение. Выполнить команду Онлайн, Пуск (или кнопкой F5).
 13. Изменяя значения входных переменных, нажимая на элементы визуализации, наблюдать за изменением выходной переменной. Убедиться, что реализованная схема работает также, как в предыдущих заданиях.
 14. Остановить проект. Выполнить команду Онлайн, Стоп.

Варианты индивидуальных заданий

1. $Q0 = I0 \& \bar{I1} \vee I2 \& I3$
2. $Q1 = I4 \& (I4 \vee I6) \& I7$
3. $Q2 = I0 \vee I1 \& I2 \& \bar{I3}$
4. $Q3 = I5 \vee I4 \vee I6 \& I7$
5. $Q4 = I0 \& I1 \vee I2 \vee I3$
6. $Q5 = I4 \& I5 \& (I6 \vee I7)$
7. $Q6 = I0 \& \bar{I1} \& I2 \& I3$
8. $Q7 = I4 \vee I1 \vee I2 \vee I3$
9. $Q0 = I4 \& I5 \vee I2 \& I3$
10. $Q1 = I0 \& (\bar{I1} \vee I2) \& I3$
11. $Q2 = I5 \& I6 \vee I2 \& \bar{I3}$
12. $Q3 = I0 \& I1 \& I2 \& \bar{I3}$
13. $Q4 = I0 \& I1 \& I4 \vee I5$
14. $Q5 = I0 \& \bar{I1} \vee I2 \& I5$

$$15. \quad Q6 = I0 \& I1 \vee I2 \vee \bar{I3}$$

$$16. \quad Q0 = I4 \& \bar{I5} \vee I2 \& I3$$

$$17. \quad Q7 = I2 \& I1 \vee \bar{I3} \vee I4$$

& - логическое «И»

\vee – логическое «ИЛИ»

Содержание практической работы и требования к отчету

В результате выполнения практической работы должны быть разработаны:

- главная программа PLC_PRG на трех языках программирования (Список Инструкций (IL), Диаграммы Функциональных блоков (FBD) и Релейно-контактные схемы (LD));
- визуализация проекта для каждого языка программирования;
- отчет.

Отчет должен содержать следующие разделы:

- задание;
 - введение, в котором описывается решаемая задача, кратко описываются разделы лабораторной работы;
 - разработка главной программы PLC_PRG на трех языках программирования (Список Инструкций (IL), Диаграммы Функциональных блоков (FBD) и Релейно-контактные схемы (LD));
 - разработка визуализации проекта для каждого языка программирования;
 - результаты вычислительного эксперимента в виде таблицы истинности;

выводы.

Практическая работа №2

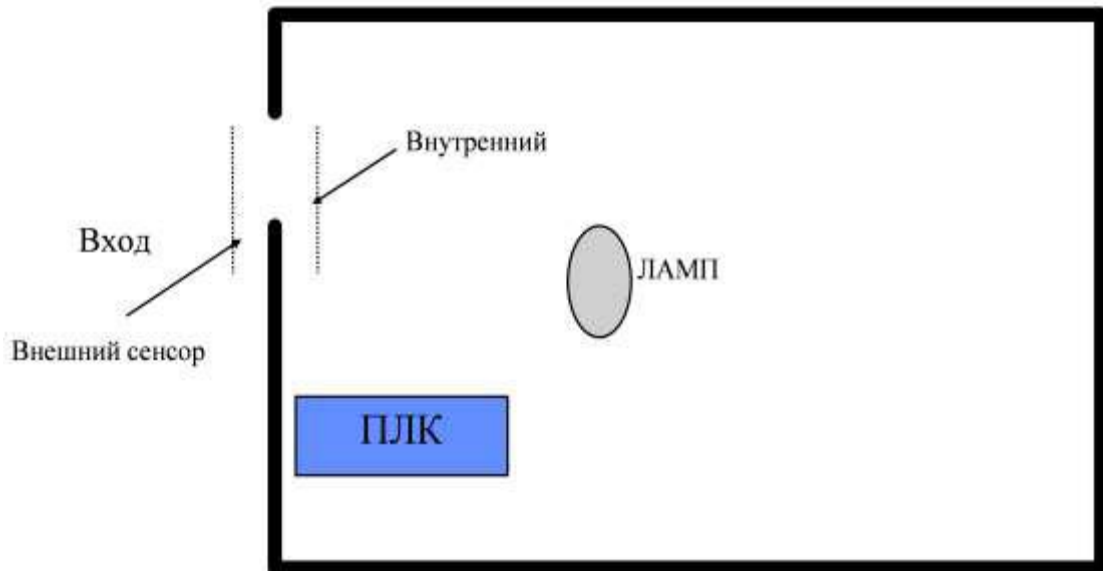
Примеры создания в CoDeSys систем управления простыми объектами

Цель работы: приобретение и закрепление навыков программирования в среде программного комплекса CoDeSys путем создания систем управления простыми объектами.

1. Управление освещением в комнате

Цель - свет должен быть выключен, когда в комнате никого нет.

На входе установлены два дискретных датчика: один снаружи комнаты, другой внутри. Когда срабатывает сначала внешний датчик, затем внутренний, это означает, что человек зашел в комнату. Когда срабатывает сначала внутренний датчик, затем внешний, это означает, что человек вышел из комнаты.

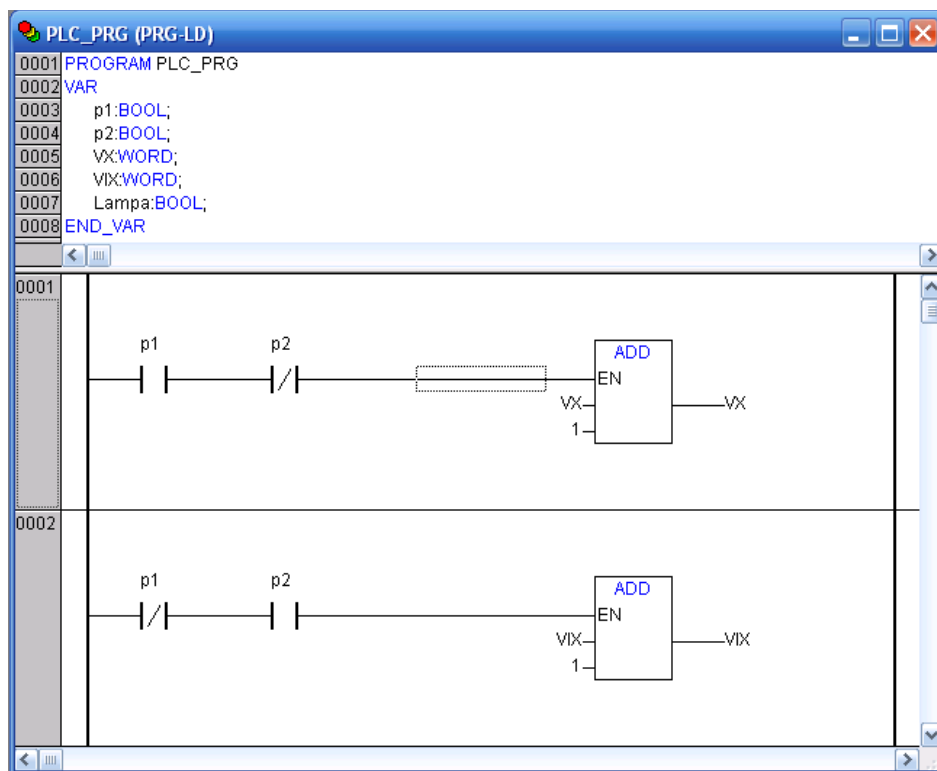


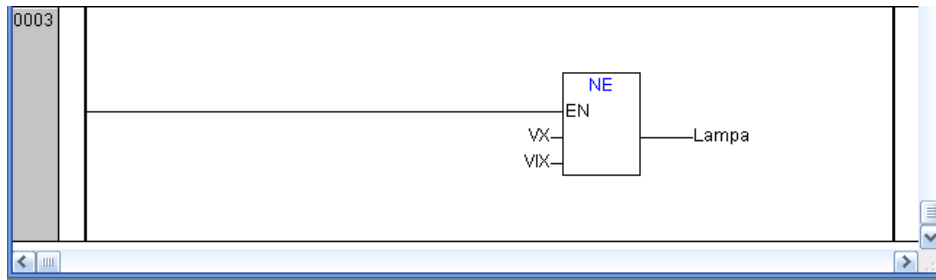
Если человек вошел – включить свет, если человек вышел – выключить свет.

Необходимо считать количество людей, заходящих и выходящих из комнаты. Пока в комнате остается хотя бы один человек, свет должен быть включен.

1. Создадим новый проект, для чего проделаем первые шаги для программирования в CoDeSys, которые описаны в практической работе №1.

2. Реализуем систему управления освещением на языке программирования LD:





Комментарии:

- Для решения данной задачи необходимо было считать количество входящих в комнату людей (VX) и выходящих (VIX).
- Когда срабатывает сначала внешний датчик (p1), затем внутренний (p2), это означает, что человек зашел в комнату, и счетчик считает входящих людей (VX).
- Когда срабатывает сначала внутренний датчик (p2), затем внешний (p1), это означает, что человек вышел из комнаты, и пошел счет выходящих людей (VIX).
- Затем сравнив результаты подсчетов, можно сделать вывод: если количество входящих в комнату людей равно количеству выходящих, то лампочка гаснет, если не равно (NE), то горит.

3. Реализуем систему управления освещением на языке программирования SFC с использованием конфигурирования входов-выходов контроллера:

На входе установлены два дискретных датчика: один снаружи (**ex**), другой внутри комнаты (**in**).

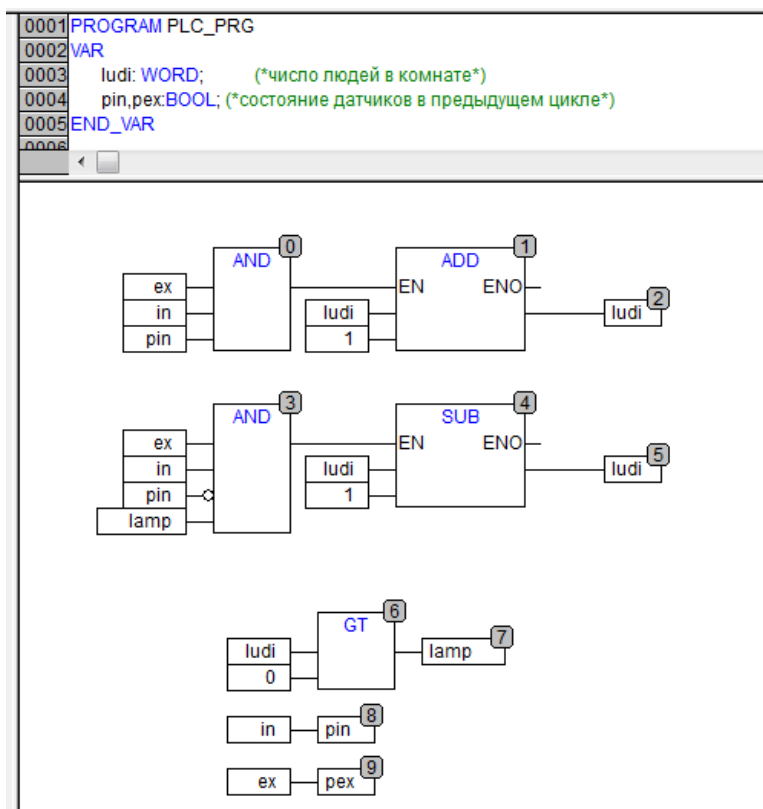
Когда срабатывает сначала внешний датчик, затем внутренний, это означает, что человек зашел в комнату.

Когда срабатывает сначала внутренний датчик, затем внешний, это означает, что человек вышел из комнаты.

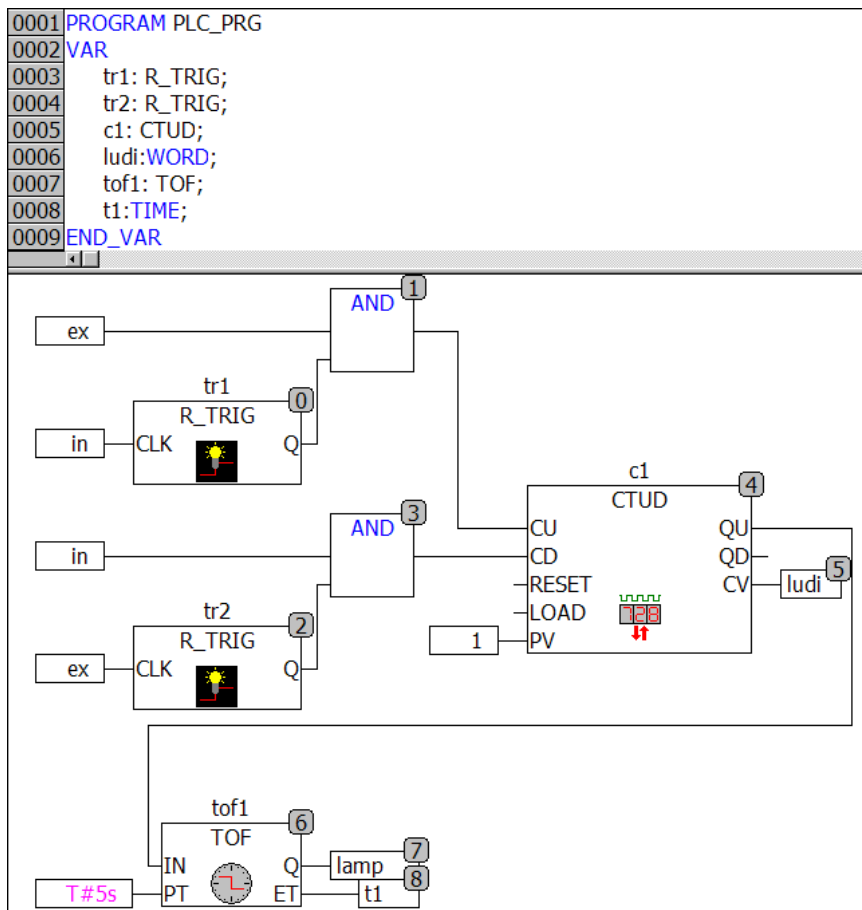
Необходимо считать количество людей (**ludi**) в комнате.

Если человек вошел – включить свет (**lamp**), если человек вышел – выключить свет (**lamp**).

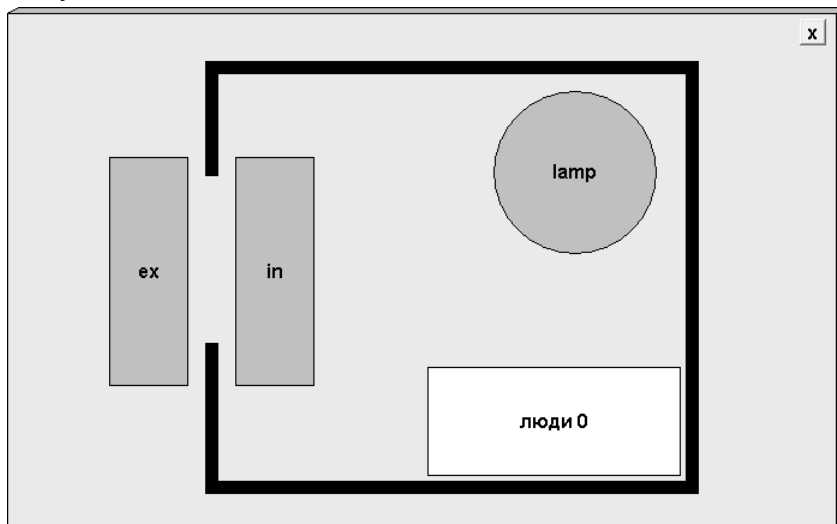
Пока в комнате остается хотя бы один человек, свет должен быть включен.



4. Реализуем систему управления освещением на языке программирования CFC с использованием таймеров, предусмотрев, что если вышел последний человек – выключить свет (**lamp**) с задержкой 5 сек.

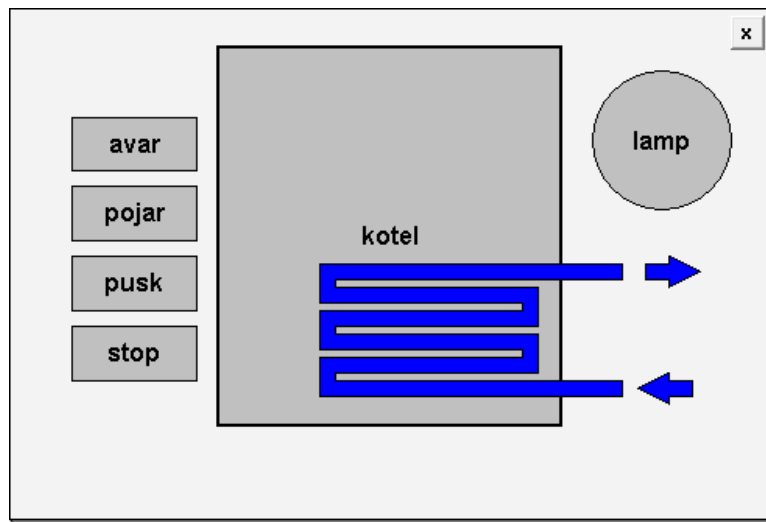


Визуализация:



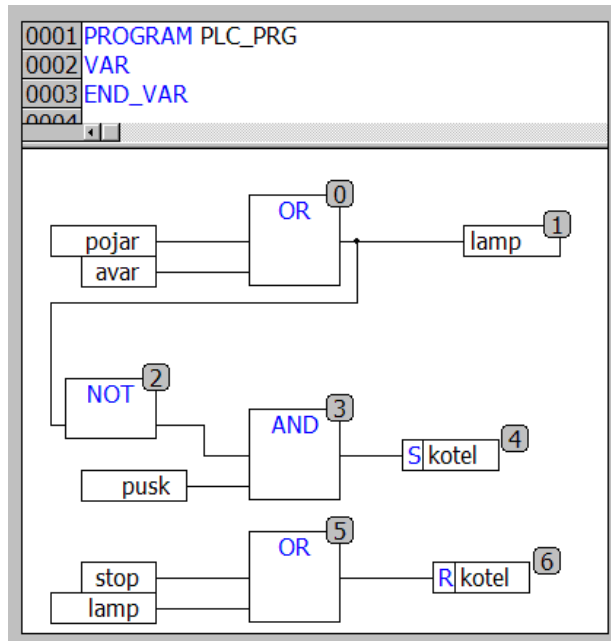
Необходимо реализовать проект на двух языках программирования (LD, CFC), разработать визуализацию, проверить работу проекта в режиме эмуляции.

2. Управление котлом



Необходимо реализовать с использованием языка программирования CFC:

- Включение сигнализации (**lamp**) при возникновении любой из аварий (**avar** или **pojar**).
- Отключение котла (**kotel**) при возникновении любой из аварий.
- Включение котла с кнопки (**pusk**), при условии отсутствия аварий.
- Отключение котла с кнопки (**stop**).



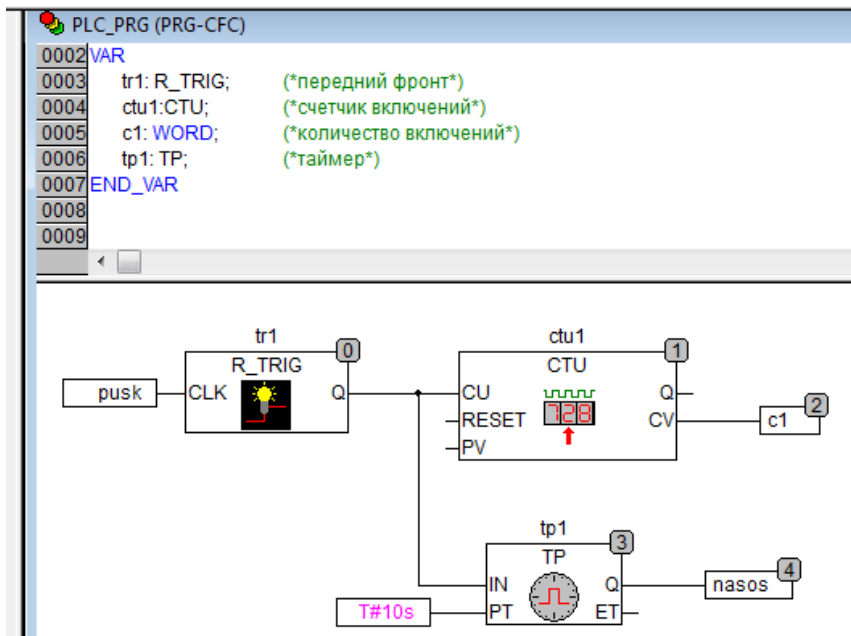
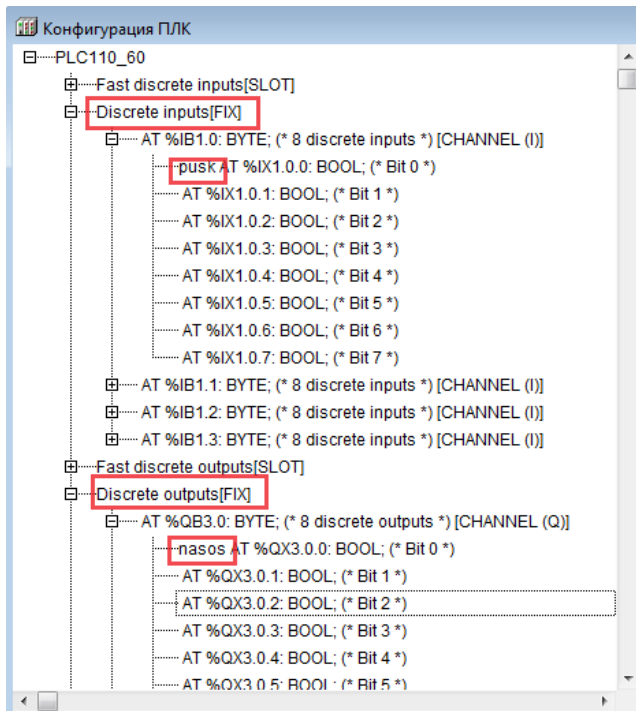
Необходимо реализовать проект на языке программирования CFC, разработать визуализацию, проверить работу проекта в режиме эмуляции.

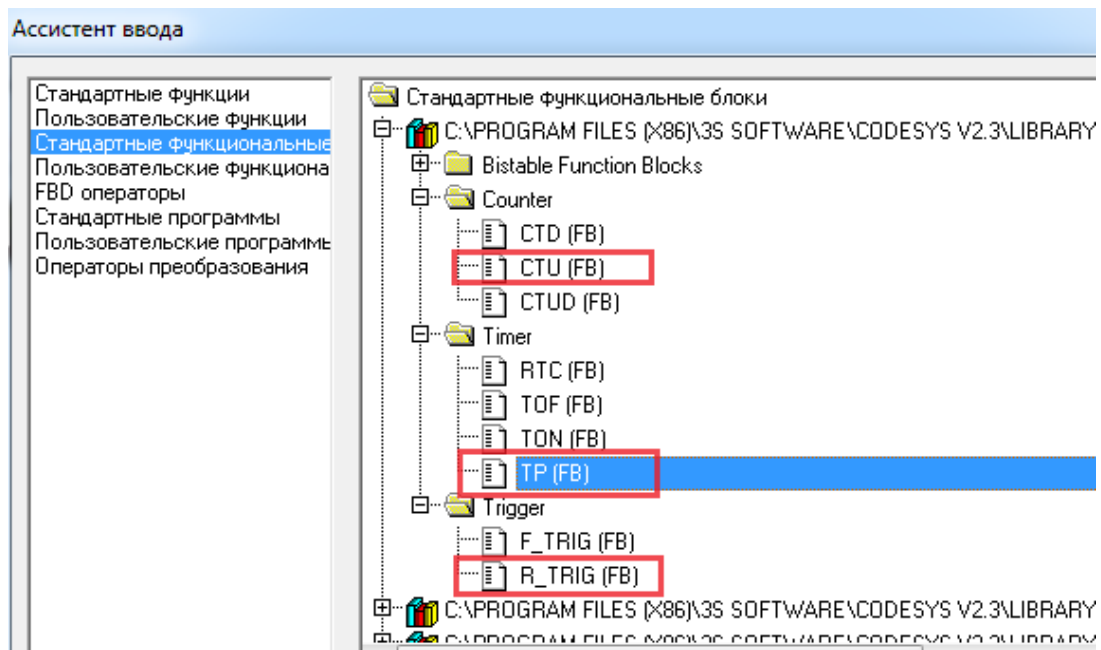
3. Управление включением насоса с задержкой по времени



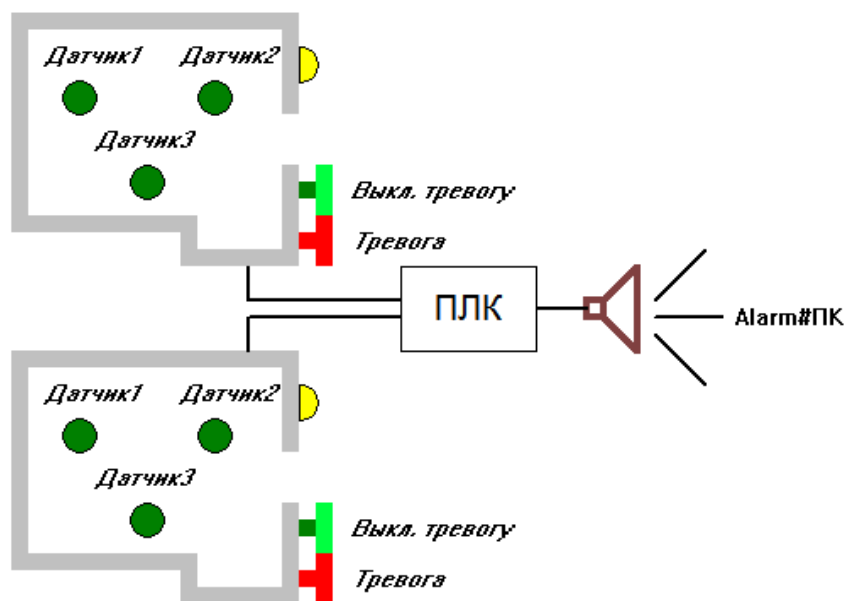
При нажатии на кнопку (**pusk**), насос (**nasos**) должен включиться и проработать 10 секунд, затем автоматически отключиться.

Необходимо подсчитывать количество включений (**c1**) двигателя.





4. Система пожарной сигнализации здания



В здании две одинаковые комнаты.

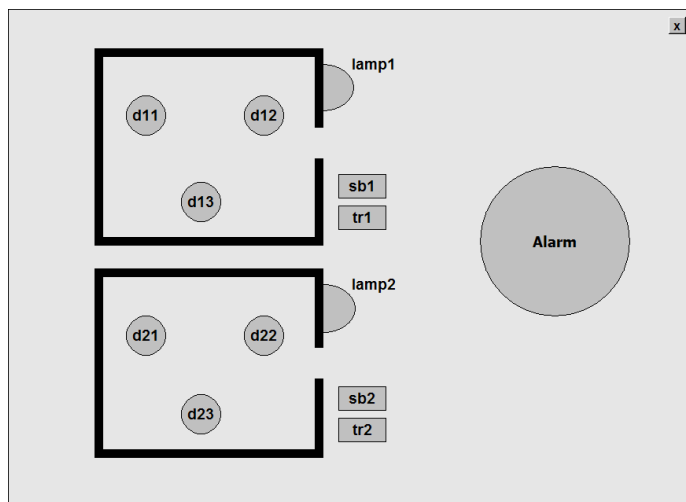
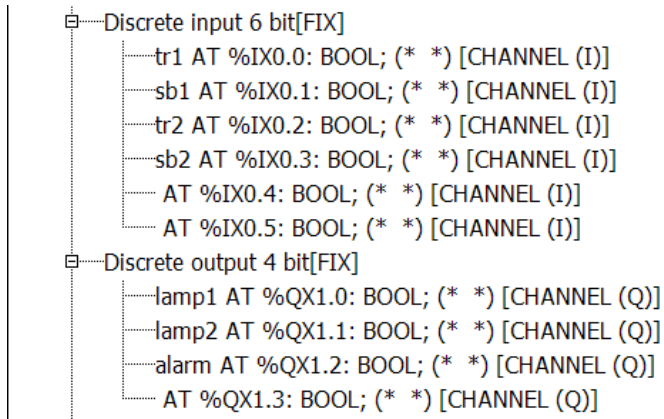
В каждой комнате установлено три пожарных датчика (**d11, d12, d13** и **d21, d22, d23**), кнопка ручного включения сигнализации (**tr1** и **tr2**) и кнопка ручного отключения сигнализации (**sb1** и **sb2**).

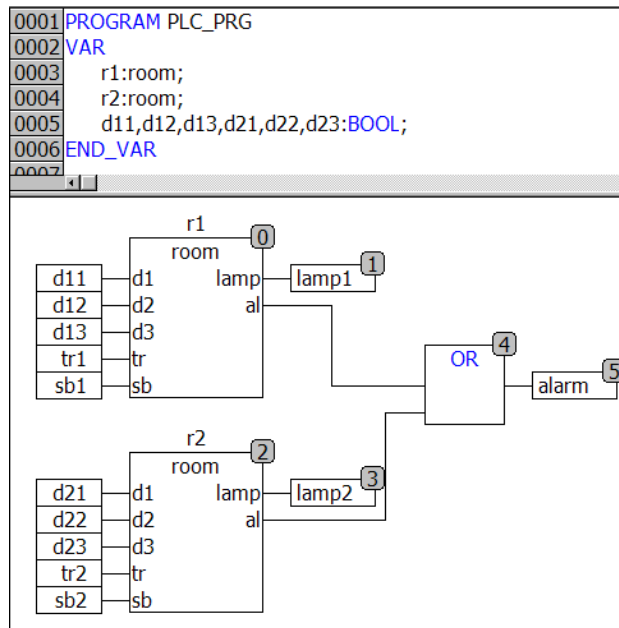
Для каждой комнаты предусмотрена сигнальная лампа (**lamp1, lamp2**). Сигнализация пожара (**alarm**) является общей для обеих комнат.

Если в комнате срабатывает хотя бы один из датчиков, то загорается сигнальная лампа для соответствующей комнаты. Лампа гаснет, если все датчики в комнате отключены.

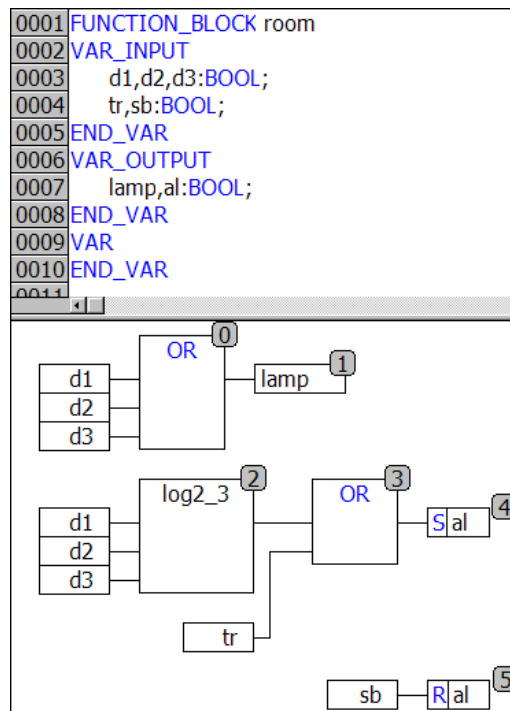
Если в комнате срабатывает любые два из трех датчиков, то включается пожарная сигнализация. Сигнализация работает до тех пор, пока ее не отключат соответствующей кнопкой.

Сигнализация может быть включена кнопкой проверки вне зависимости от состояния датчиков.





Функциональный блок создается отдельно:



Практическая работа №3

Реализация системы управления ленточным конвейером в среде CoDeSys

Цель работы: реализация в среде программного комплекса CoDeSys системы управления ленточным конвейером.

1. Создание блока управления трехфазным асинхронным электродвигателем с прямым пуском

Создадим функциональный блок с именем **PRIVOD** на языке программирования **LD** (рис.1).

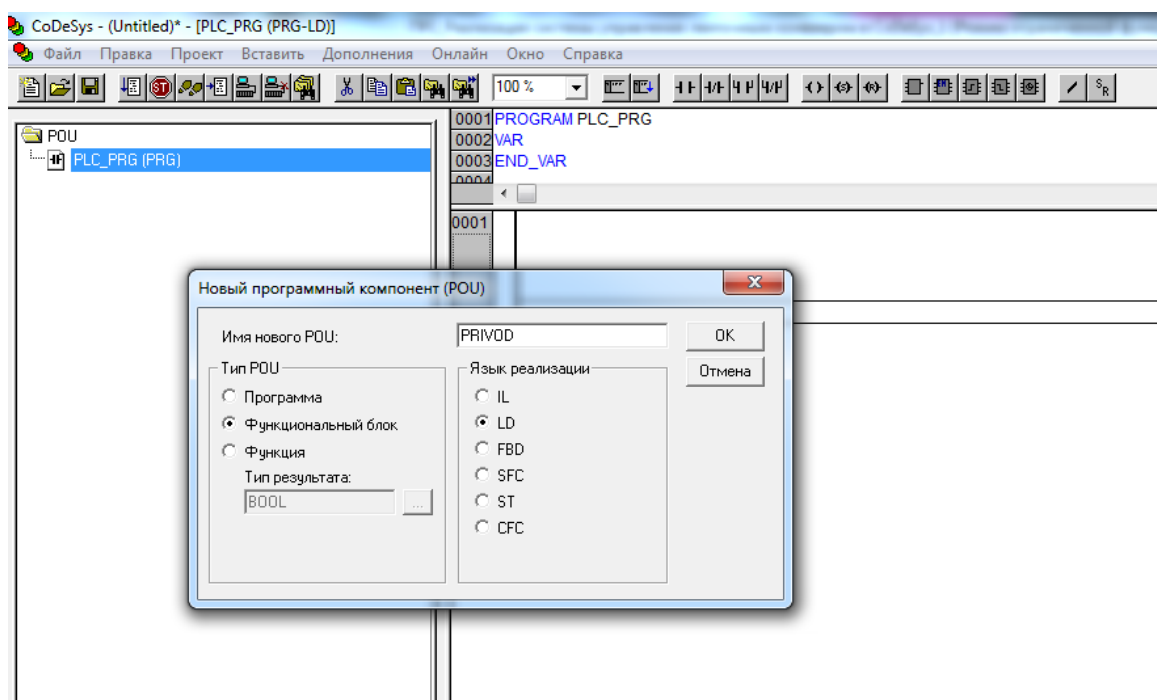


Рис. 1. Создание функционального блока с именем **PRIVOD**

Для включения трехфазного асинхронного электродвигателя с короткозамкнутым ротором (далее «двигателя», позиция на рисунке =9.3-M131) по схеме по схеме «прямого пуска» требуются автоматический выключатель для защиты двигателя (позиция =9.3-QF131), выполняющий функции защиты от токов короткого замыкания и перегрузки двигателя (т.е. выполняет еще и функцию теплового реле), а также контактор (позиция =9.3-KM131). С дополнительных контактов автомата и контактора подаются сигналы на вход ПЛК для контроля состояния пусковой аппаратуры. Катушка контактора подключается к выходу ПЛК.

Сигнал с автоматического выключателя свидетельствует о том, что автоматический выключатель включен (т.е. нет аварийного отключения по току короткого замыкания или току перегрузки). Таким образом, отсутствие этого сигнала следует отнести к аварийной ситуации.

Сигнал с контактора свидетельствует о том, что на электродвигатель подано напряжение, т.е. что двигатель включен и соответствующий технологический агрегат

запущен. Данный сигнал используется в цепи «самоподхвата» (включается параллельно с пусковой кнопкой).

Включается контактор сигналом с выхода ПЛК, или же через промежуточное реле, если параметры выходов контроллера не позволяют включить катушку контактора напрямую. Логика включения контактора определяется исходя из технологии автоматизируемого объекта.

Помимо силовой пусковой аппаратуры необходимы управляющие сигналы – сигнал запуска и сигнал останова. Эти сигналы могут приходиться от разных устройств, их может быть несколько. Как правило, необходимо реализовать как минимум два источника сигналов: сигналы с местного поста управления и сигналы с поста оператора. Это могут быть также сигналы из SCADA-системы. Таким образом, необходимо реализовать логику управления для всех источников сигналов с учетом режима их работы (наладочный, ручной или автоматический).

Также нужно реализовать обработку аварийных сигналов, которые приходят от соответствующих первичных преобразователей, датчиков и вторичных приборов. Количество этих сигналов зависит от конкретного технологического агрегата и его технического оснащения. Как упоминалось выше, к аварийным сигналам также можно отнести отсутствие сигнала с автоматического выключателя двигателя.

Для того, чтобы электродвигатель включался в соответствии с технологией, нужно предусмотреть сигнал готовности, который будет свидетельствовать о том, что по технологической цепочке привод можно запускать в работу. Аналогично нужно реализовать блокировку отключения, чтобы остановка агрегатов происходила в обратном порядке.

Кроме того, можно предусмотреть сигналы включения сигнальных ламп, которые зачастую устанавливаются на местных постах управления и совмещаются с кнопками запуска и останова. Чаще всего, лампа запуска работает в двух режимах: «Мигание» - привод готов к запуску, «Свечение» - привод запущен. Лампа останова также работает в двух режимах: «Свечение» - привод останавливается (ждет сигнала останова в соответствии с технологией), «Мигание» - авария технологического агрегата. Если ни одна лампа не горит, значит привод остановлен и запуск его запрещен. Можно также предусмотреть шифровку аварий посредством изменения частоты мигания лампы останова.

Дополнительно можно предусмотреть выход для формирования звукового сигнала при запуске и аварии привода.

Исходя из вышесказанного, объявление функционального блока **PRIVOD** будет выглядеть следующим образом:

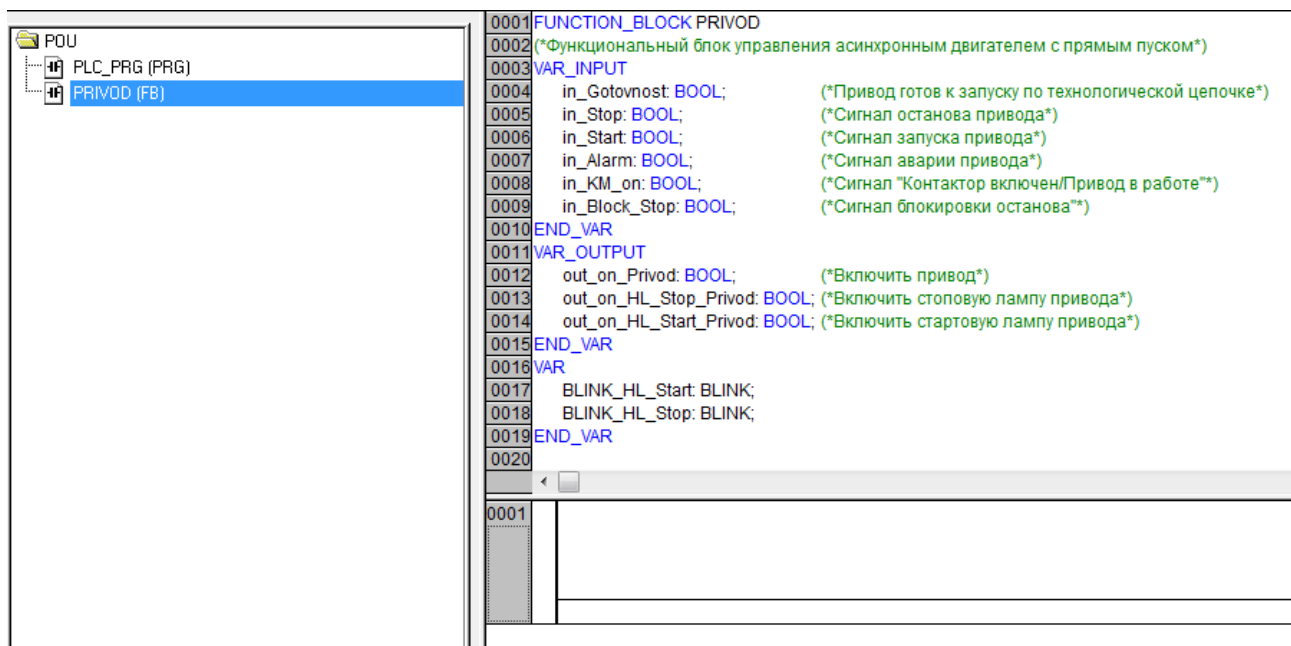


Рис.2. Список входных и выходных переменных функционального блока PRIVOD

!!! По мере написания функционального блока в раздел VAR будут добавлены локальные переменные для выполнения промежуточных операций !!!

Итак, приступим к написанию кода блока PRIVOD.

Для **включения привода** нам необходимо выполнение следующей логической цепочки:

- должен быть сигнал разрешения запуска по технологической цепочке;
- не должно быть сигнала аварии;
- не должно быть сигнала останова;
- есть сигнал запуска или самоподхвата.

При этом, при наличии сигнала останова и сигнала блокировки останова, в случае отсутствия аварийного сигнала, привод не должен отключаться.

Включение стартовой лампы осуществляется следующим образом: если нет аварийного и стопового сигналов, то в случае, когда привод запущен, лампа горит, а в случае, когда привод не запущен, но готов к запуску, лампа мигает.

Включение стоповой лампы осуществляется следующим образом: лампа горит, когда есть сигнал останова, и мигает, когда есть сигнал аварии.

Мигание осуществляется с помощью функционального блока **BLINK** из библиотеки *Util.lib*. Для вставки функционального блока нужно щелкнуть правой кнопкой мыши на цепи, в которую его нужно вставить, и в контекстном меню выбрать **Функциональный блок**, или нажать сочетание клавиш **Ctrl+B**, или нажать кнопку **Элемент** на панели инструментов. При этом появится окно выбора функциональных блоков, где нужно выбрать соответствующую библиотеку и в ней найти требуемый функциональный блок (**BLINK** находится в папке *signals*).

В результате, код функционального блока **PRIVOD** будет выглядеть следующим образом:

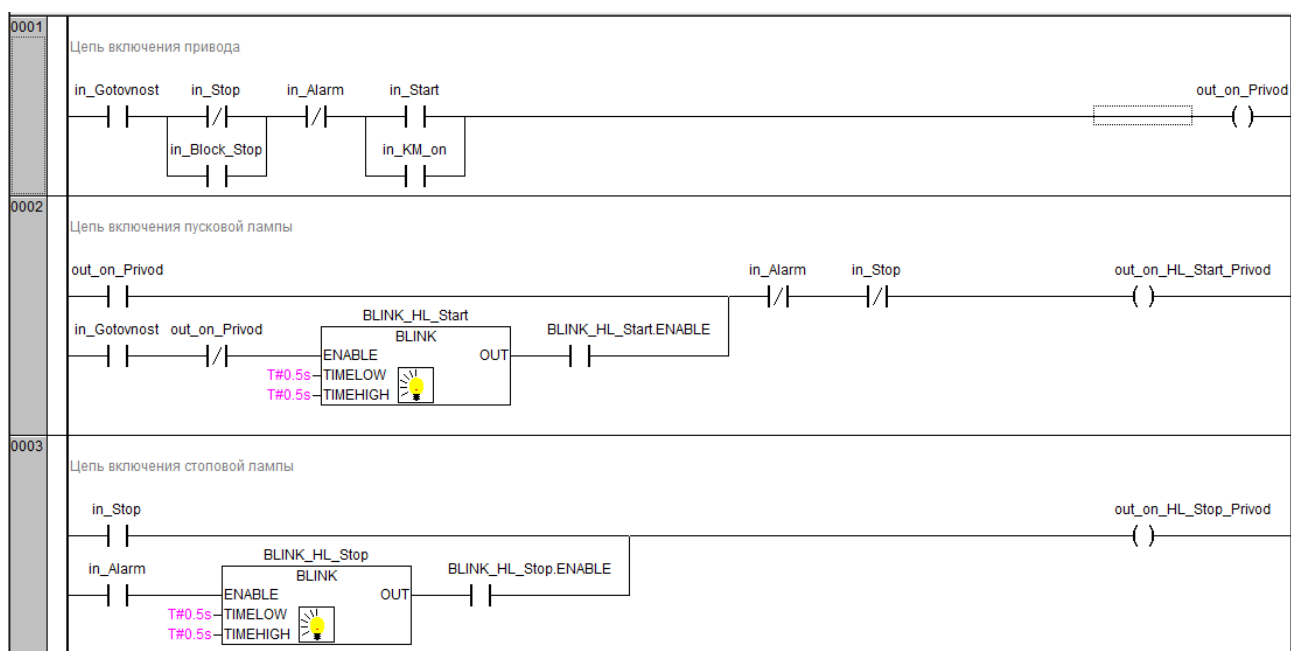


Рис.3. Код функционального блока PRIVOD

2. Вызов функционального блока PRIVOD и определение конфигурации контроллера

Для начала поставим перед собой простую задачу. Предположим, что у нас есть шлюзовой питатель, который подает сыпучий материал из бункера на ленточный конвейер, который в свою очередь транспортирует этот материал в конечную точку назначения. Нам нужно управлять запуском приводов этих агрегатов, осуществляя элементарную технологическую цепочку. В дальнейшем будем расширять рамки системы управления, но на начальном этапе реализуем следующие функции:

- контроль последовательности включения и отключения приводов;
- отработку аварийных сигналов;
- запуск и останов приводов по входным сигналам «Пуск» и «Стоп».

Начнем с ленточного конвейера. Мы уже знаем, что у нас должно быть три сигнала от пусковой аппаратуры:

- «Автоматический выключатель включен» (вход);
- «Контактор включен» (вход);
- «Включить контактор» (выход).

Кроме того, у нас должно быть четыре сигнала для светосигнальной аппаратуры: кнопка «Пуск» (вход) и лампа, совмещенная с кнопкой «Пуск» (выход); кнопка «Стоп» (вход) и лампа, совмещенная с кнопкой «Стоп» (выход).

Аналогично, такие сигналы будут использованы и для других агрегатов с асинхронными трехфазными двигателями, в том числе и для шлюзового питателя.

Также, для ленточного конвейера используются датчики противоаварийной защиты: трос безопасности (зачастую совмещенный с кнопкой аварийного останова), датчик схода ленты, а также датчик движения ленты.

Трос безопасности (и/или кнопка аварийного останова) предназначен для останова конвейера технологическим персоналом в случае возникновения аварийной ситуации (рис.11). При натяжении троса размыкается контакт концевого выключателя и разрывается цепь аварийного останова. При этом может разрываться питание оперативных цепей управления, что приведет к мгновенному пропаданию напряжения на катушке контакторов и останову технологической линии. Возможно формирование аварийного сигнала в контроллере; при этом останов должен быть безусловным, т.е. однозначно и мгновенно останавливать технологическую линию.

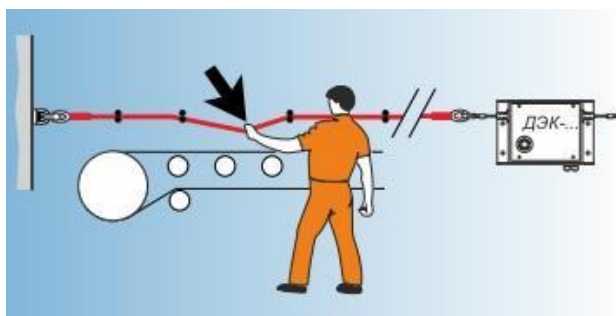


Рис.3. Принцип действия троса безопасности

Датчик схода ленты предназначен для предотвращения схода ленты с опорных роликов. Сход ленты – серьезная аварийная ситуация. Даже если при этом не произойдет повреждения ленты, восстановление работы конвейера после схода – сложная и трудоемкая задача. Поэтому практически все ленточные конвейеры оснащаются датчиками схода ленты, иногда (когда конвейер длинный, например, более 30 метров) несколькими парами. Принцип действия датчиков схода ленты можно уяснить из рис.4.

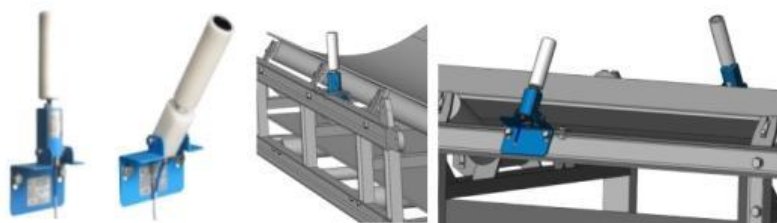


Рис.4. Принцип действия датчиков схода ленты

Датчик движения вала устанавливается на ведомый барабан ленточного конвейера. Он представляет собой индукционный датчик наличия материала, мимо которого проходит металлическая метка, наваренная на барабан. Таким образом, каждый оборот метки проходит мимо датчика, в результате чего возникает импульс. Наличие импульсов с заданной частотой говорит о нормальной работе конвейера. Если же конвейер включен, а частота импульсов равна нулю, значит, имеет место пробуксовка ленты, что нередко бывает в зимний период времени. В результате лента может быть повреждена, что повлечет немалые материальные и трудовые затраты. Поэтому необходимо контролировать наличие изменяющегося сигнала с датчика движения. Принцип действия датчика движения приведен на рис.5.



Рис.5. Принцип работы датчика движения

Таким образом, для ленточного конвейера нужно предусмотреть еще как минимум три сигнала с аварийных датчиков.

Итак, в общей сложности для ленточного конвейера нужно выделить 7 входов и 3 выхода.

Практика показывает, что пространство входов и выходов ПЛК целесообразно организовывать, основываясь на логических критериях их использования. К примеру, можно условиться, что первые 8 входов и 8 выходов будут отведены под общие нужды системы управления (например, сигналов «Наладочный режим», «Сброс аварии», «Звуковая аварийно-предупредительная сигнализация» и т.д.). Далее группами по 8 входов и 4 выхода (по 1 резервному входу и выходу на агрегат) будут идти сигналы технологических агрегатов.

При этом переменные, относящиеся к определенному агрегату должны содержать его условное обозначение (например, постфикс или префикс **LK**). Это существенно повышает наглядность кода.

Теперь, основываясь на вышеприведенных соображениях, создадим в конфигурации ПЛК переменные для управления ленточным конвейером. Вид конфигурации приведен на рис.6.

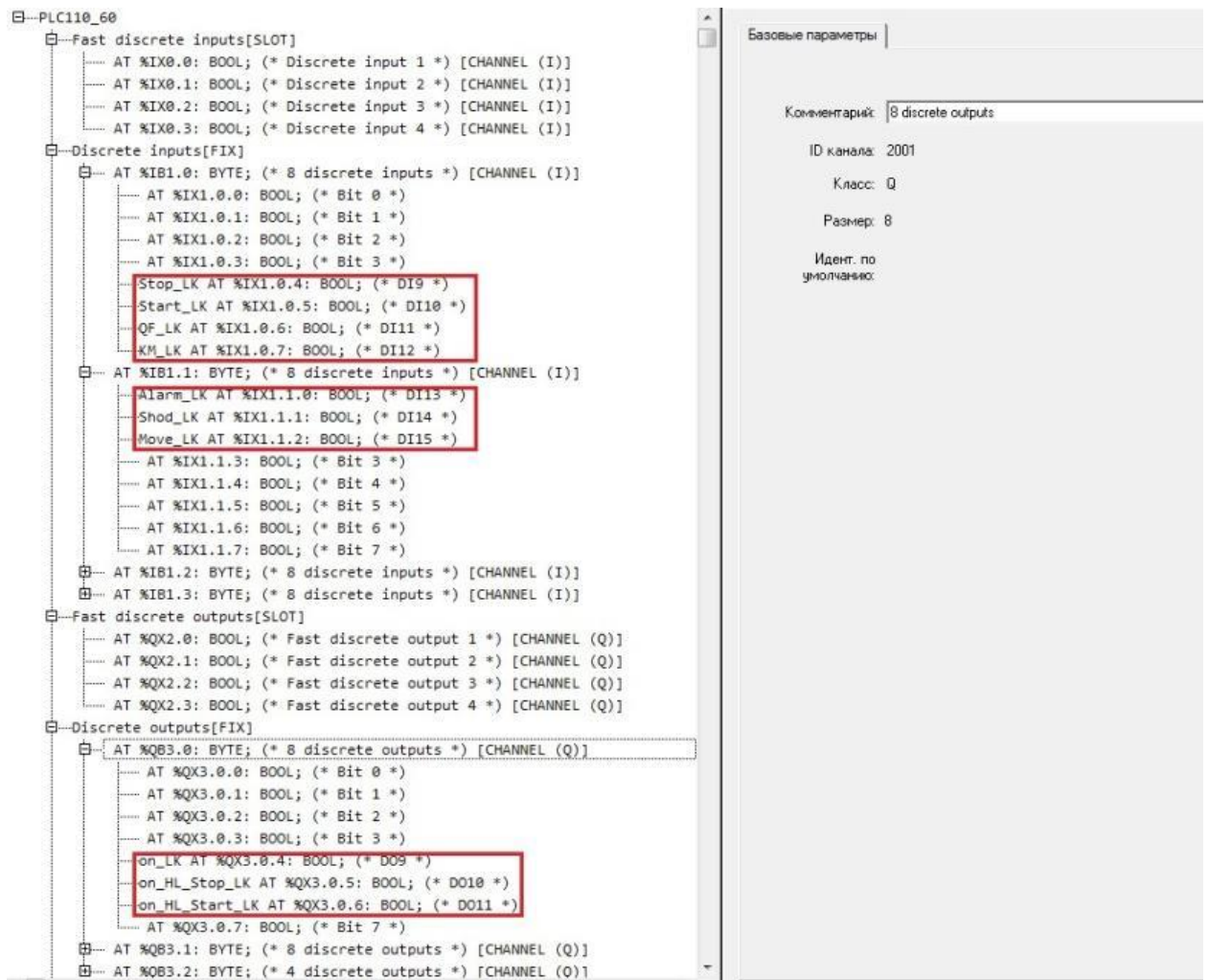


Рис.6. Конфигурация ПЛК с сигналами для ленточного конвейера

Теперь, когда создан блок PRIVOD и объявлены переменные для работы с ленточным конвейером, мы можем создать экземпляр блока PRIVOD для ленточного конвейера. Вызов блока управления ленточным конвейером (**LK**) приведен на рис.7.

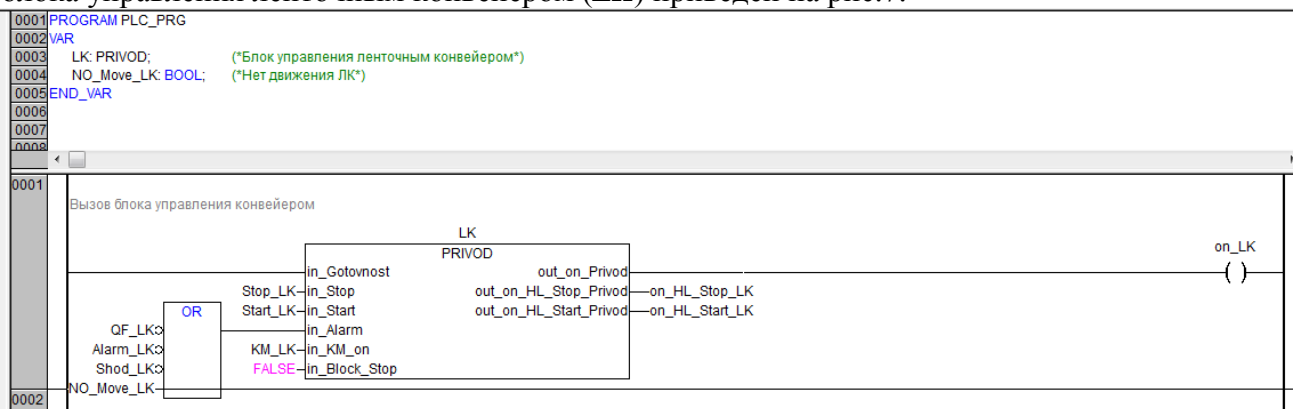


Рис.7. Вызов блока управления ленточным конвейером (LK)

При компиляции проекта ошибок быть не должно. В случае их возникновения внимательно проверьте правильность написания имен всех переменных. После подключения (**Онлайн – Подключение**) в режиме эмуляции (должна быть установлена метка **Режим эмуляции** в меню **Онлайн**) и выставления в конфигурации ПЛК значений переменных **QF_LK**, **Alarm_LK** и **Shod_LK** в значение **TRUE**, вы сможете включать и

отключать ленточный конвейер (DO9 в конфигурации ПЛК) с помощью кнопок (сигналов) *Stop_LK* и *Start_LK*.

По аналогии с блоком ленточного конвейера создадим экземпляр блока шлюзового питателя, назовем его *PIT*, и вызовем его в *PLC_PRG*. Для шлюзового питателя будет использоваться кнопка аварийного останова (вместо троса безопасности), сход ленты использоваться не будет (вход необходимо оставить под резерв), датчик движения функционирует аналогичным образом. Соответственно, нужно объявить переменные для питателя (с постфиксом Pit) в конфигурации ПЛК с резервным входом (точней двумя, с учетом отсутствия сигнала схода ленты) и резервным выходом (рис. 8). А затем необходимо сделать вызов блока с учетом того, что шлюзовой питатель может быть запущен только спустя 3 секунды после запуска ленточного конвейера (рис. 9).

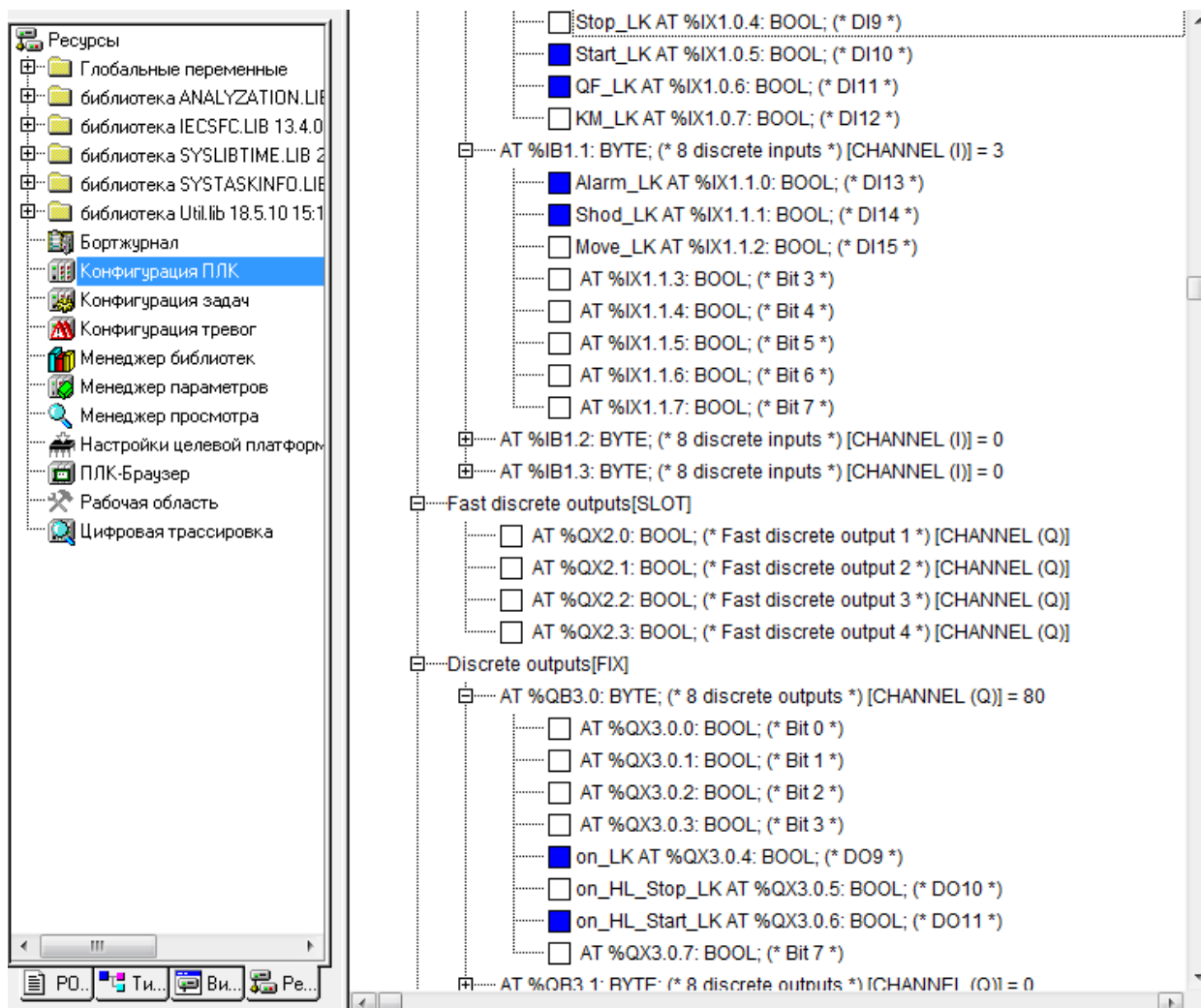


Рис. 8. Управление ленточным конвейером через ресурс «Конфигурация ПЛК»

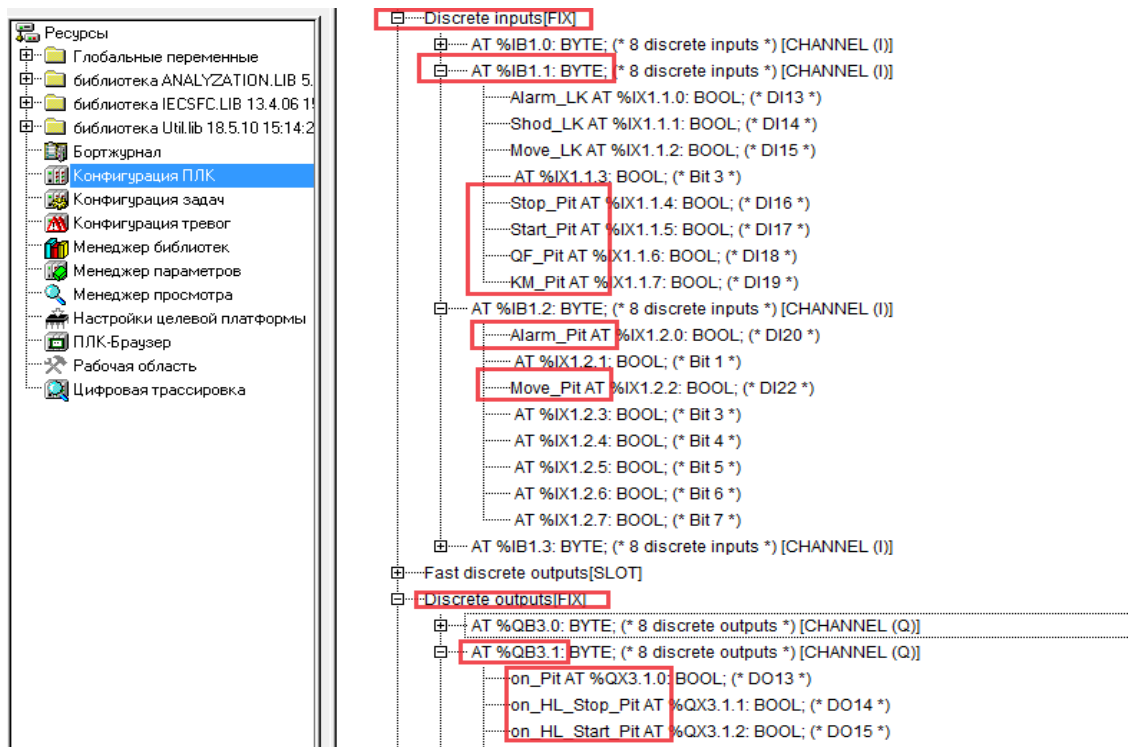


Рис.9. Конфигурация ПЛК с сигналами для шлюзового питателя

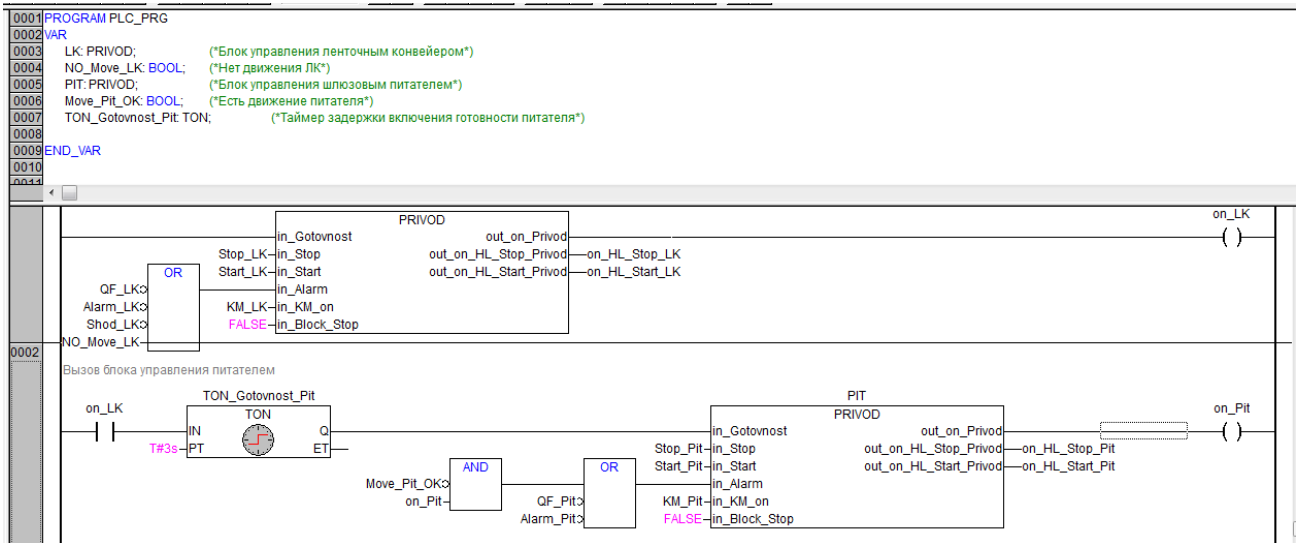


Рис.10. Вызов блока управления шлюзовым питателем (Pit)

Так как входные сигналы ПЛК мы не можем изменять из программы, цепь самоподхвата будет работать некорректно, пока мы не подключим реальный ПЛК и реальный контактор (или хотя бы реле). Поэтому в программе **PLC_PRG** нужно сделать имитацию срабатывания контактора (вставить после каждого из блоков). Логика проста: если есть сигнал на включение соответствующего агрегата, то наш «виртуальный контактор» срабатывает. Эти сигналы нужно подать через блок **OR** вместе с физическим сигналом на соответствующий вход блоков **LK** и **PIT**. В результате цепь самоподхвата станет работать аналогично реальной ситуации (рис. 11).

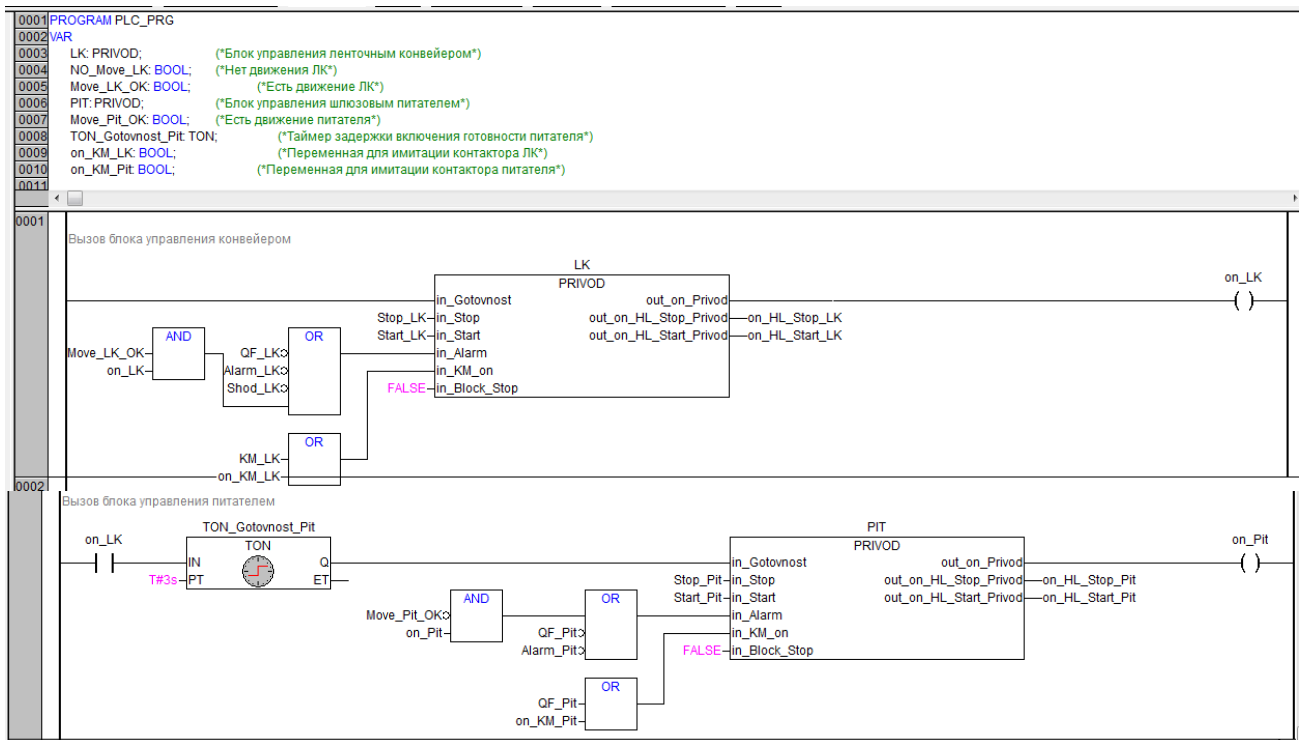


Рис. 11. Имитация срабатывания контакторов *LK* и *PIT*

Кроме того, необходимо реализовать имитацию включения катушки и дополнительного контакта ленточного конвейера и шлюзового питателя (рис.12).

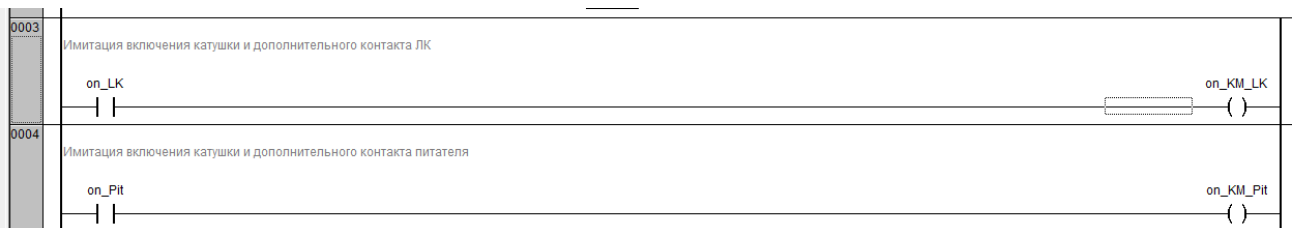


Рис. 12. Имитация включения катушки и дополнительного контакта *LK* и *PIT*

Примечание: если мы просто включим сигнал *in_KM* на входе ПЛК для соответствующего привода, то он будет запускаться без кнопки старт.

Кроме того, не будет и адекватной отработки сигнала с датчика движения, потому что он должен появляться и пропадать с определенной частотой. Таким образом, необходимо реализовать имитацию датчика движения и отработку аварии по отсутствию движения. Имитацию можно осуществить простым блоком **BLINK**, на вход которого будет приходиться сигнал датчика с входа ПЛК. При этом время импульса и время паузы в реальной ситуации соответствуют времени появления сигнала и времени отсутствия, и в сумме должны составлять приблизительно один период вращения вала агрегата, на котором установлен датчик. Обработка этого сигнала также довольно проста: если при включенном приводе в течении 2-3 периодов вращения вала нет изменения сигнала датчика, то необходимо сформировать сигнал аварии и отключить привод.

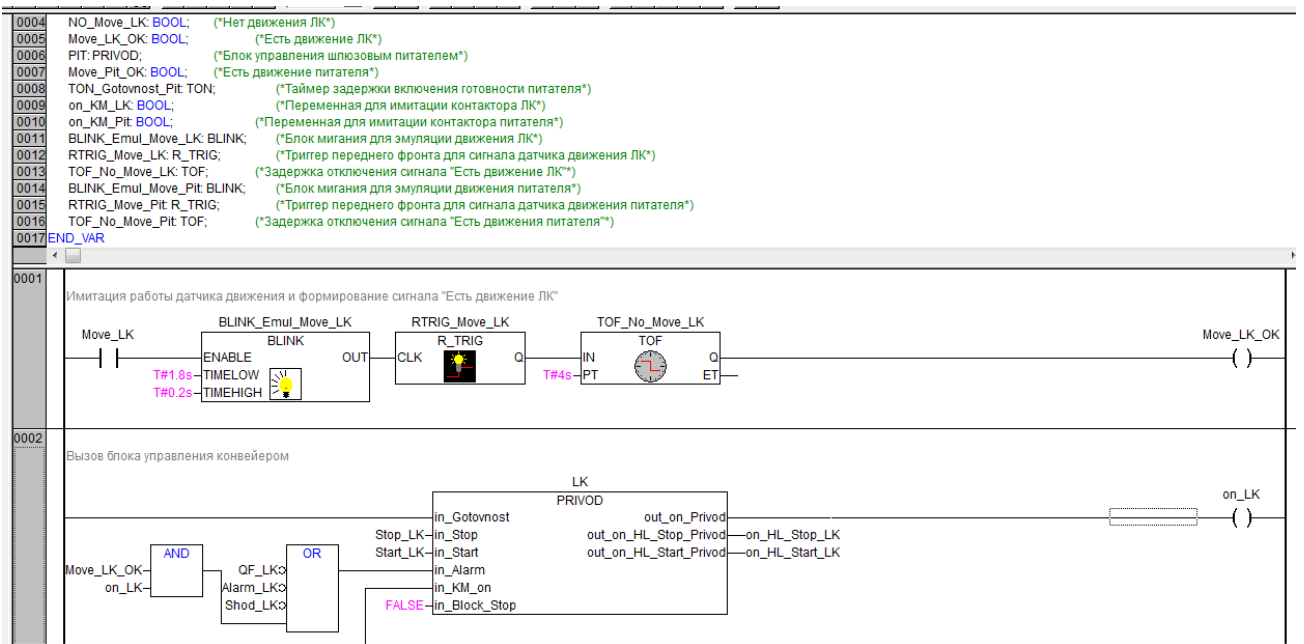


Рис. 12. Имитация датчика движения и формирование сигнала «Есть движение ЛК»

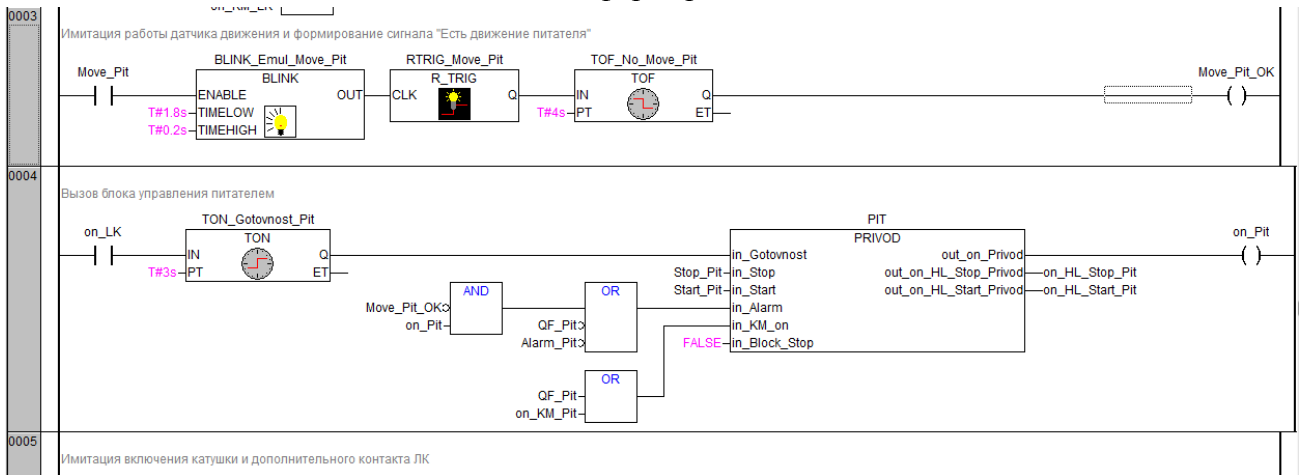


Рис. 13. Имитация датчика движения и формирование сигнала «Есть движение шлюзового питателя»

3. Реализация фиксации аварии и квитирования аварийной ситуации

Фиксация аварии подразумевает, что при возникновении аварийной ситуации сигнал аварии фиксируется и не сбрасывается автоматически после устранения причин аварии, а сбрасывается только по специальному сигналу. Это может быть кнопка на посту управления оператора или сигнал из SCADA-системы.

Предположим, что для сброса аварии используется кнопка, нормально открытый контакт которой подает сигнал на 8 вход контроллера (назовите переменную **Reset_Alarm**). Таким образом, нужно зафиксировать сигнал аварии и сбрасывать его только после

устранения всех неполадок **и нажатия кнопки «Сброс аварии»**. Отсутствие движения при этом в учет не брать – данный сигнал может появиться только после запуска агрегата.

Кроме того, нужно известить персонал о наличии аварии включением звуковой сигнализации, которая будет включаться выходом 8 (назовите переменную **on_Sound_Signal**).

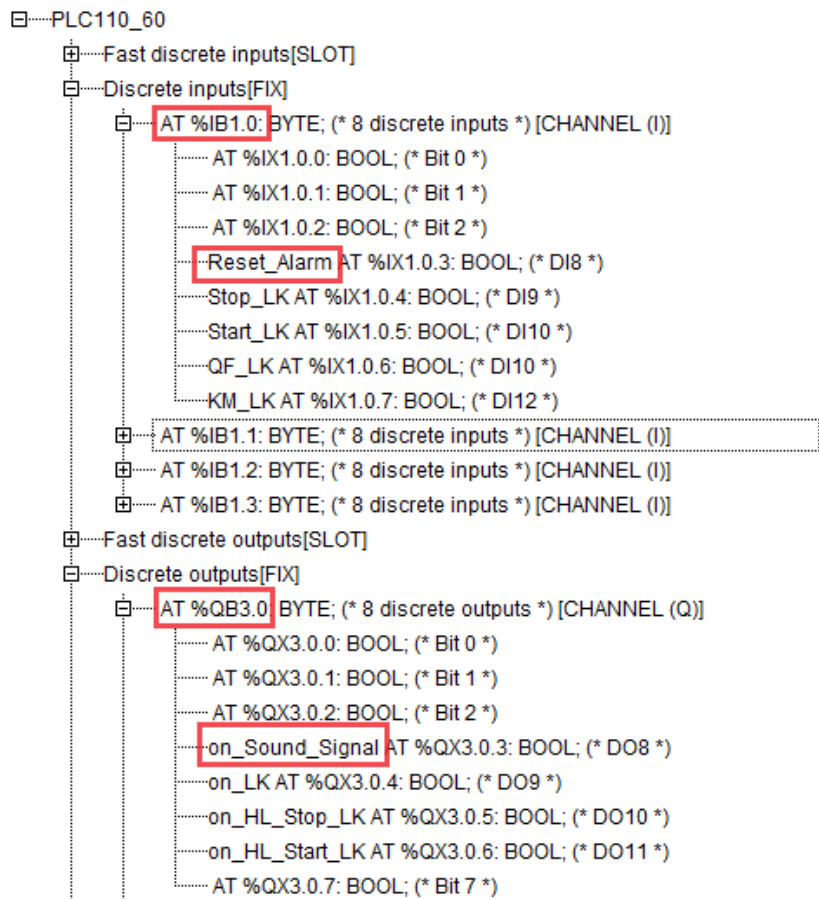


Рис. 14. Введение переменных **Reset_Alarm** и **on_Sound_Signal** в конфигурацию контроллера

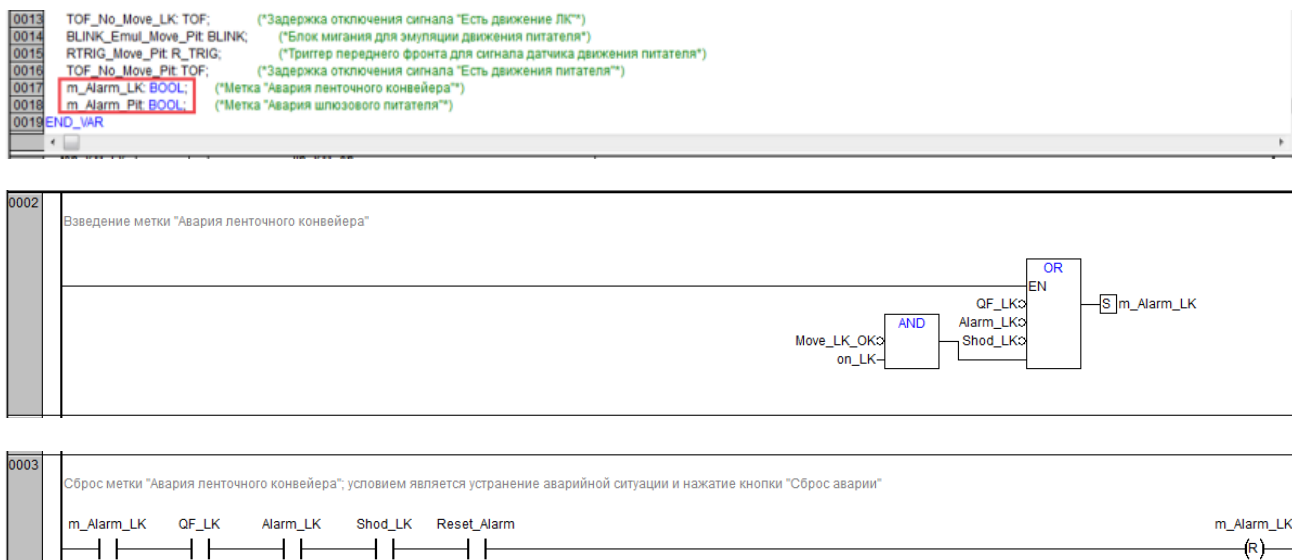




Рис. 15. Фиксация аварии ленточного конвейера

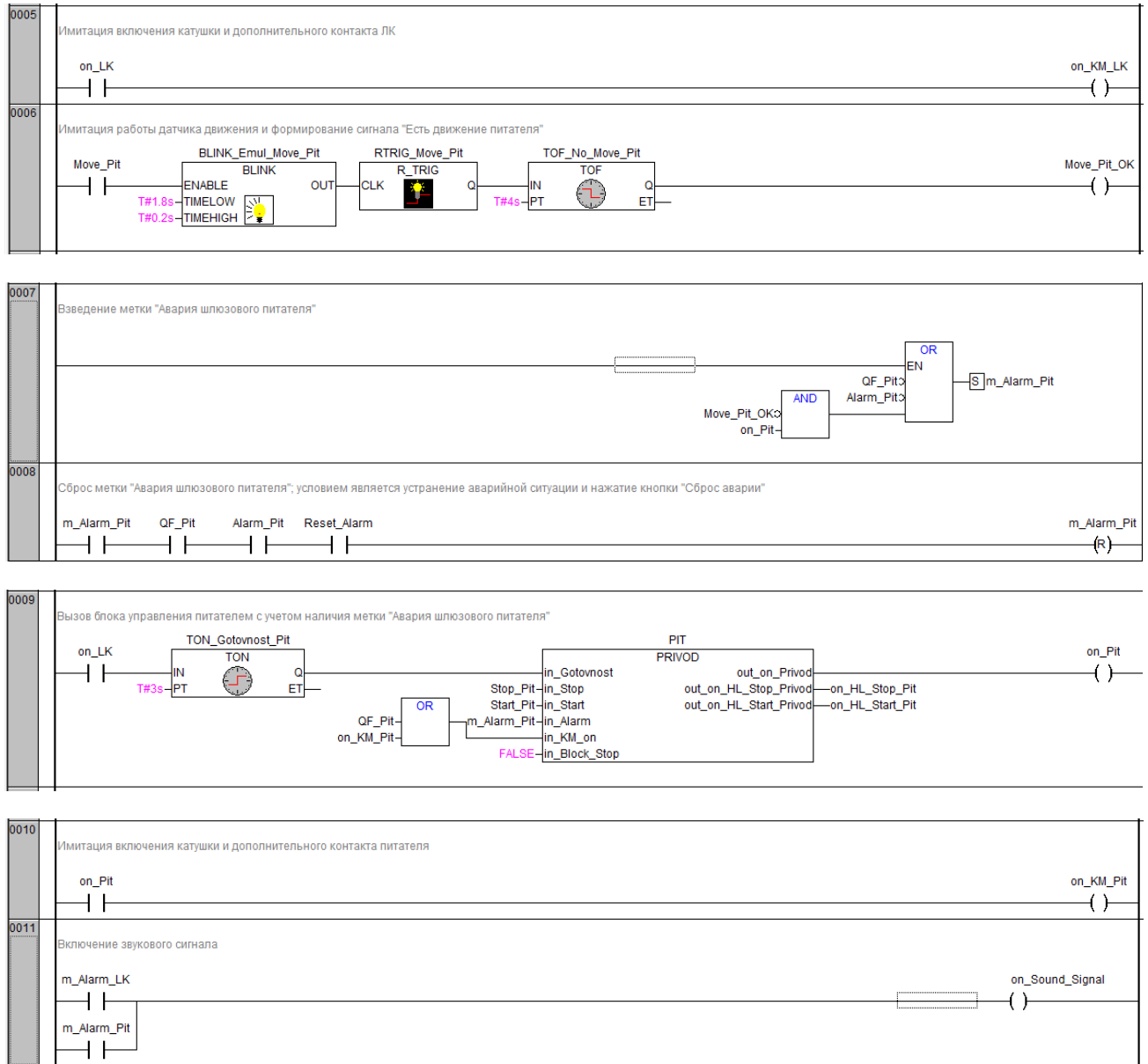


Рис. 16. Фиксация аварии шлюзового питателя и реализация включения звукового сигнала

4. Реализация предупредительной сигнализации

Предупредительная сигнализация является необходимым условием запуска технологической линии. Первый агрегат можно запустить только после проведения предупредительной

сигнализации. Таким образом, на вход готовности первого привода сигнал будет идти не напрямую, а с соответствующим логическим условием.

Предположим, что для включения предпусковой сигнализации используется кнопка, нормально открытый контакт которой подает сигнал на 7 вход контроллера (назовите переменную *Start_Warn*). Таким образом, *при нажатии кнопки «Предпусковая сигнализация»* нужно включить предпусковую сигнализацию на определенное время (примем его равным 5 секунд) и по окончании предпусковой сигнализации дать готовность к запуску первому приводу (в нашем случае, ленточному конвейеру). Пусть с кнопкой предпусковой сигнализации совмещена лампа предпусковой сигнализации, которая включается выходом 7 ПЛК (назовите переменную *on_HL_Start_Warn*). Эта лампа должна мигать, если предпусковая сигнализация не было проведена, и гореть – если предпусковая сигнализация уже прошла.

Кроме того, нужно известить персонал о начале запуска линии включением звуковой сигнализации, которая будет включаться выходом 8 (переменная *on_Sound_Signal* была введена ранее).

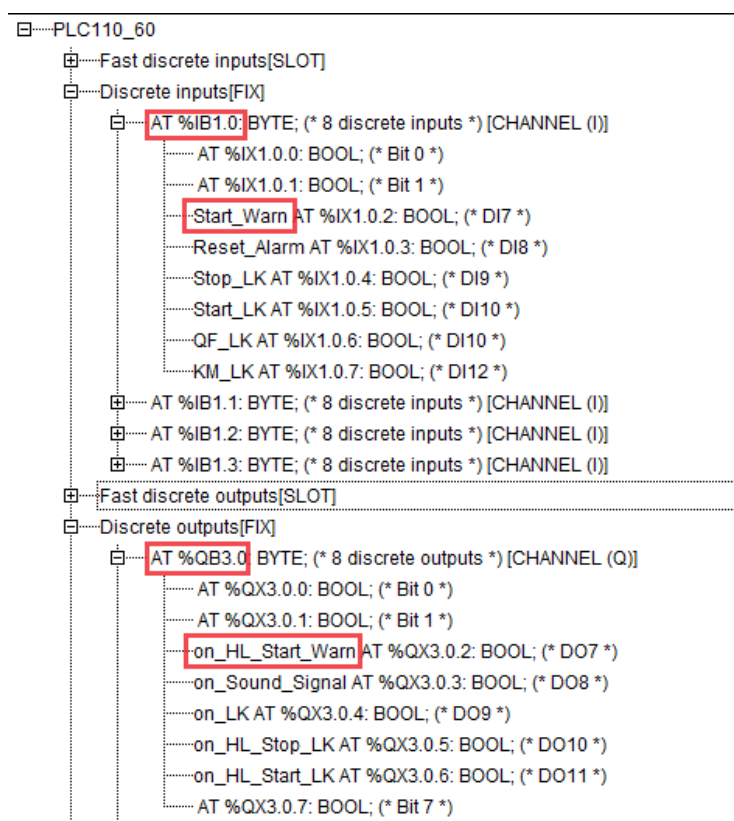


Рис. 17. Введение переменных *Start_Warn* и *on_HL_Start_Warn* в конфигурацию контроллера

```

0001 PROGRAM PLC_PRG
0002 VAR
0003   LK: PRIVOD;           (*Блок управления ленточным конвейером*)
0004   NO_Move_LK: BOOL;    (*Нет движения ЛК*)
0005   Move_LK_OK: BOOL;    (*Есть движение ЛК*)
0006   PIT: PRIVOD;        (*Блок управления шлюзовым питателем*)
0007   Move_Pit_OK: BOOL;   (*Есть движение питателя*)
0008   TON_Gotovnost_Pit: TON; (*Таймер задержки включения готовности питателя*)
0009   on_KM_LK: BOOL;      (*Переменная для имитации контактора ЛК*)
0010   on_KM_Pit: BOOL;    (*Переменная для имитации контактора питателя*)
0011   BLINK_Emul_Move_LK: BLINK; (*Блок мигания для эмуляции движения ЛК*)
0012   RTRIG_Move_LK: R_TRIG; (*Триггер переднего фронта для сигнала датчика движения ЛК*)
0013   TOF_No_Move_LK: TOF; (*Задержка отключения сигнала "Есть движение ЛК"*)
0014   BLINK_Emul_Move_Pit: BLINK; (*Блок мигания для эмуляции движения питателя*)
0015   RTRIG_Move_Pit: R_TRIG; (*Триггер переднего фронта для сигнала датчика движения питателя*)
0016   TOF_No_Move_Pit: TOF; (*Задержка отключения сигнала "Есть движение питателя"*)
0017   m_Alarm_LK: BOOL;   (*Метка "Авария ленточного конвейера"*)
0018   m_Alarm_Pit: BOOL;  (*Метка "Авария шлюзового питателя"*)
0019   RTRIG_Start_Warn: R_TRIG; (*Триггер переднего фронта кнопки "Пусковая сигнализация"*)
0020   TP_Start_Warn: TP;   (*Импульс предпусковой сигнализации*)
0021   m_on_Start_Warn: BOOL; (*Сигнал включения предпусковой сигнализации*)
0022   FTRIG_Start_Warn: F_TRIG; (*Триггер заднего фронта включения предпусковой сигнализации*)
0023   m_Start_Warn_Ok: BOOL; (*Метка "Предпусковая сигнализация проведена"*)
0024   FTRIG_on_LK: F_TRIG; (*Триггер отключения ЛК*)
0025   BLINK_HL_Start_Warn: BLINK; (*Мигание лампы предпусковой сигнализации*)
0026 END_VAR

```

Рис. 18. Введение дополнительных локальных переменных для реализации предпусковой сигнализации

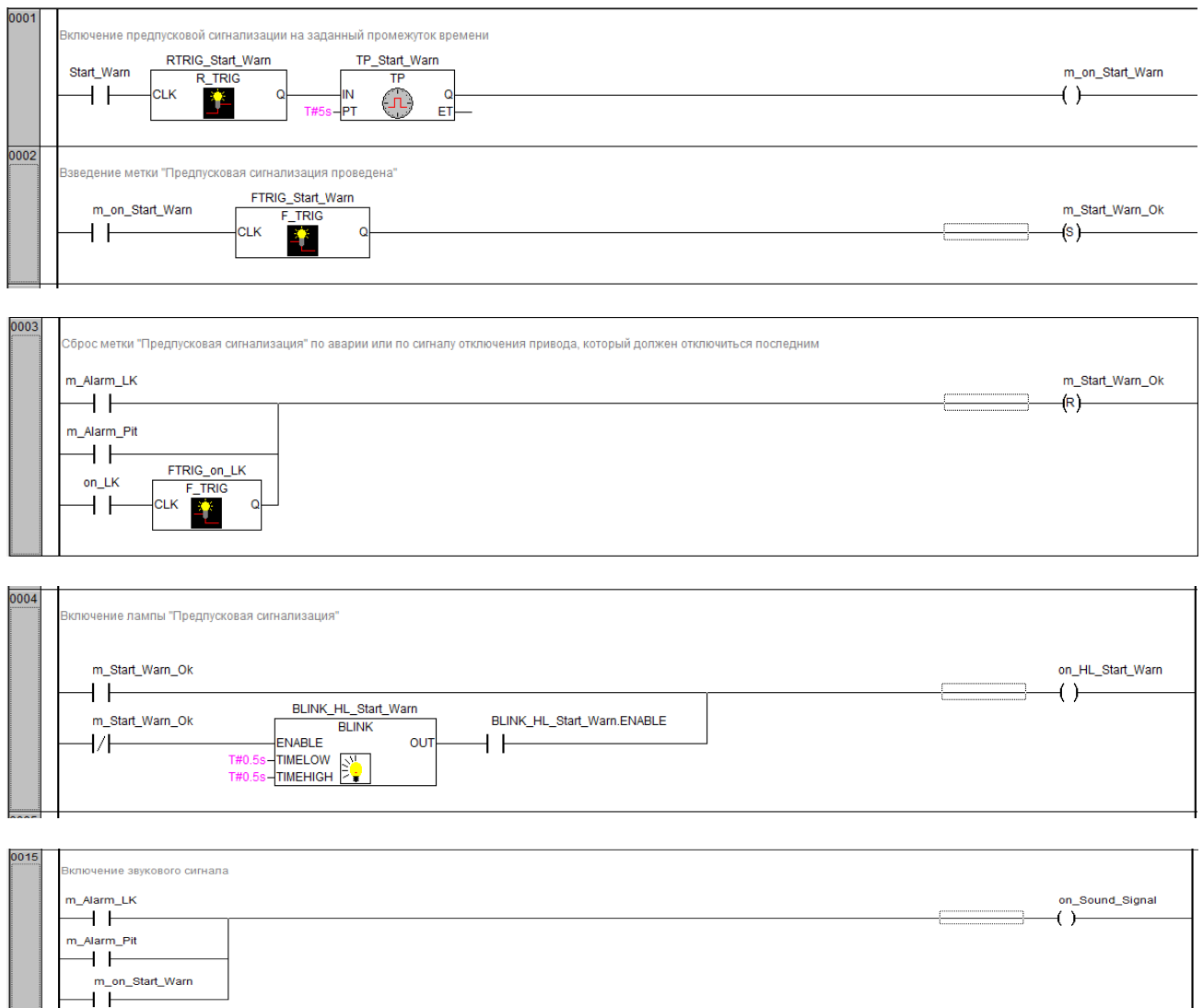


Рис. 19. Реализация предпусковой сигнализации

5. Реализация наладочного режима работы

Наладочный режим работы предусмотрен для отладки и ремонта оборудования с местных постов управления. При этом работы осуществляются без материала и последовательность запуска и останова игнорируется. Включается данный режим работы как правило ключ-биркой, которую ремонтный персонал уносит с собой – чтобы исключить возможность запуска оборудования с поста оператора. Подчеркнем еще раз: в наладочном режиме запуск возможен только с местных постов управления. Предположим, нормально открытый сигнал с ключ-бирки **«Наладка»** подает сигнал на 6 вход ПЛК (назовите переменную **Naladka**).

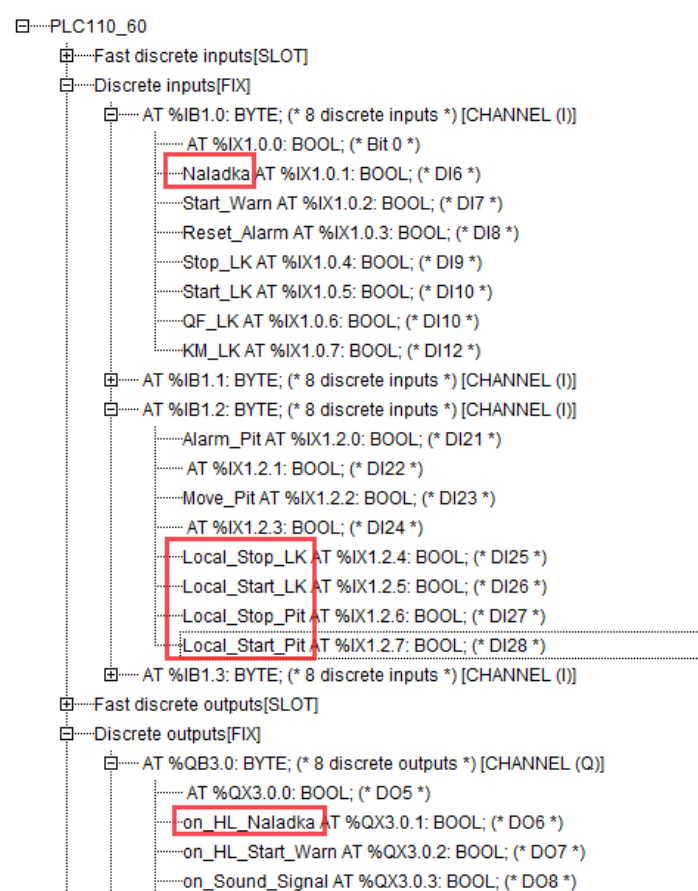


Рис. 20. Введение новых переменных в конфигурацию контроллера
для реализации наладочного режима

Предположим, что местный пост подключен к входам 25-28 и состоит из четырех кнопок (назовите переменные аналогично используемым именам пусковых и стоповых кнопок с префиксом **Local_**). Таким образом, **при переходе в режим «Наладка»** нужно заблокировать сигналы с поста оператора и передать управления на местный пост, а также снять ограничения на последовательности останова и запуска. Если при технической реализации местного поста управления используются кнопки с подсветкой, то сигналы для ламп зачастую берутся параллельно аналогичным сигналам ламп на poste оператора,

поэтому отдельно программировать выходы не будем. Однако для оператора нужно предусмотреть лампу «Наладка», которая будет включаться выходом 6 ПЛК (назовите переменную *on_HL_Naladka*).

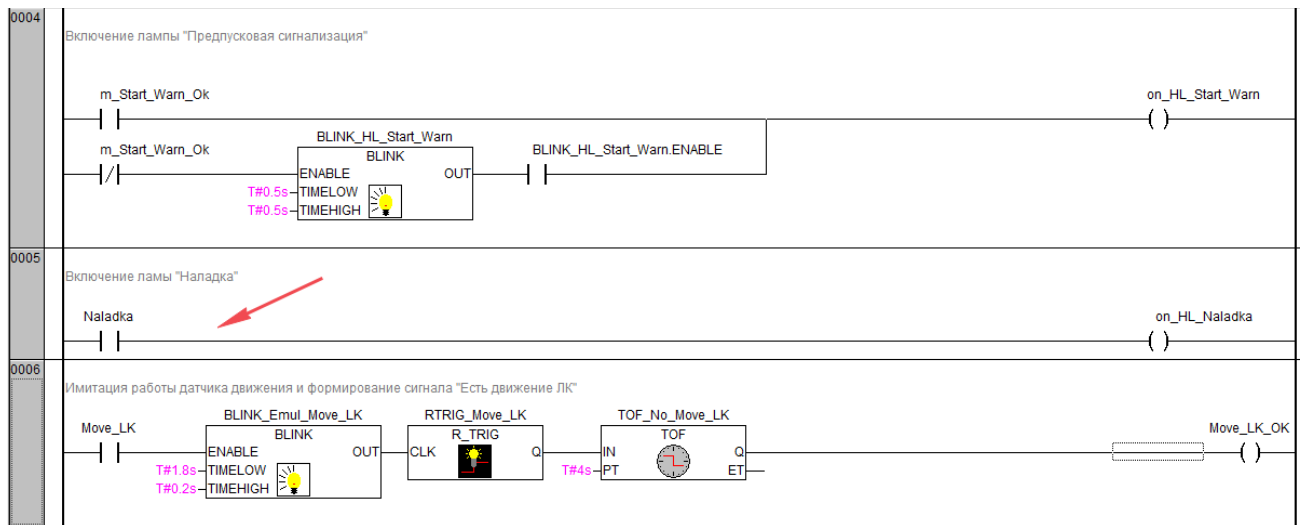


Рис. 21. Включение лампы «Наладка»

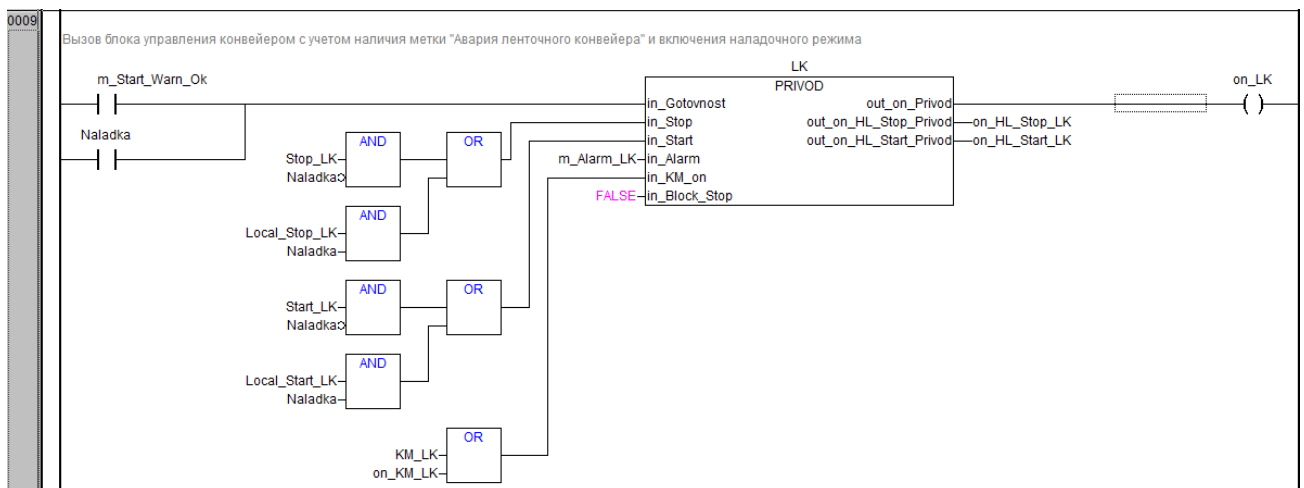


Рис. 22. Вызов блока управления конвейером с учетом наличия метки «Авария ленточного конвейера» и включения наладочного режима (цепи перед ним без изменений)

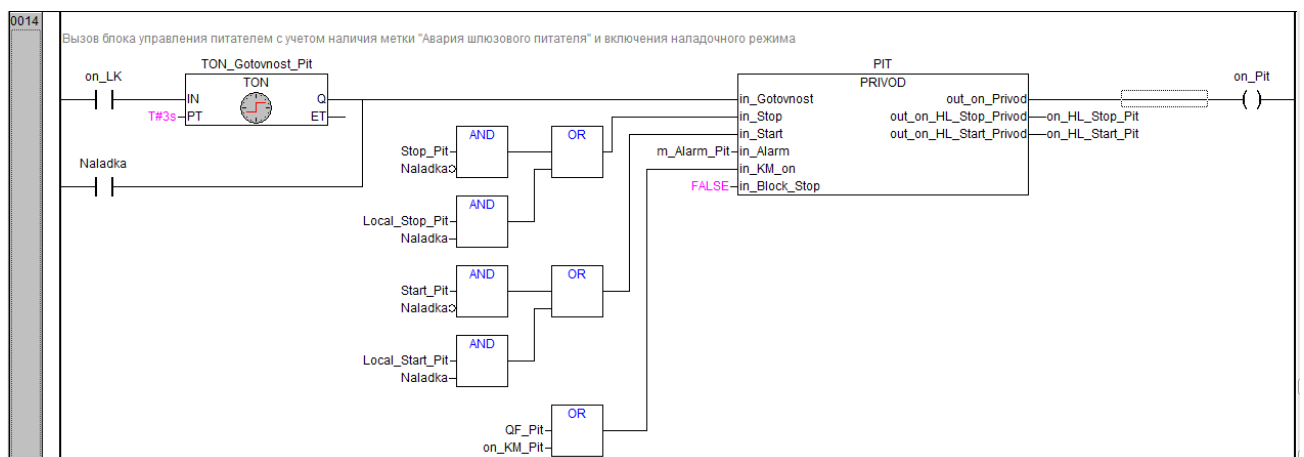


Рис. 23. Вызов блока управления питателем с учетом наличия метки «Авария ленточного конвейера» и включения наладочного режима (цепи перед ним и после без изменений)

6. Реализация автоматического режима работы

Автоматический режим работы предусмотрен для автоматического запуска всего оборудования в соответствии технологией. Включается данный режим работы как правило переключателем или ключ-биркой. Предположим, нормально открытый сигнал с переключателя «*Авто*» подает сигнал на 5 вход ПЛК (назовите переменную *Auto_Start*).

Запуск и останов линии при этом будет осуществляться кнопками «Общий ПУСК» и «Общий «СТОП», которые будут подавать соответствующие сигналы на входы 1 и 2 ПЛК (назовите переменные *Start_All* и *Stop_All*). Таким образом, *при переходе в режим «Авто»* нужно организовать запуск и останов приводов по технологической цепочке. При этом нужно учесть, что ленточный конвейер может быть остановлен только после останова ленточного питателя.

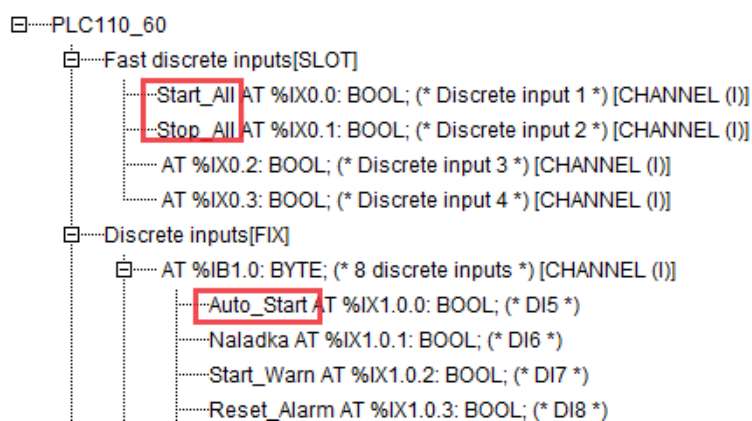


Рис. 24. Введение новых переменных в конфигурацию контроллера
для реализации автоматического режима

0002	VAR
0003	LK: PRIVOD; (*Блок управления ленточным конвейером*)
0004	NO_Move_LK: BOOL; (*Нет движения ЛК*)
0005	Move_LK_OK: BOOL; (*Есть движение ЛК*)
0006	PIT: PRIVOD; (*Блок управления шлюзовым питателем*)
0007	Move_Pit_OK: BOOL; (*Есть движение питателя*)
0008	TON_Gotovnost_Pit: TON; (*Таймер задержки включения готовности питателя*)
0009	on_KM_LK: BOOL; (*Переменная для имитации контактора ЛК*)
0010	on_KM_Pit: BOOL; (*Переменная для имитации контактора питателя*)
0011	BLINK_Emul_Move_LK: BLINK; (*Блок мигания для эмуляции движения ЛК*)
0012	RTRIG_Move_LK: R_TRIG; (*Триггер переднего фронта для сигнала датчика движения ЛК*)
0013	TOF_No_Move_LK: TOF; (*Задержка отключения сигнала "Есть движение ЛК"*)
0014	BLINK_Emul_Move_Pit: BLINK; (*Блок мигания для эмуляции движения питателя*)
0015	RTRIG_Move_Pit: R_TRIG; (*Триггер переднего фронта для сигнала датчика движения питателя*)
0016	TOF_No_Move_Pit: TOF; (*Задержка отключения сигнала "Есть движения питателя"*)
0017	m_Alarm_LK: BOOL; (*Метка "Авария ленточного конвейера"*)
0018	m_Alarm_Pit: BOOL; (*Метка "Авария шлюзового питателя"*)
0019	RTRIG_Start_Warn: R_TRIG; (*Триггер переднего фронта кнопки "Пусковая сигнализация"*)
0020	TP_Start_Warn: TP; (*Импульс предупредительной сигнализации*)
0021	m_on_Start_Warn: BOOL; (*Сигнал включения предупредительной сигнализации*)
0022	FTRIG_Start_Warn: F_TRIG; (*Триггер заднего фронта включения предупредительной сигнализации*)
0023	m_Start_Warn_Ok: BOOL; (*Метка "Предупредительная сигнализация проведена"*)
0024	FTRIG_on_LK: F_TRIG; (*Триггер отключения ЛК*)
0025	BLINK_HL_Start_Warn: BLINK; (*Мигание лампы предупредительной сигнализации*)
0026	m_Alarm: BOOL; (*Метка аварии, включается при наличии любой аварии в системе*)
0027	m_Stop_All: BOOL; (*Метка общего останова*)
0028	m_Start_All: BOOL; (*Метка общего пуска*)
0029	FTRIG_Stop_All: F_TRIG; (*Триггер отключения ленточного конвейера*)
0030	TOF_Reset_Stop_All: TOF; (*Задержка отключения сброса метки общего останова*)
0031	TOF_Block_Stop_LK: TOF; (*Задержка отключения ЛК*)
0032	END VAR

Рис. 25. Введение дополнительных локальных переменных для реализации автоматического режима

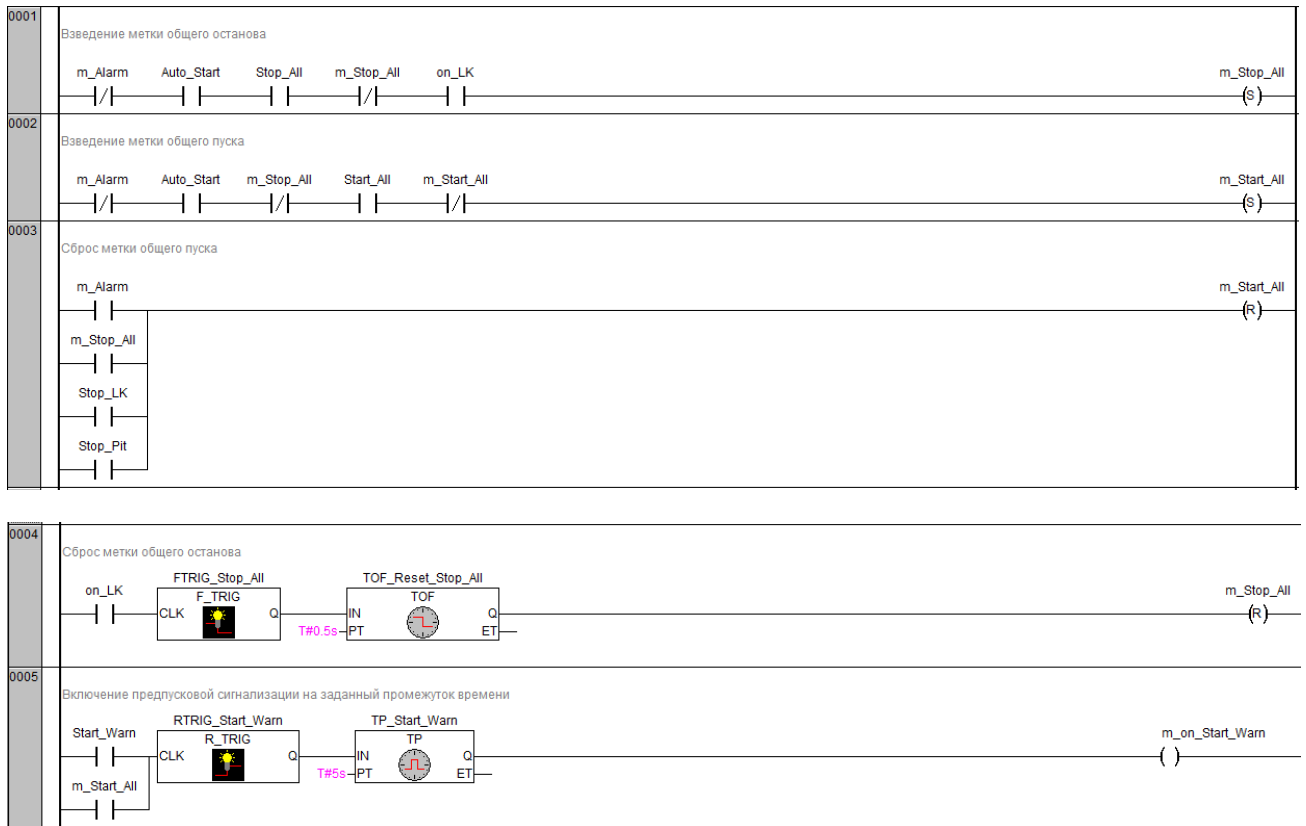


Рис. 26. Введение и сброс меток общего пуска и останова

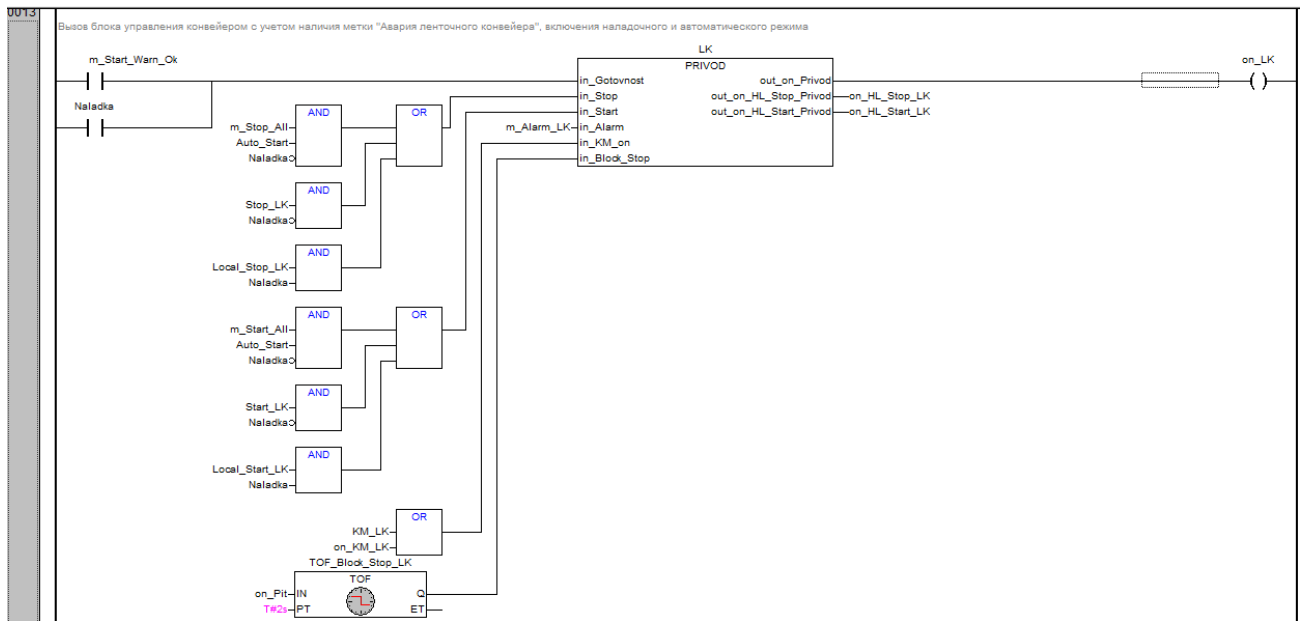


Рис. 27. Вызов блока управления конвейером с учетом наличия метки «Авария ленточного конвейера», включения наладочного и автоматического режима

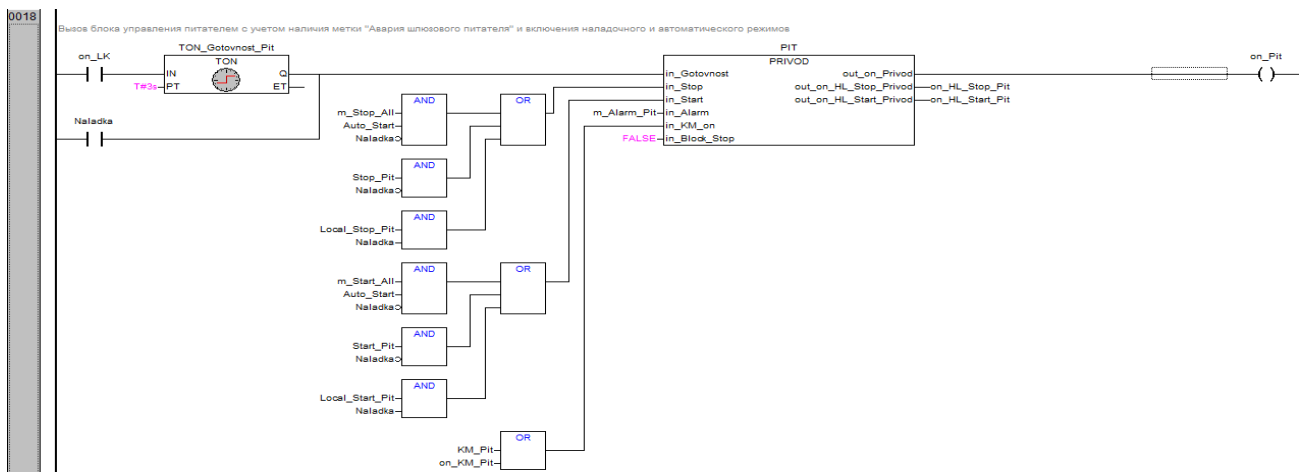


Рис. 28. Вызов блока управления питателем с учетом наличия метки «Авария шлюзового питателя», включения наладочного и автоматического режима

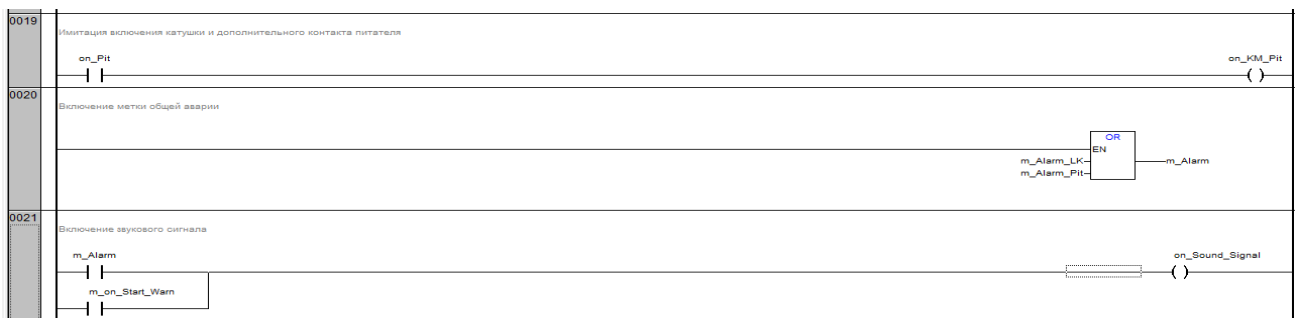


Рис. 29. Включение метки общей аварии и звукового сигнала с ее учетом

На основе конечного проекта в дальнейшем будет построена визуализация, позволяющая управлять технологической линией.

Практическая работа №4

Визуализация системы управления ленточным конвейером в CoDeSys

Внимание: в этой части практикума в качестве основы используется полный проект, выполненный в результате работы над первой частью (ПР №3).

Под термином «визуализация» будем понимать графический человеко-машинный интерфейс, представляющий собой набор условных изображений технологического процесса и его стадий, называемых «мнемосхемами», с отображением значимых технологических параметров и ситуаций, а также средств взаимодействия с автоматизированной системой управления.

На мнемосхемах изображаются (условно) агрегаты технологической линии. Чаще всего создается основная мнемосхема, на которой изображается вся технологическая цепочка и наиболее важные технологические параметры, а также набор мнемосхем для каждого агрегата, на которых можно отследить и задать технические параметры его работы. Также могут быть созданы отдельные мнемосхемы для энергоучета, формирования различного рода отчетности и т.д.

Визуализация на платформе контроллера имеет существенно меньше возможностей по сравнению с визуализацией SCADA-системы. Кроме того, дополнительные возможности контроллера (формирование отчетности, хранение данных и т.д.) на порядок ниже возможностей промышленного компьютера. Однако, для нужд небольших систем управления их может быть вполне достаточно, и тогда визуализация, реализуемая в контроллере может использоваться для отображения и управления с помощью графических панелей оператора.

Основное назначение визуализации – организация рационального и эргономичного взаимодействия оператора и системы управления. Поэтому организация ее объектов должна быть предельно проста и информативна.

Для создания визуализации в CoDeSys нужно перейти на вкладку **Визуализация** в организаторе проекта и в контекстном меню выбрать *Добавить объект*. Создайте визуализацию и назовите ее **PLC_VISU** (это имя визуализации по умолчанию, которая будет запускаться при старте контроллера). В свойствах созданного объекта на вкладке **Визуализация** вы можете указать тип визуализации. **Управляющая панель** – это визуализация, автоматически отображаемая на всех других визуализациях. Это может быть меню, критически важные технологические параметры и т.д. **Визуализация без управляющей панели** – это визуализация, в которой не отображается имеющаяся в проекте управляющая панель. И наконец, **Визуализация** – это стандартный объект визуализации. Для PLC_VISU оставляем это свойство без изменений.

Для вставки элементов в окно визуализации следует воспользоваться панелью элементов, расположенных над окном. Изобразим ленточный конвейер, для чего используем элемент **Прямоугольник со скругленными углами**. При двойном щелчке на созданном объекте (или выборе пункта контекстного меню **Конфигурировать**) появляется

окно **Конфигурирование** элемента. На вкладках этого окна задаются свойства выбранного объекта и привязываются переменные проекта, управляющие и управляемые этим объектом.

На вкладке **Форма** определяется форма фигуры (рис.1).

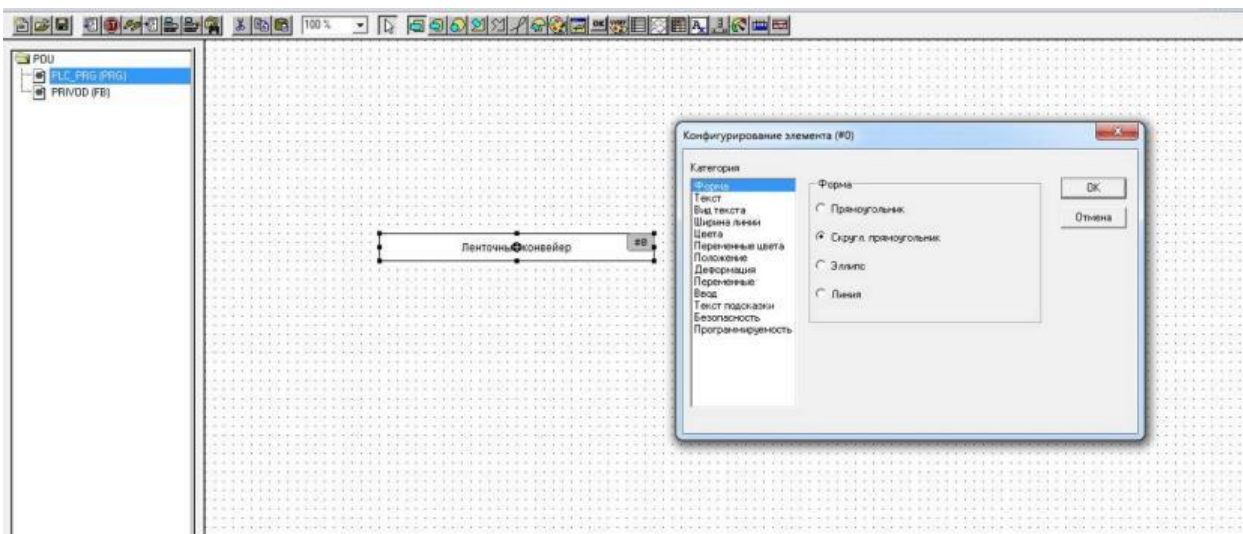


Рис.1. Вкладка «Форма» окна «Конфигурирование элемента»

На вкладке **Текст** задается текст, связанный с элементом, а также параметры этого текста. Для данного элемента нужно написать текст «Ленточный конвейер» (рис.2).

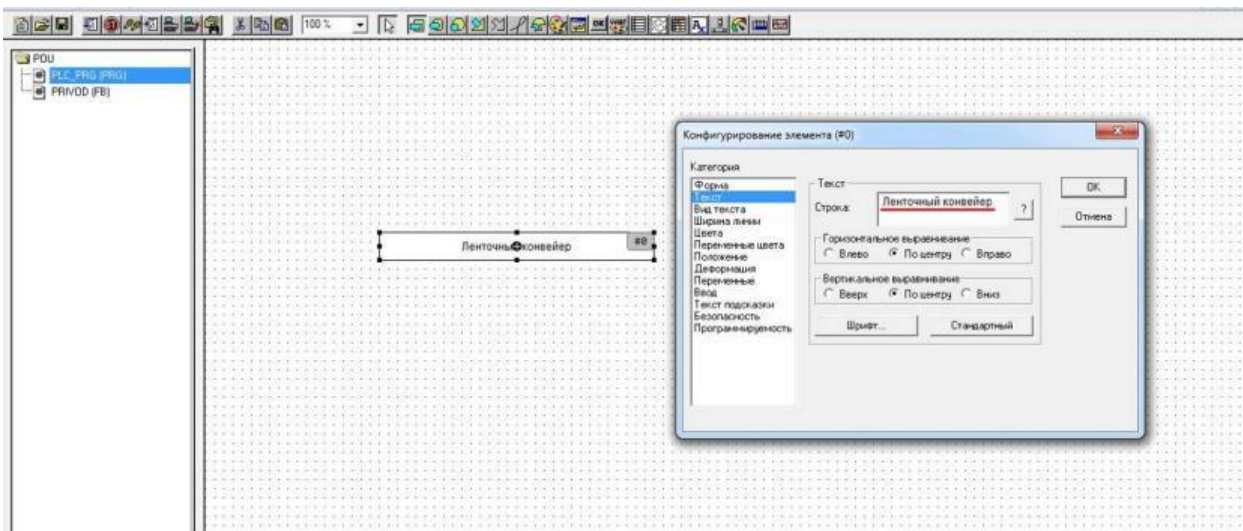


Рис.2. Вкладка «Текст» окна «Конфигурирование элемента»

Далее на вкладке **Цвета** нужно изменить цвет заливки на серый (рис.3). Тревожный цвет для объекта «Ленточный конвейер» сейчас изменять не нужно.

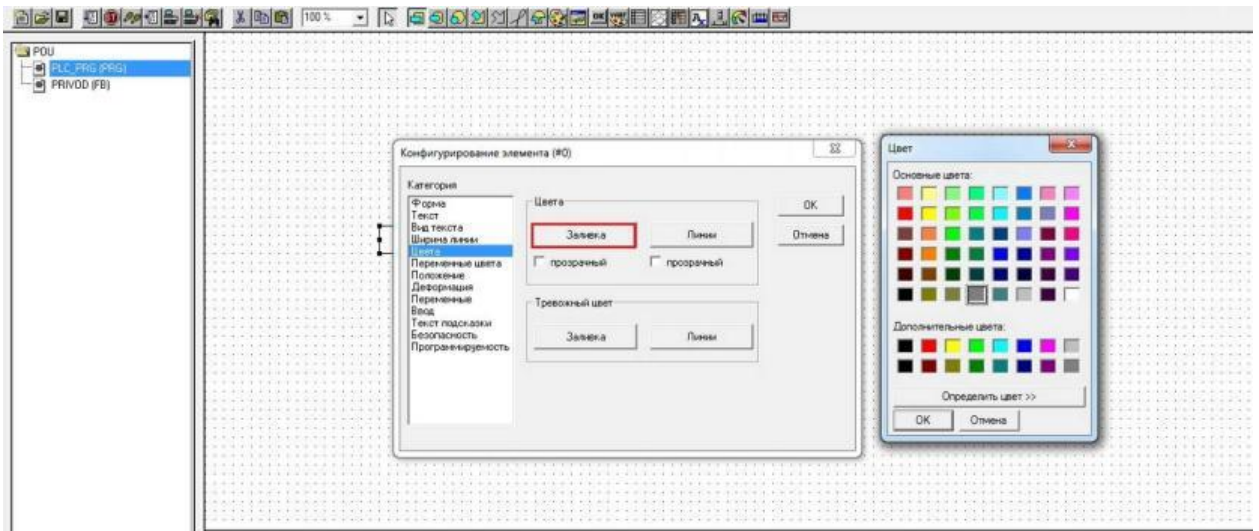


Рис.3. Вкладка «Цвета» окна «Конфигурирование элемента»

Аналогичным образом создайте кнопки «Пуск» и «Стоп» для ленточного конвейера (два круга, один темно зеленый, второй темно красный). Для того чтобы сделать надписи просто добавьте прямоугольники, и на вкладке *Текст* введите соответствующие данные (рис.4).

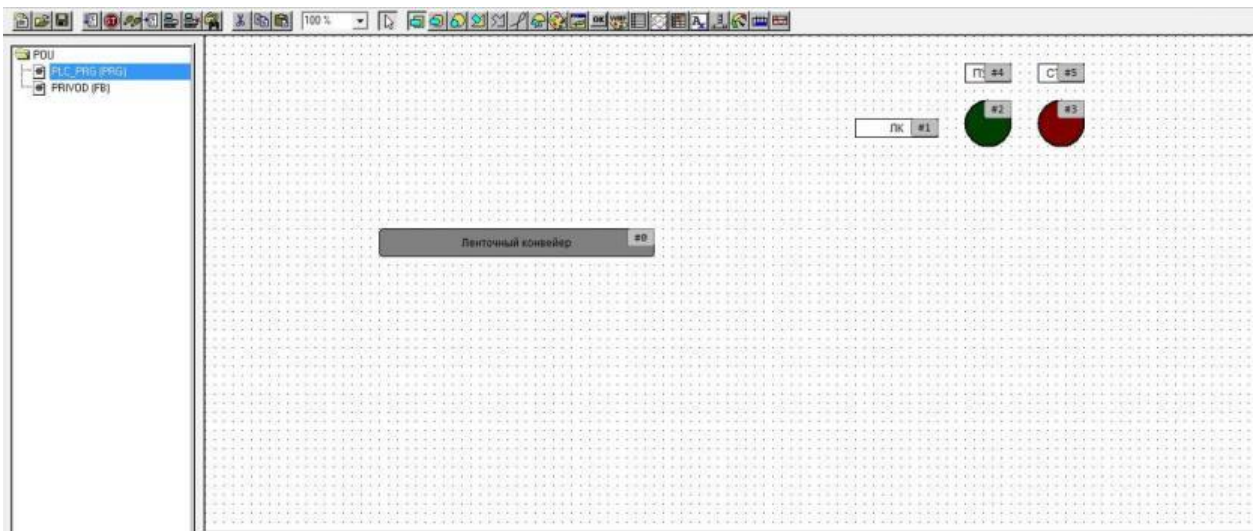


Рис.4. Окно основной визуализации после создания ленточного конвейера и кнопок

Для кнопок измените тревожный цвет. С помощью тревожного цвета можно имитировать лампу, совмещенную с кнопкой. Для кнопки «Пуск» выберите тревожный цвет ярко-зеленым, а для кнопки «Стоп» - ярко-красным.

Для того, чтобы лампы загорались, нужно на вкладке Переменные окна Конфигурирование элемента соответствующей лампы/кнопки в поле *Изменение цвета* ввести имя переменной из конфигурации ПЛК, которая отвечает за включение аналогичного выхода ПЛК (рис.5).

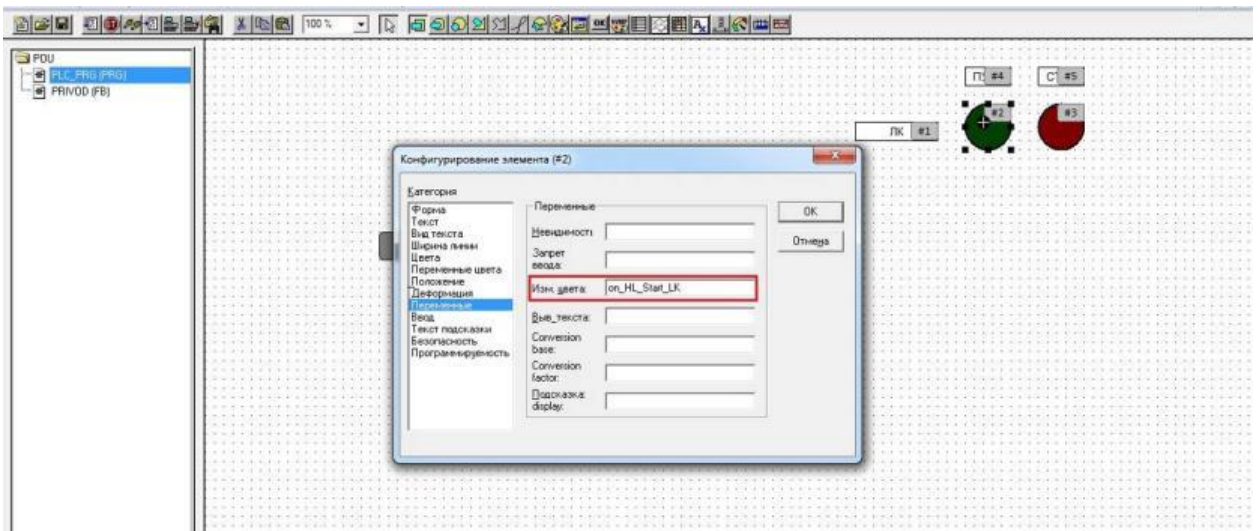


Рис.5. Задание переменной для изменения цвета кнопки/лампы

Задайте переменные для кнопок/ламп, скомпилируйте проект и проверьте корректность работы визуализации. Если лампы не горят, обратитесь к преподавателю.

Теперь нужно добавить функционал кнопок. Допустим, что визуализация нужна нам для отладочных целей и кнопки визуализации включены параллельно кнопкам управления оператора (сигналам с входов ПЛК). Объявим глобальные переменные viz_Start_LK и viz_Stop_LK в отдельно созданной вкладке глобальных переменных (рис.6).



Рис.6. Объявление глобальных переменных визуализации

После того, как глобальные переменные объявлены, нужно осуществить их изменение при нажатии на соответствующую кнопку. Для этого в окне **Конфигурирование элемента** на вкладке **Ввод** нужно поставить галочку на пункте *Переменная кнопка* и в текстовом поле ввести имя соответствующей глобальной переменной (рис.7).

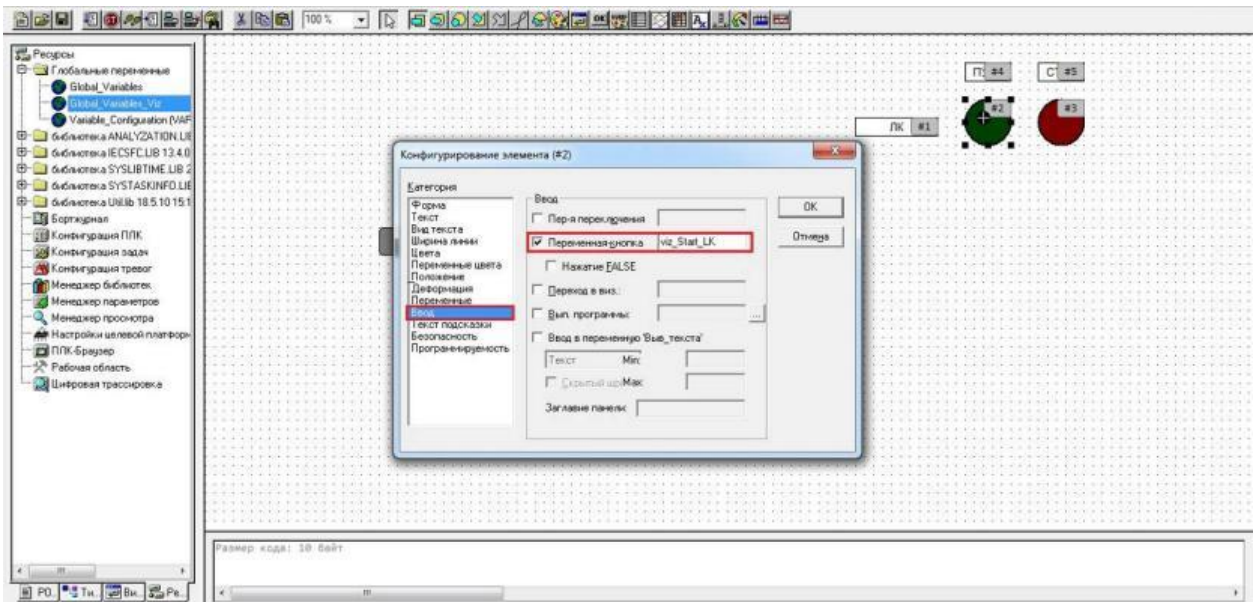


Рис.7. Выбор переменной для ввода при нажатии на кнопку

Теперь при нажатии на кнопку в режиме онлайн, указанная нами переменная будет изменять значение с нуля на единицу. Осталось включить созданные переменные параллельно с входами контроллера. Для этого нужно в пусковой и стоповой цепях блока ленточного конвейера в *PLC_PRG* с сигналами запуска и останова с входов ПЛК через блок ИЛИ добавить сигналы из визуализации (рис.8).

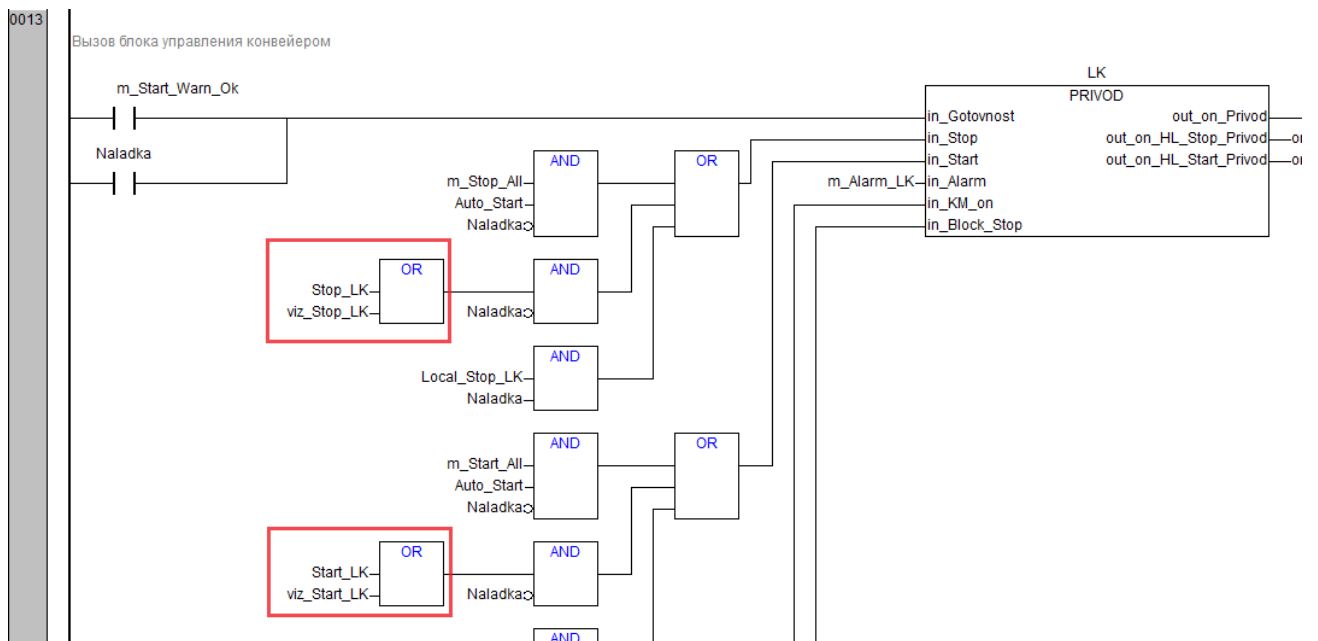


Рис.8. Добавление запуска и останова из визуализации в цепочку пуска и останова ленточного конвейера

Теперь осуществим изменение цвета изображения конвейера таким образом, чтобы он отображал одновременно и сигналы, соответствующие пусковой лампе, и сигналы, соответствующие стоповой лампе. Для этого создадим глобальные переменные viz_Change_Color_LK типа BOOL (переменная для разрешения изменения цвета) и viz_Color_LK типа DWORD (значение цвета), с помощью которых будет реализовано изменение цвета изображения ленточного конвейера (рис.9).



Рис.9. Добавление глобальных переменных для управления цветом изображения ленточного конвейера

Переменную viz_Change_Color_LK нужно указать в поле *Изменение цвета* на вкладке **Переменные** окна **Конфигурирование элемента** изображения ленточного конвейера, а переменную viz_Color_LK в поле *Тревожный* вкладки **Переменные цвета** (рис.10 и 11).

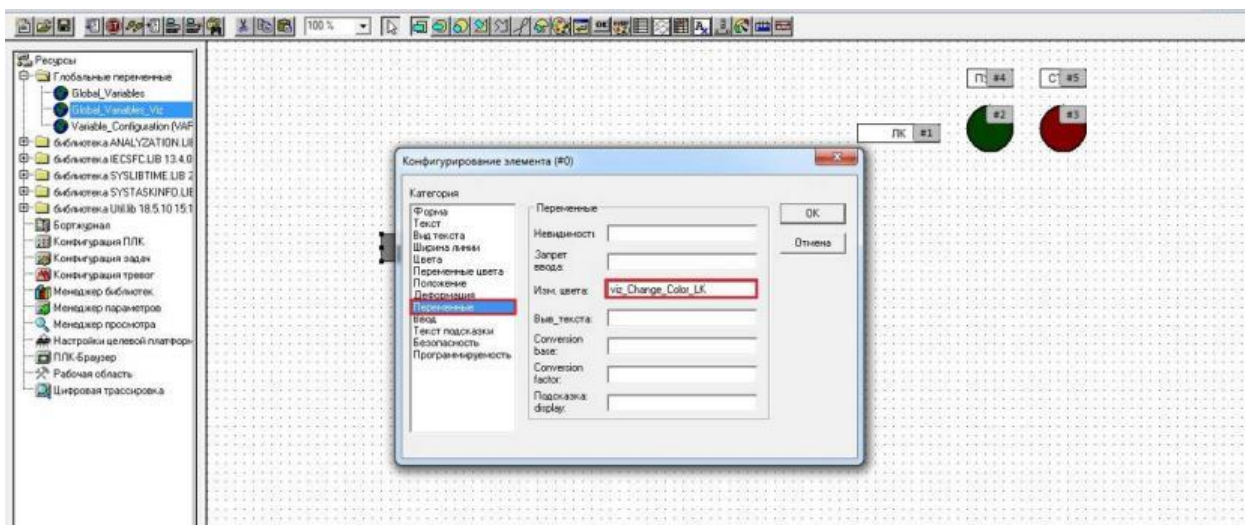


Рис.10. Указание переменной viz_Change_Color_LK

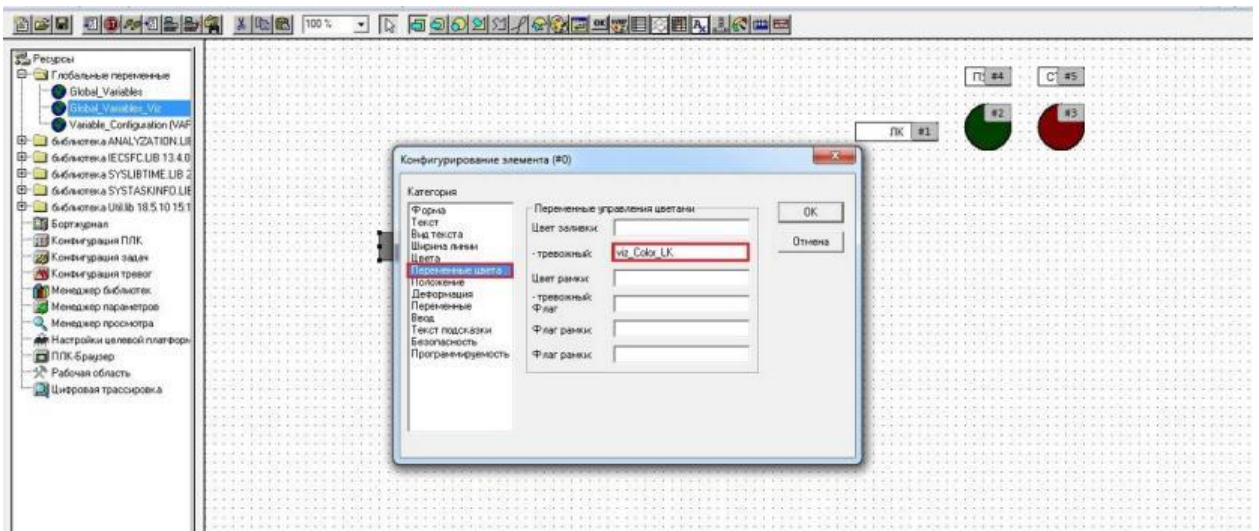


Рис.11. Указание переменной viz_Color_LK

Теперь нужно создать функциональный блок на языке ST для задания значений созданных переменных (назовите его **Change_Color_Privod**). Логика его работы довольно проста: на вход принимаются сигналы включения сигнальных ламп; если есть сигнал хотя бы одной из ламп, то выдается выходной сигнал «Разрешение на изменение цвета», а значение принимает цвет в соответствии с имеющимся сигналом (для стартовой лампы – зеленый, для стоповой – красный). Объявление переменных и код функционального блока приведены на рис.12.

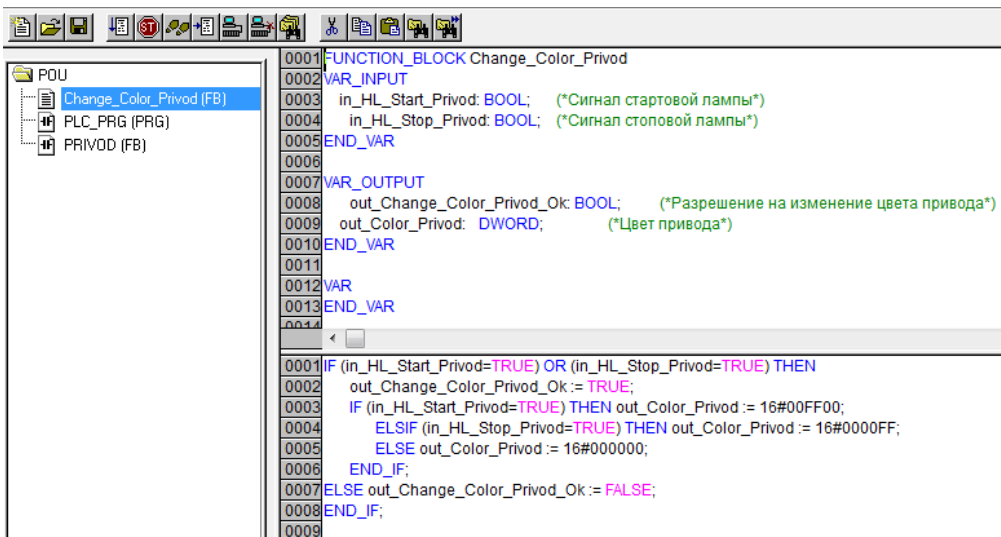


Рис.12. Функциональный блок изменения цвета привода

Чтобы изменять цвет изображения ленточного конвейера, нужно создать для него экземпляр блока **Change_Color_Privod** (назовите его **FB_Change_Color_LK**). Вызовем этот блок сразу после вызова блока привода ленточного конвейера (**LK**). На вход блока нужно подать сигналы включения ламп запуска и останова ленточного конвейера, а значения выходов присвоить созданным глобальным переменным (рис.13).

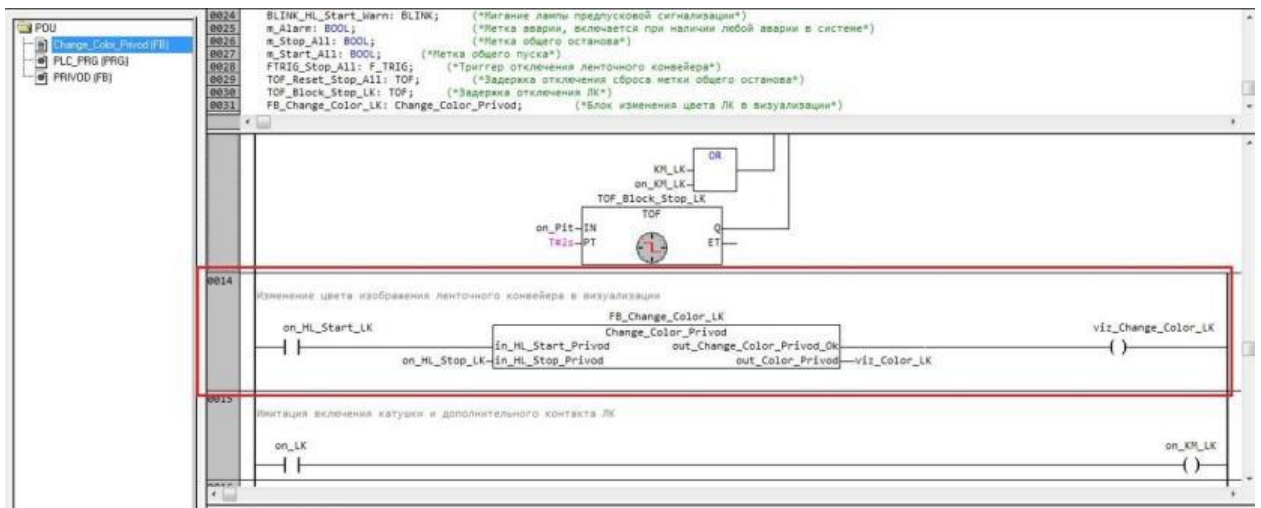


Рис.13. Вызов функционального блока изменения цвета ленточного конвейера

Скомпилируйте проект и запустите его. Проверьте правильность работы визуализации. В случае успешного выполнения переходите к самостоятельной работе.

Самостоятельная работа (общее задание).

По аналогии с ленточным конвейером необходимо создать элементы визуализации для шлюзового питателя (изобразите его многогранником). Результат выполнения задания сверить с преподавателем.

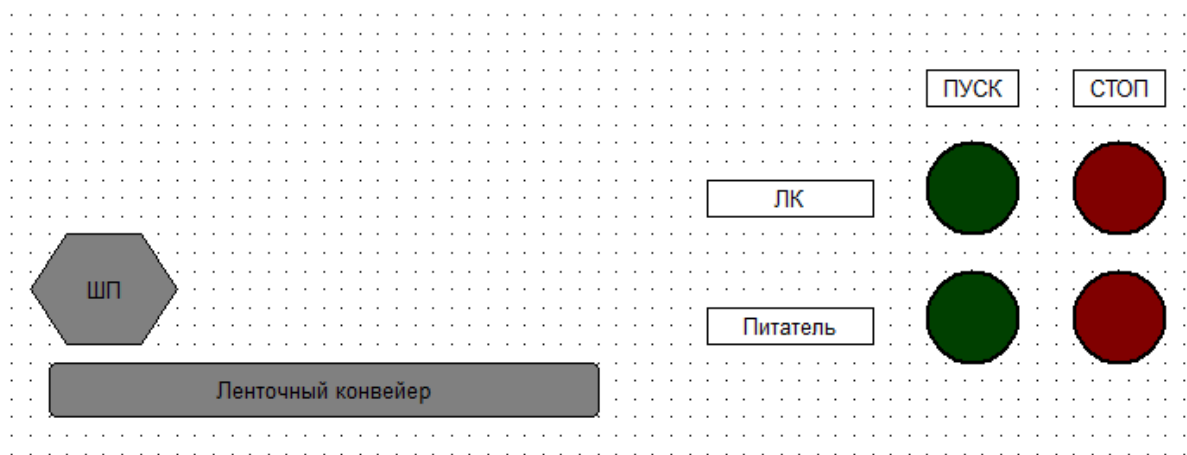


Рис. 14. Добавление элементов для визуализации работы шлюзового питателя

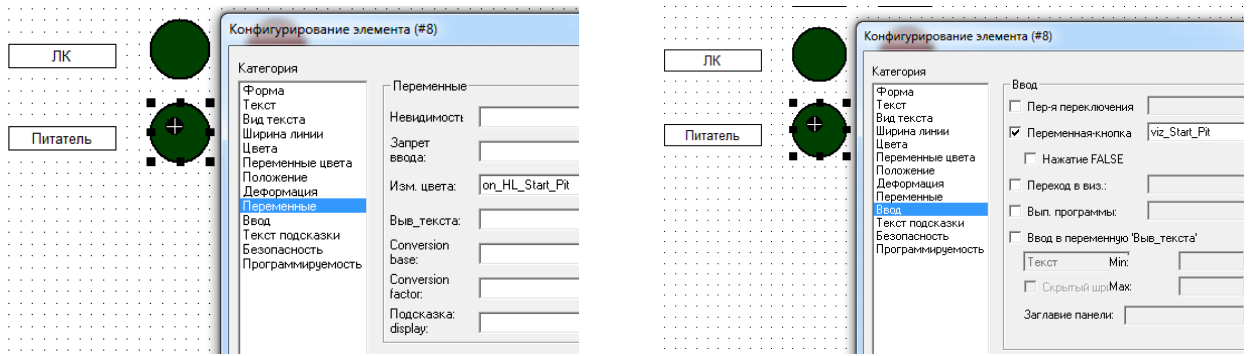


Рис. 15. Задание переменной для изменения цвета кнопки/лампы «Старт» питателя и выбор переменной для ввода при нажатии на кнопку «Старт» питателя

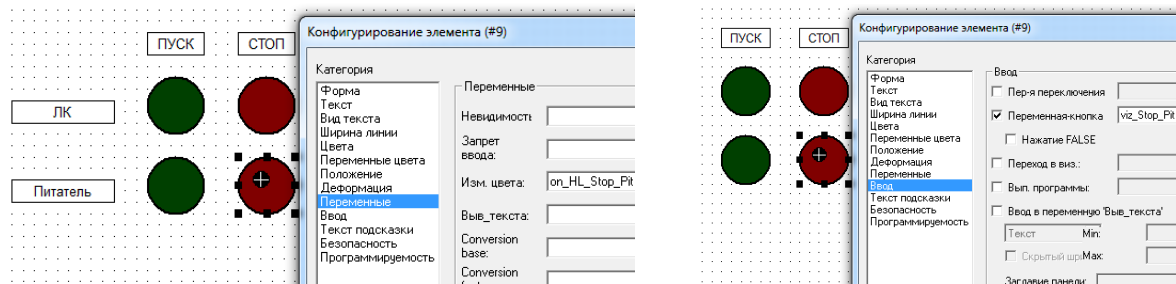


Рис. 16. Задание переменной для изменения цвета кнопки/лампы «Стоп» питателя и выбор переменной для ввода при нажатии на кнопку «Стоп» питателя

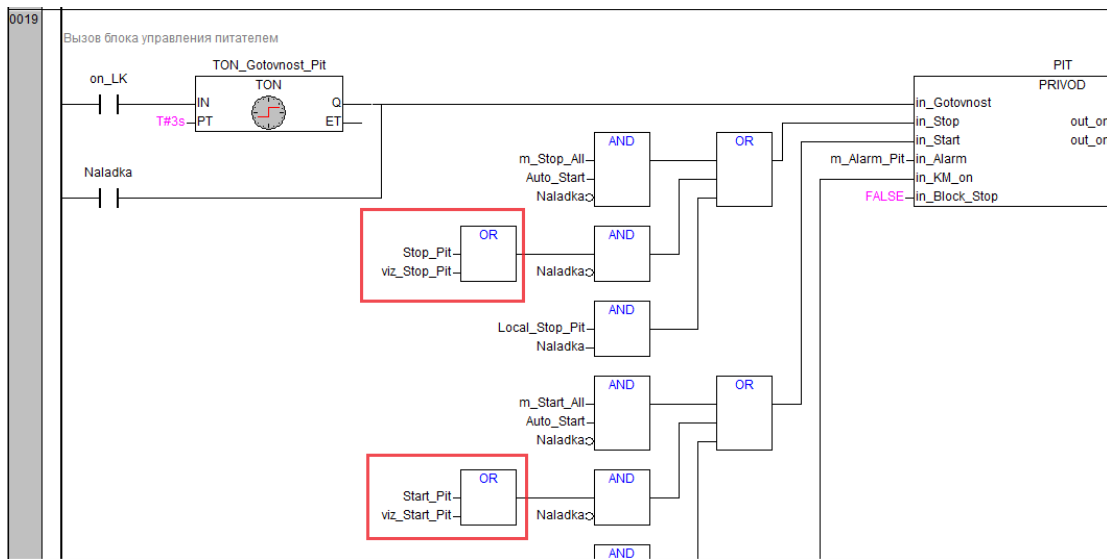


Рис.17. Добавление запуска и останова из визуализации в цепочку пуска и останова шлюзового питателя

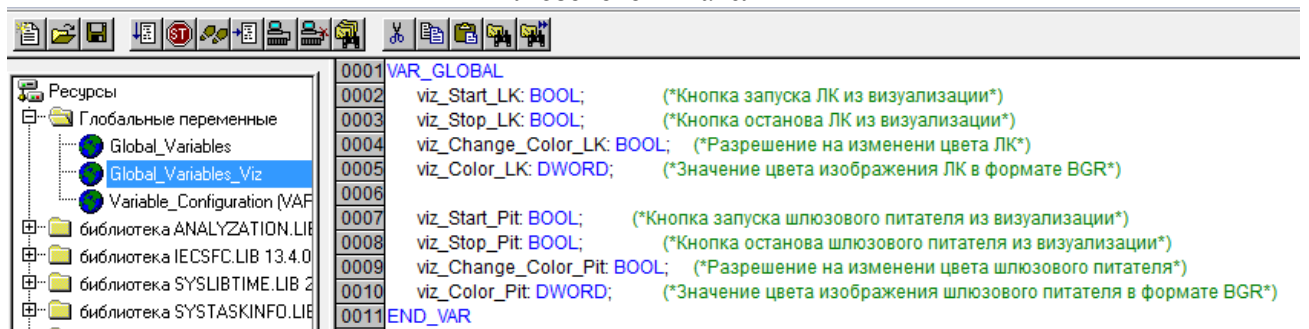


Рис. 18. Добавление глобальных переменных для управления цветом изображения шлюзового питателя

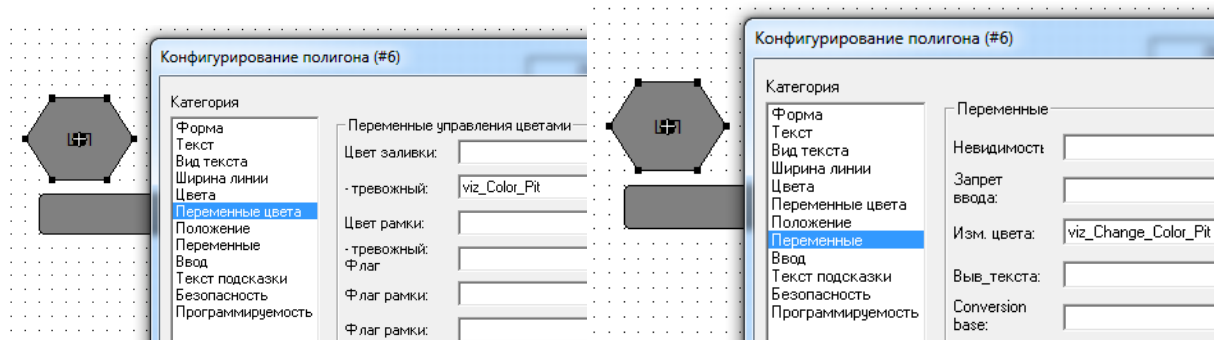


Рис. 19. Привязка глобальных переменных для управления цветом изображения шлюзового питателя

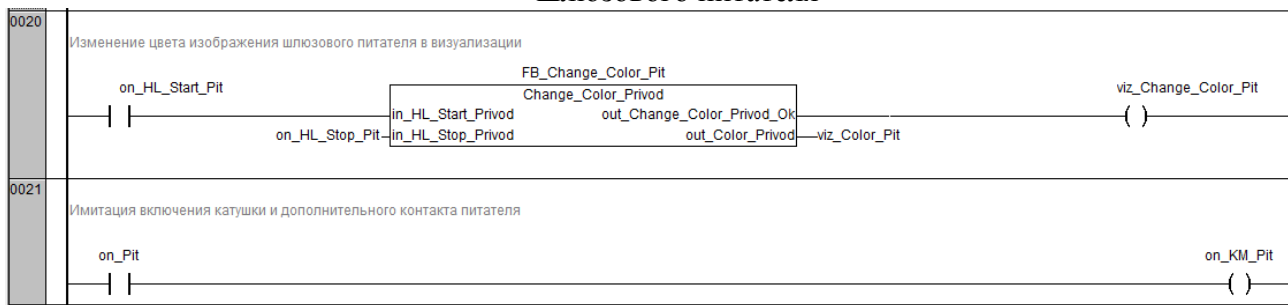


Рис.20. Вызов функционального блока изменения цвета шлюзового питателя

Далее необходимо:

1. Реализовать визуализацию аварийной сигнализации и квитирование аварийной ситуации из визуализации.
2. Реализовать визуализацию предупредительной сигнализации и включение ее из визуализации.
3. Реализовать визуализацию наладочного режима работы (сделать в визуализации местный пост управления и его работу в наладочном режиме).
4. Реализовать визуализацию автоматического режима работы (сделать в визуализации кнопки «Общий Пуск» и «Общий Стоп» и их работу в автоматическом режиме).

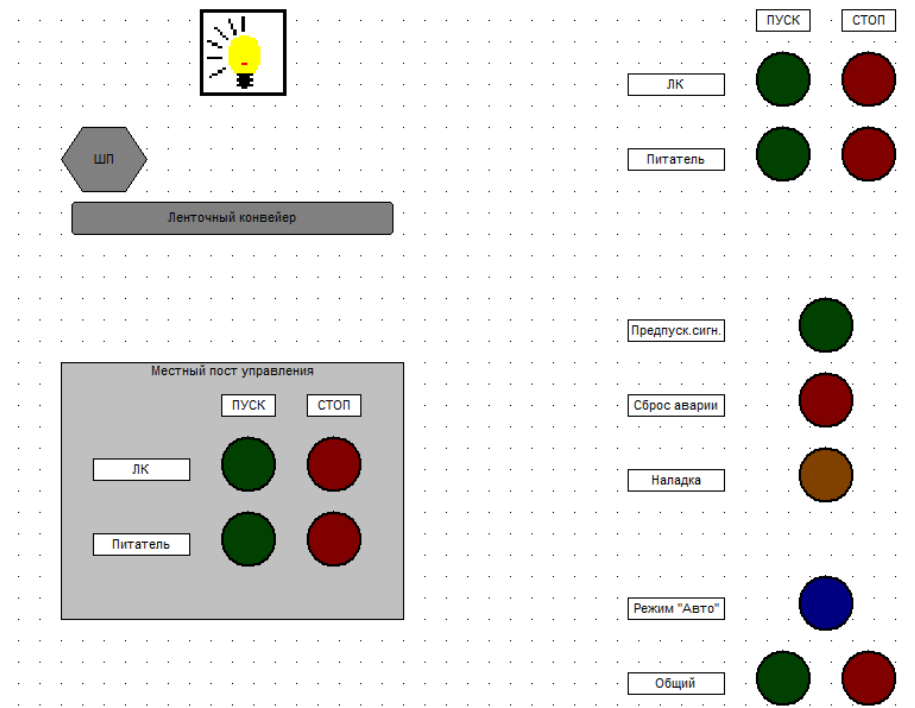


Рис. 21. Добавление элементов для визуализации аварийной и предупредительной сигнализации, наладочного и автоматического режимов работы

```

0001 VAR_GLOBAL
0002 viz_Start_LK: BOOL;      (*Кнопка запуска ЛК из визуализации*)
0003 viz_Stop_LK: BOOL;     (*Кнопка останова ЛК из визуализации*)
0004 viz_Change_Color_LK: BOOL; (*Разрешение на изменени цвета ЛК*)
0005 viz_Color_LK: DWORD;   (*Значение цвета изображения ЛК в формате BGR*)
0006
0007 viz_Start_Pit: BOOL;    (*Кнопка запуска шлюзового питателя из визуализации*)
0008 viz_Stop_Pit: BOOL;    (*Кнопка останова шлюзового питателя из визуализации*)
0009 viz_Change_Color_Pit: BOOL; (*Разрешение на изменени цвета шлюзового питателя*)
0010 viz_Color_Pit: DWORD;  (*Значение цвета изображения шлюзового питателя в формате BGR*)
0011
0012 viz_Reset_Alarm: BOOL;  (*Кнопка сброса аварии в визуализации*)
0013 viz_Start_Warn: BOOL;  (*Кнопка предупредительной сигнализации в визуализации*)
0014
0015 viz_on_Sound_Signal: BOOL; (*Звуковая сигнализация в визуализации*)
0016
0017 viz_Naladka: BOOL;     (*Включение наладочного режима из визуализации*)
0018 viz_Local_Start_LK: BOOL; (*Кнопка местного запуска ЛК из визуализации*)
0019 viz_Local_Stop_LK: BOOL; (*Кнопка местного останова ЛК из визуализации*)
0020 viz_Local_Start_Pit: BOOL; (*Кнопка местного запуска шлюзового питателя из визуализации*)
0021 viz_Local_Stop_Pit: BOOL; (*Кнопка местного останова шлюзового питателя из визуализации*)
0022
0023 viz_Auto_Start: BOOL;   (*Включение автоматического режима из визуализации*)
0024 viz_Start_All: BOOL;    (*Кнопка "Общий Пуск" в визуализации*)
0025 viz_Stop_All: BOOL;    (*Кнопка "Общий Стоп" в визуализации*)
0026 viz_on_HL_Start_All: BOOL; (*Ключение лампы "Общий Старт" в визуализации*)
0027 viz_on_HL_Stop_All: BOOL; (*Ключение лампы "Общий Стоп" в визуализации*)
0028 END_VAR

```

Рис.22. Добавление глобальных переменных для визуализации аварийной и предупредительной сигнализации, наладочного и автоматического режимов работы

0013	BLINK_Emul_Move_Pit: BLINK;	(*Блок мигания для эмуляции движения питателя*)
0014	RTRIG_Move_Pit: R_TRIG;	(*Триггер переднего фронта для сигнала датчика движения питателя*)
0015	TOF_No_Move_Pit: TOF;	(*Задержка отключения сигнала "Есть движение питателя"*)
0016	m_Alarm_LK: BOOL;	(*Метка "Авария ленточного конвейера"*)
0017	m_Alarm_Pit: BOOL;	(*Метка "Авария шлюзового питателя"*)
0018	RTRIG_Start_Warn: R_TRIG;	(*Триггер переднего фронта кнопки "Предусовая сигнализация"*)
0019	TP_Start_Warn: TP;	(*Импульс предпусковой сигнализации*)
0020	m_on_Start_Warn: BOOL;	(*Сигнал включения предпусковой сигнализации*)
0021	FTRIG_Start_Warn: F_TRIG;	(*Триггер заднего фронта включения предпусковой сигнализации*)
0022	m_Start_Warn_Ok: BOOL;	(*Метка "Предпусковая сигнализация проведена"*)
0023	FTRIG_on_LK: F_TRIG;	(*Триггер отключения ЛК*)
0024	BLINK_HL_Start_Warn: BLINK;	(*Мигание лампы предпусковой сигнализации*)
0025	m_Alarm: BOOL;	(*Метка аварии, включается при наличии любой аварии в системе*)
0026	m_Stop_All: BOOL;	(*Метка общего останова*)
0027	m_Start_All: BOOL;	(*Метка общего пуска*)
0028	FTRIG_Stop_All: F_TRIG;	(*Триггер отключения ленточного конвейера*)
0029	TOF_Reset_Stop_All: TOF;	(*Задержка отключения сброса метки общего останова*)
0030	TOF_Block_Stop_LK: TOF;	(*Задержка отключения ЛК*)
0031	FB_Change_Color_LK: Change_Color_Privod;	(*Блок изменения цвета ЛК в визуализации*)
0032	FB_Change_Color_Pit: Change_Color_Privod;	(*Блок изменения цвета питателя в визуализации*)
0033	RTRIG_Naladka: R_TRIG;	(*Триггер включения ключа "Наладка"*)
0034	RTRIG_viz_Naladka: R_TRIG;	(*Триггер включения кнопки "Наладка" в визуализации*)
0035	m_Naladka: BOOL;	(*Метка наладочного режима работы*)
0036	FTRIG_Naladka: F_TRIG;	(*Триггер отключения ключа "Наладка"*)
0037	FTRIG_viz_Naladka: F_TRIG;	(*Триггер отключения кнопки "Наладка" в визуализации*)
0038	m_Auto_Start: BOOL;	(*Метка автоматического режима работы*)
0039	RTRIG_Auto_Start: R_TRIG;	(*Триггер включения ключа "Авто"*)
0040	RTRIG_viz_Auto_Start: R_TRIG;	(*Триггер включения автоматического режима из визуализации*)
0041	FTRIG_Auto_Start: F_TRIG;	(*Триггер отключени ключа "Авто"*)
0042	FTRIG_viz_Auto_Start: F_TRIG;	(*Триггер отключения автоматического режима из визуализации*)
0043	END_VAR	

Рис.23. Добавление локальных переменных для визуализации аварийной и предпусковой сигнализации, наладочного и автоматического режимов работы

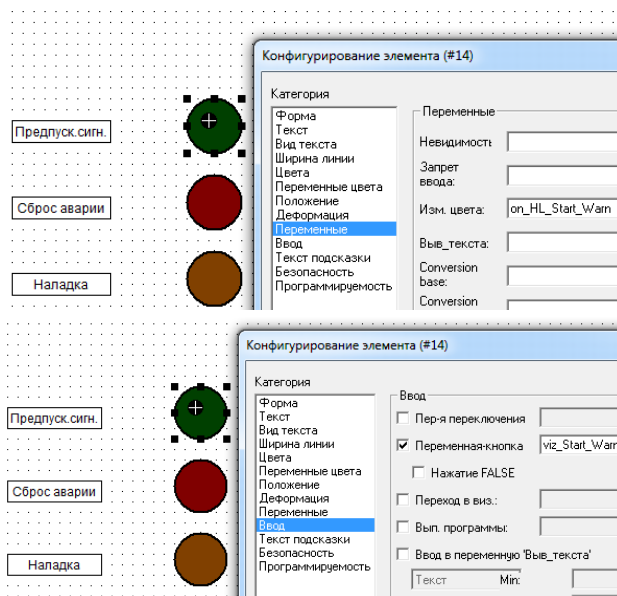


Рис. 24. Задание переменной для изменения цвета кнопки/лампы «Предпусковая сигнализация» и выбор переменной для ввода при нажатии на кнопку «Предпусковая сигнализация»

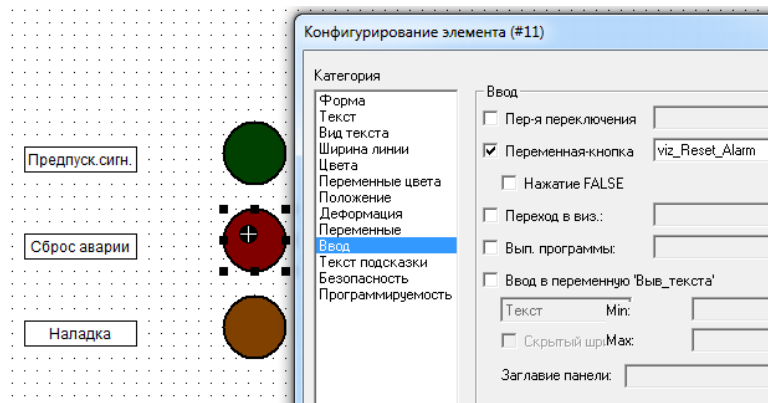


Рис. 25. Выбор переменной для ввода при нажатии на кнопку «Сброс аварии»

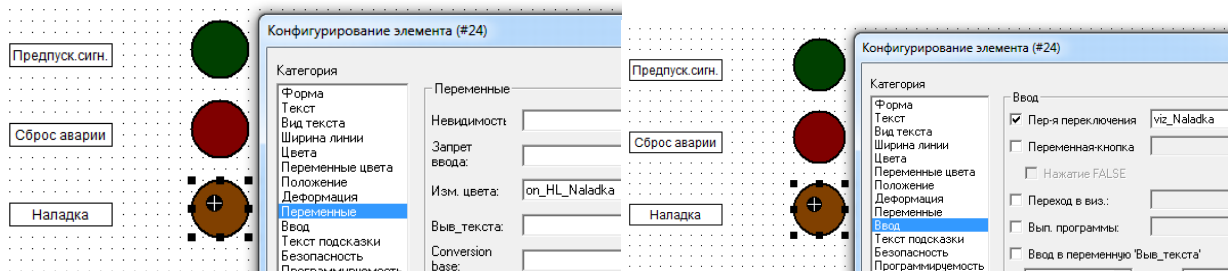


Рис. 26. Задание переменной для изменения цвета кнопки/лампы «Наладка» и выбор переменной переключения для ввода при нажатии на кнопку «Наладка»

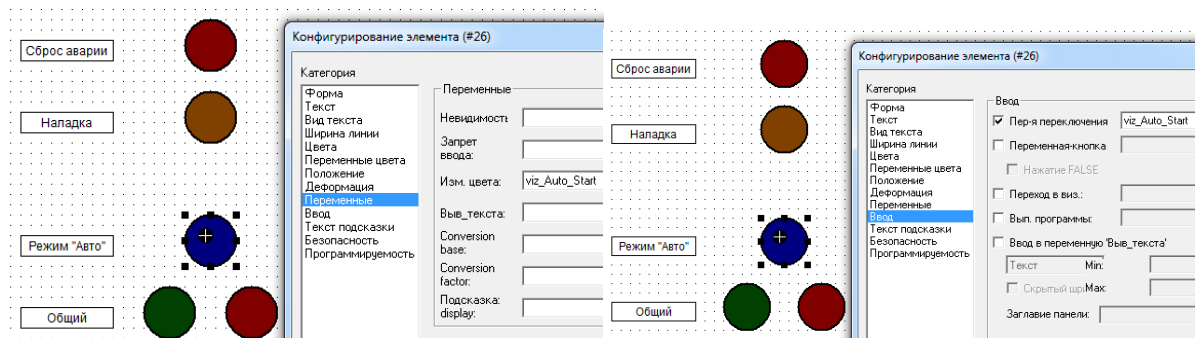


Рис. 27. Задание переменной для изменения цвета кнопки/лампы «Авто» и выбор переменной переключения для ввода при нажатии на кнопку «Авто»

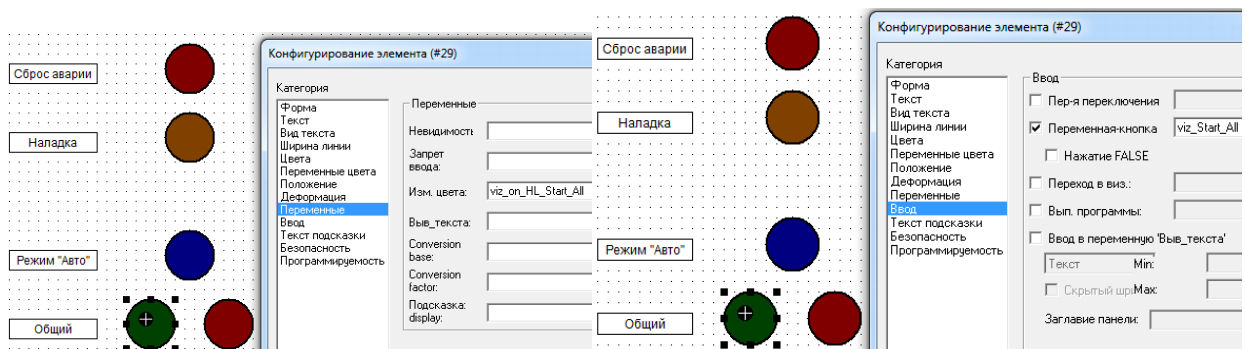


Рис. 28. Задание переменной для изменения цвета кнопки/лампы «Общий ПУСК» и выбор переменной переключения для ввода при нажатии на кнопку «Общий ПУСК»

(в автоматическом режиме)

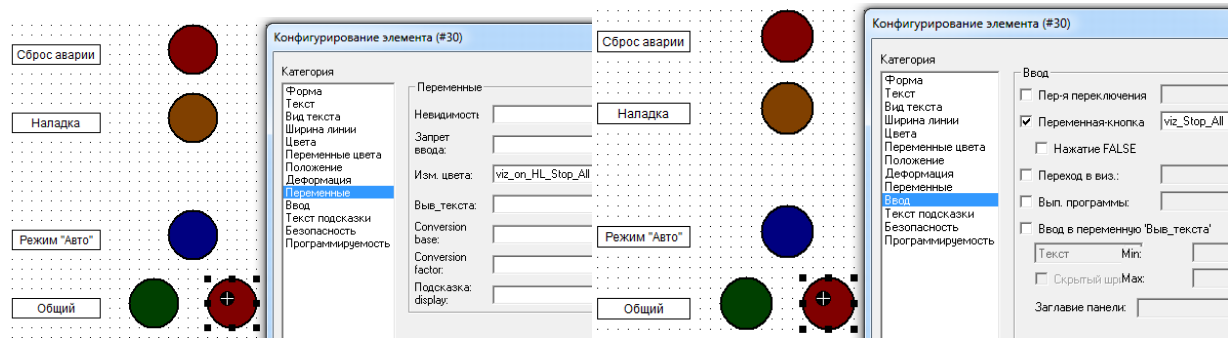


Рис. 29. Задание переменной для изменения цвета кнопки/лампы «Общий СТОП» и выбор переменной переключения для ввода при нажатии на кнопку «Общий СТОП»

(в автоматическом режиме)

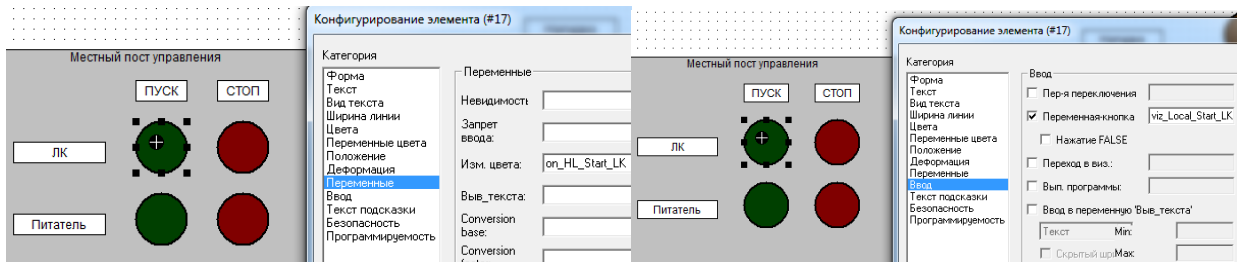


Рис. 30. Задание переменной для изменения цвета кнопки/лампы «Пуск» ленточного конвейера и выбор переменной переключения для ввода при нажатии на кнопку «Пуск» ленточного конвейера (при управлении с местного поста)

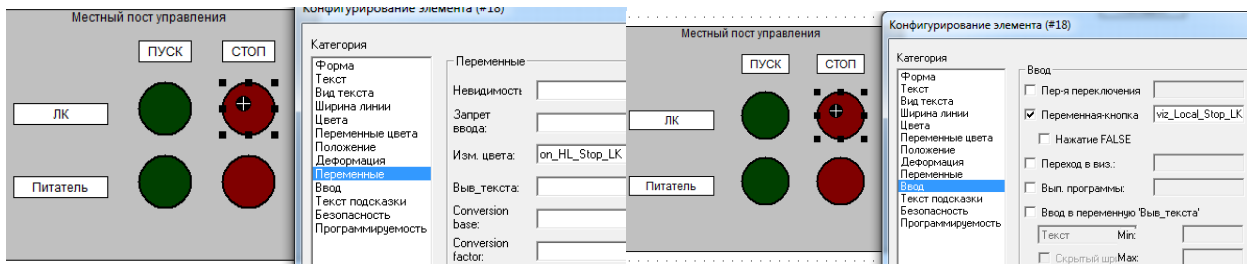


Рис. 31. Задание переменной для изменения цвета кнопки/лампы «Стоп» ленточного конвейера и выбор переменной переключения для ввода при нажатии на кнопку «Стоп» ленточного конвейера (при управлении с местного поста)

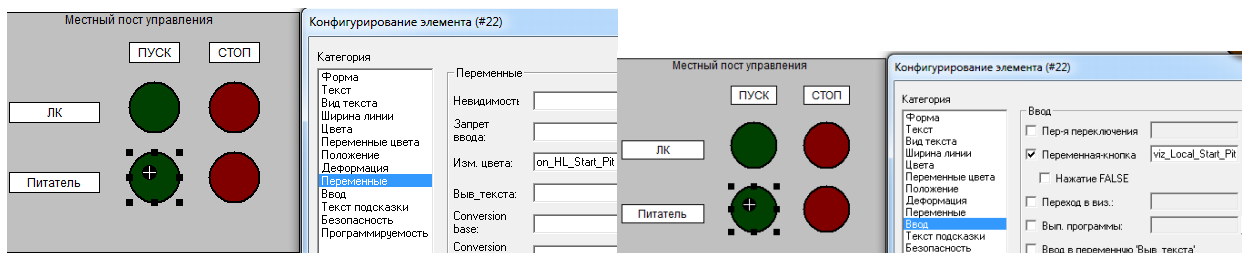


Рис. 32. Задание переменной для изменения цвета кнопки/лампы «Пуск» шлюзового питателя и выбор переменной переключения для ввода при нажатии на кнопку «Пуск» шлюзового питателя (при управлении с местного поста)

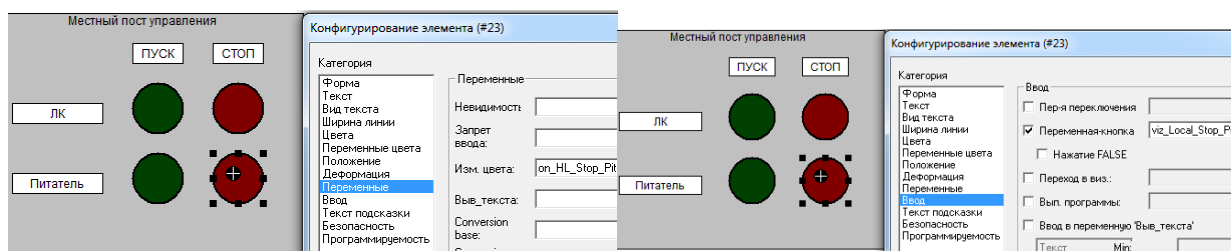


Рис. 33. Задание переменной для изменения цвета кнопки/лампы «Стоп» шлюзового питателя и выбор переменной переключения для ввода при нажатии на кнопку «Стоп» шлюзового питателя (при управлении с местного поста)

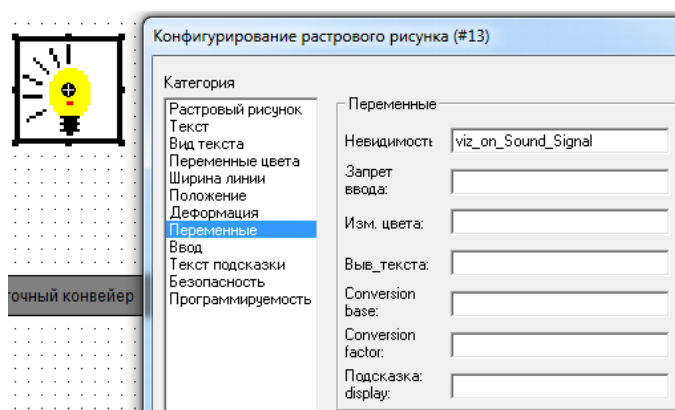
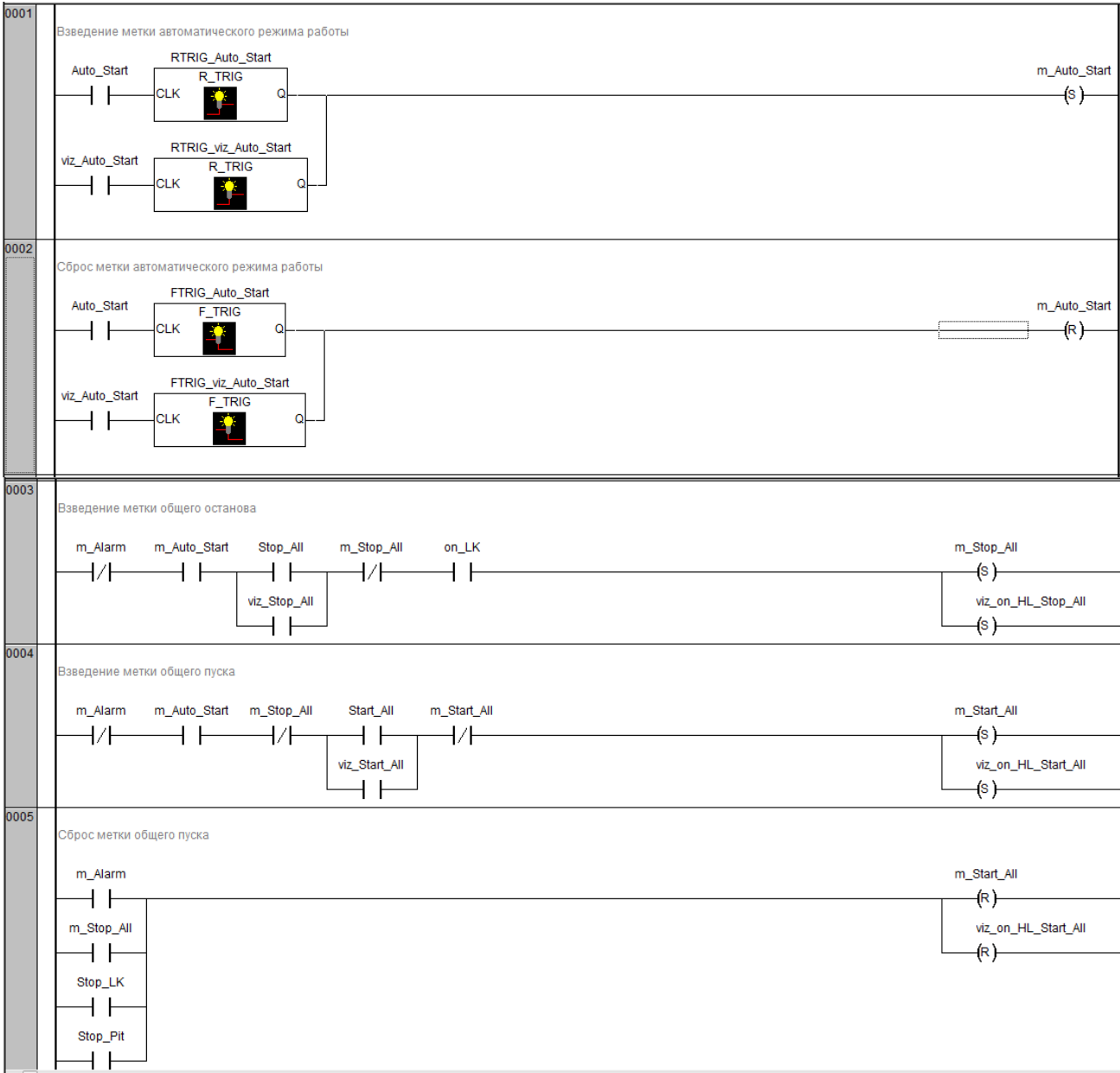
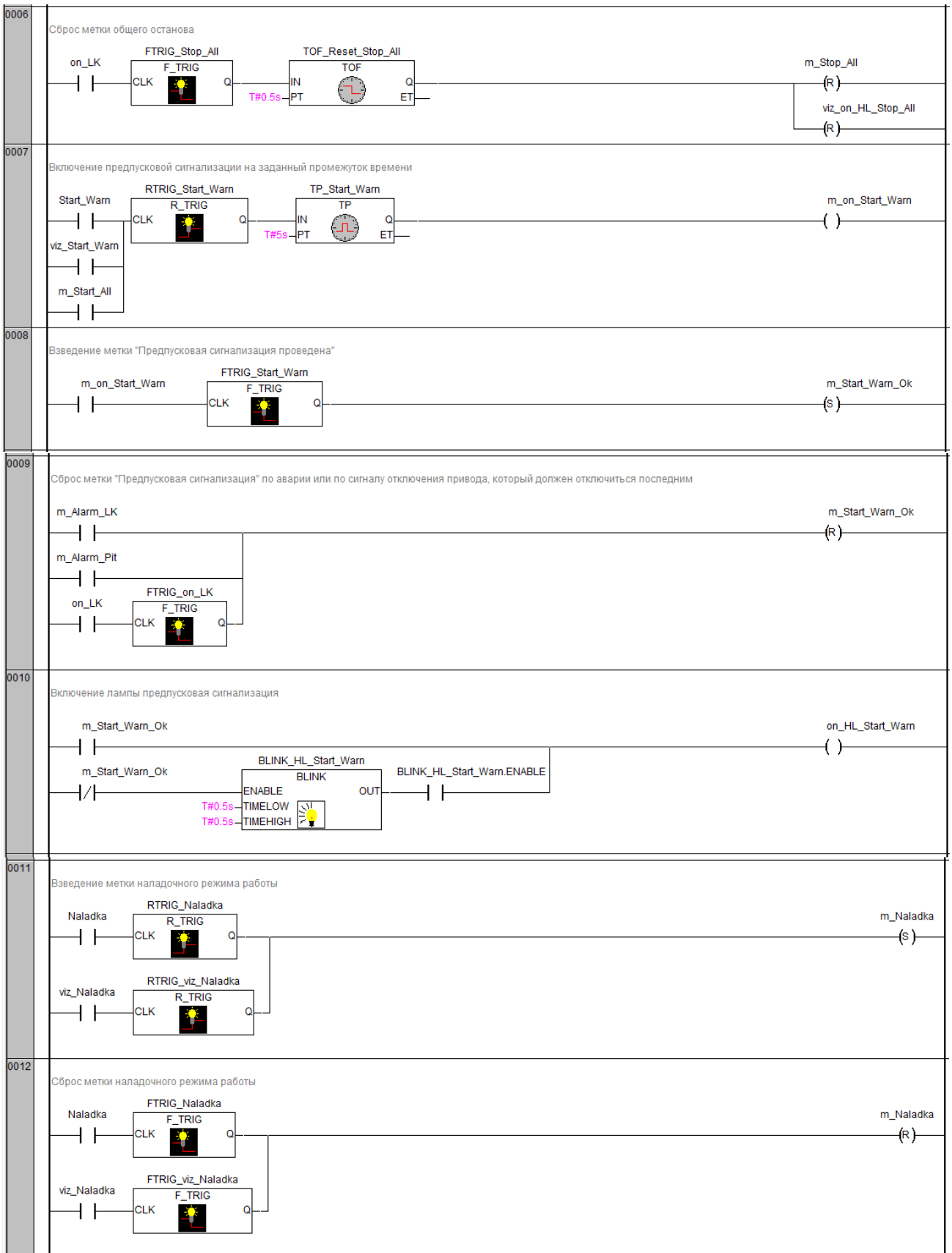
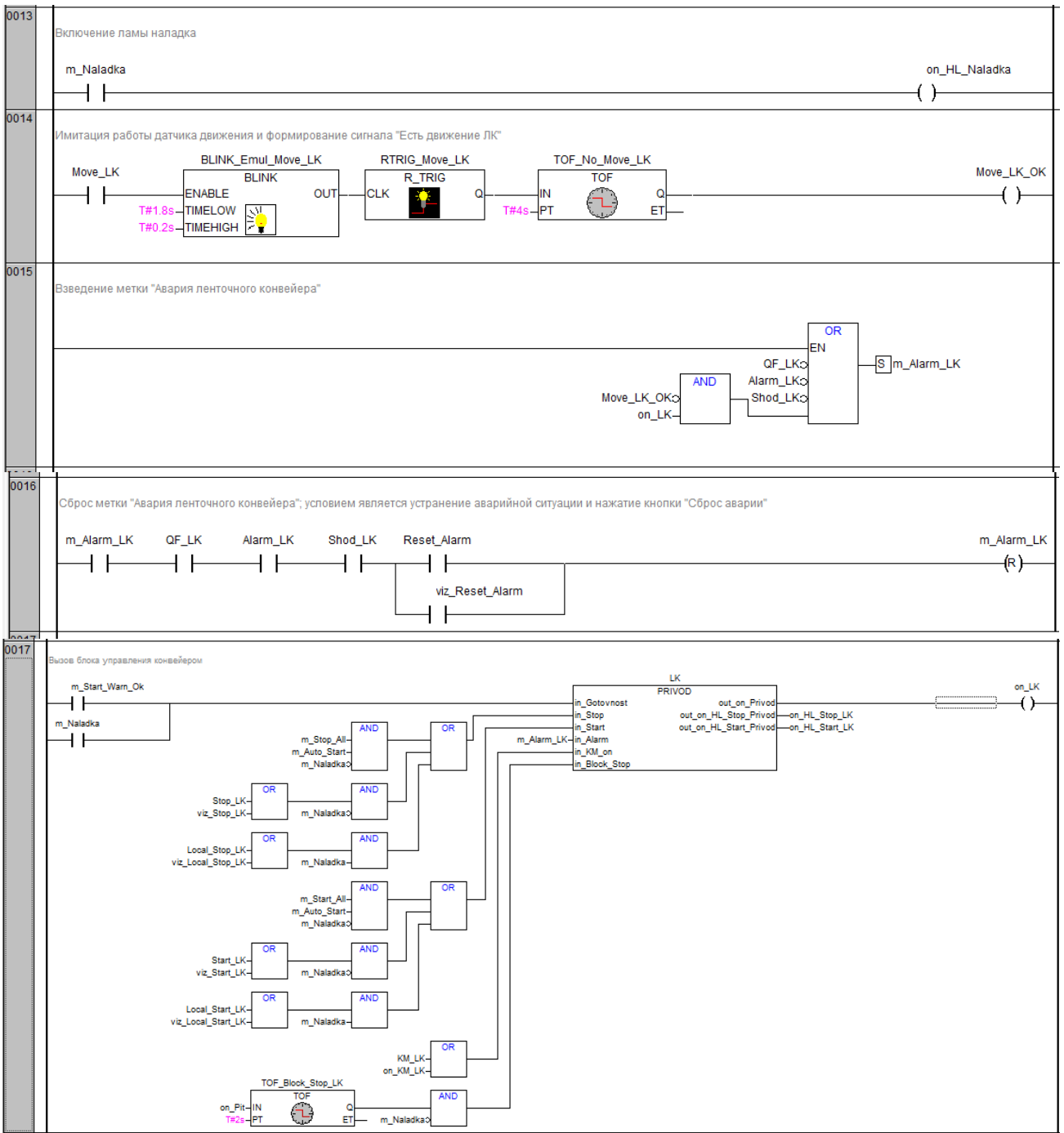
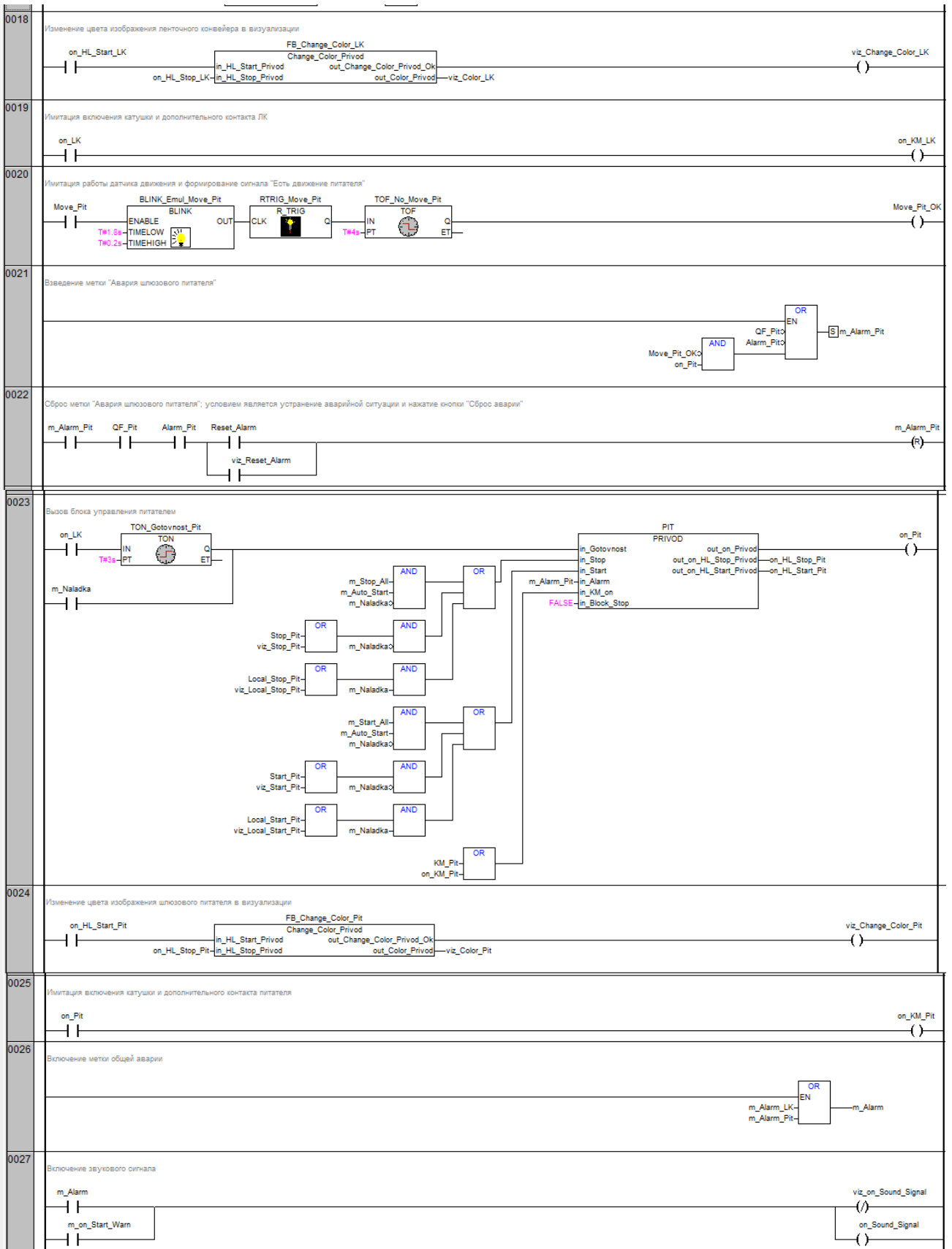


Рис. 34. Задание переменной для изменения видимости сигнальной лампы (имитация звукового сигнала)









ЗАПУСК В РАБОТУ: Онлайн – Подключение – Ресурсы – Конфигурация ПЛК (установить входные сигналы контроллера, см. рис. 35) – Онлайн – Старт – Визуализация.

При правильной работе при нажатии «Старт» должна мигать лампа «Предпусковая сигнализация». При ее нажатии включается звуковой сигнал, через 5 секунд начинает мигать лампа «Пуск» ЛК и сам ЛК, информируя о готовности ЛК к работе. После запуска ЛК через 3 секунды начинает мигать лампа «Пуск» питателя и сам питатель, информируя о готовности питателя к работе. Затем можно нажать кнопку «Пуск» питателя, запуская тем самым всю линию в работу. Если линия работает в ручном режиме, то чтобы ее остановить, нужно сначала остановить питатель (кнопка «Стоп», при этом замигает лампа «Пуск» питателя), и только после этого отключить ЛК. До отключения питателя ЛК отключить не получится.

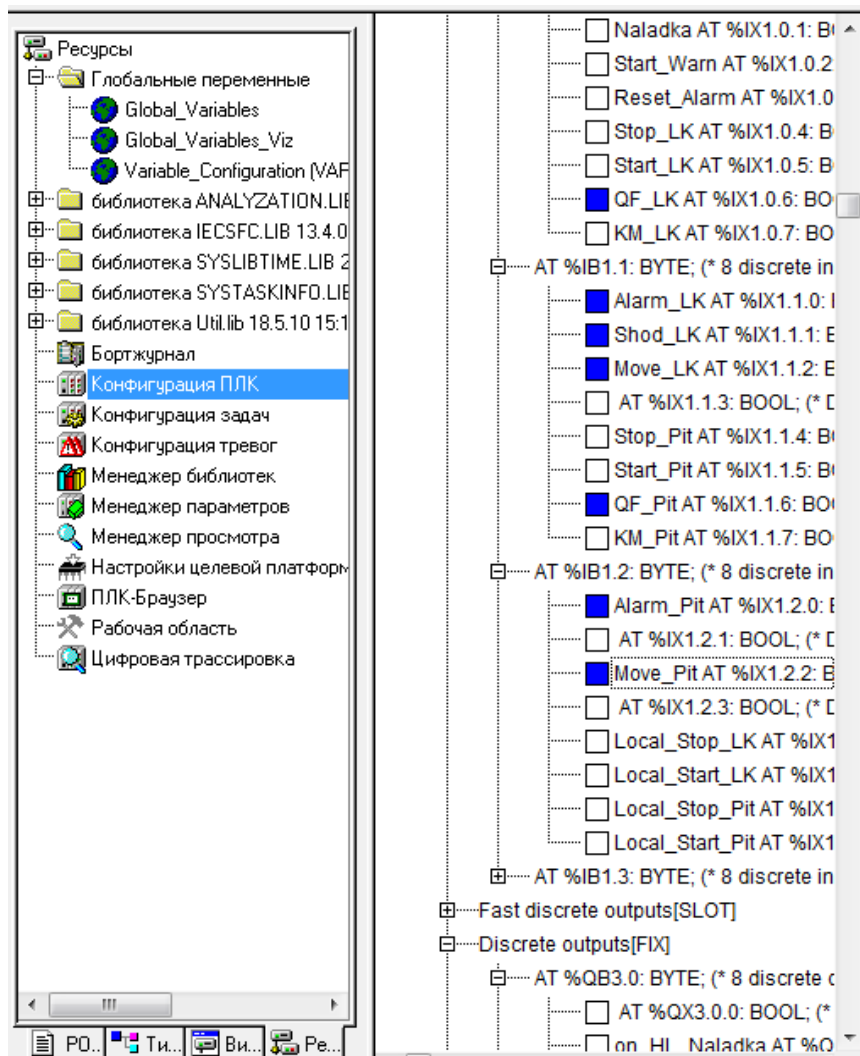


Рис. 35. Установка входных сигналов контроллера

АВАРИЯ - убрать сигнал Shod_LK. При этом ЛК замигает красным цветом, замигает красным цветом кнопка «Стоп» ЛК и будет мигать лампа предпусковой сигнализации, информируя о том, что для последующего запуска после устранения аварии нужно снова нажать кнопку предпусковой сигнализации. Оператор должен устранить аварию (снова установить сигнал Shod_LK) и нажать кнопку «Сброс аварии», подтверждая, что авария устранена.

НАЛАДОЧНЫЙ РЕЖИМ: при нажатии кнопки «Наладка» мигают ЛК, питатель, все пусковые лампы, и предпусковая сигнализация. Запуск агрегатов возможен только с

местного поста управления. Можно поочередно запускать и проверять правильность работы ЛК и питателя. Когда наладка завершена, нужно снова нажать кнопку «Наладка», при этом агрегаты останавливаются, и снова загорается лампа предупредительной сигнализации.

АВТОМАТИЧЕСКИЙ РЕЖИМ: запускаем переключатель «Авто» и даем общий «Пуск», при этом линия начинает автоматически работать в соответствии с технологией: подается звуковой сигнал, запускается ЛК, запускается питатель. При нажатии общего «Стоп» агрегаты останавливаются, и затем загорается предупредительная сигнализация.

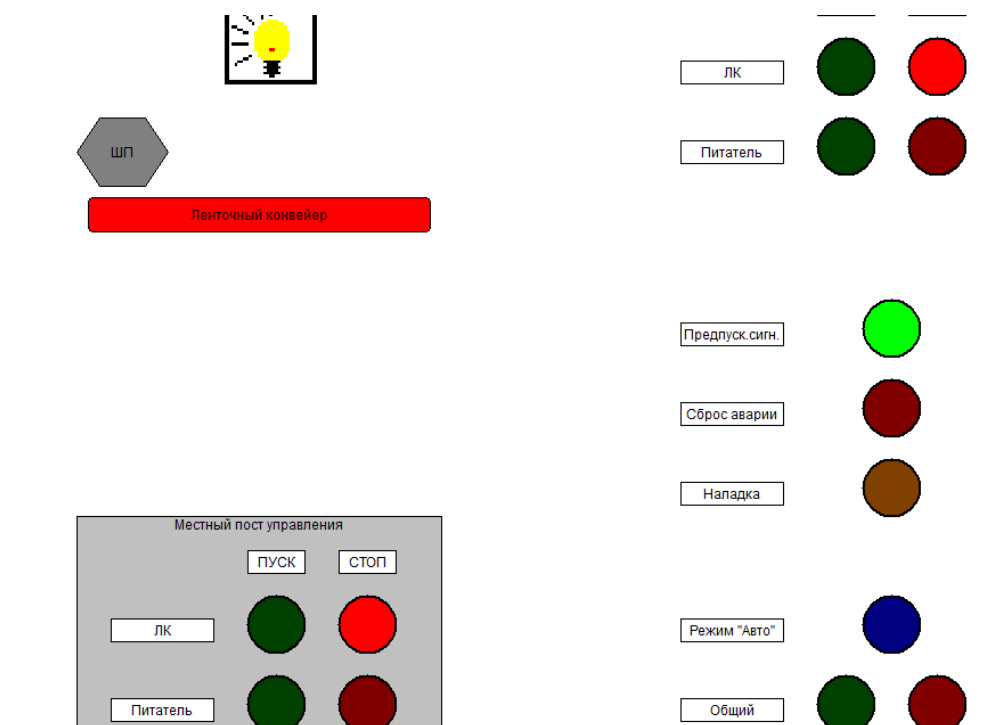


Рис. 36. Визуализация аварийной ситуации

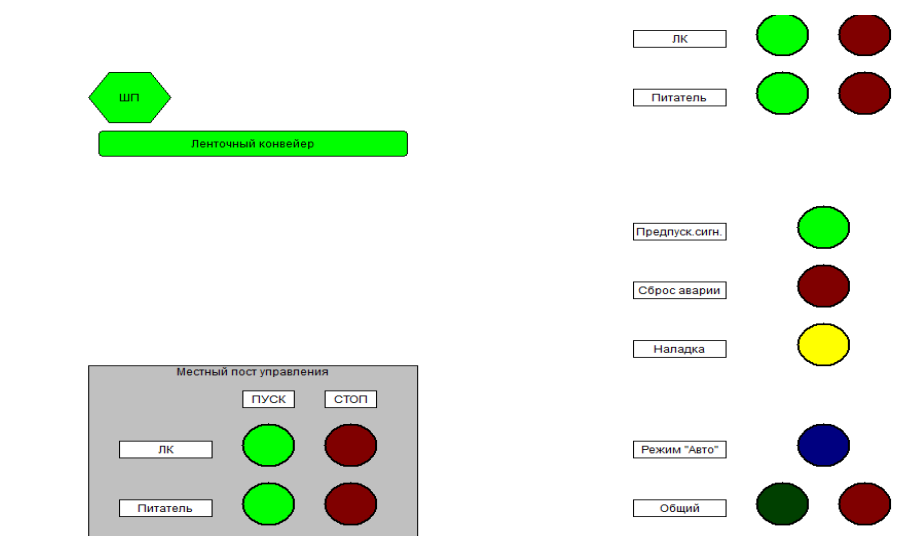


Рис. 37. Визуализация наладочного режима

Практическая работа №5

Моделирование системы автоматического регулирования температуры в MATLAB Simulink

1. Описание объекта управления и системы допущений

Объект управления включает в себя проточную ёмкость, в которую установлен паровой подогреватель воды. Вода в ёмкость подаётся с температурой 20°C и массовым расходом 40кг/мин . В ёмкости поддерживается постоянный уровень; масса воды, находящейся в ёмкости – 100 кг . Температура воды, выходящей из ёмкости – 80°C .

Стабилизация температуры воды осуществляется изменением расхода пара через змеевик.

Возмущающим воздействием является изменение расхода воды, поступающей в объект.

Необходимо подобрать настройки регулятора так, чтобы выполнялись ограничения на требуемую температуру $T=80^{\circ}\text{C}$, а интегрально–квадратичный критерий имел бы минимальное значение.

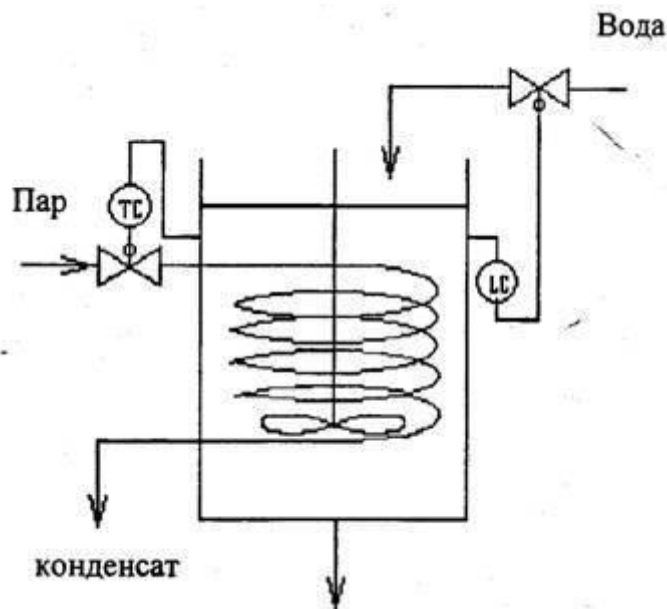


Рисунок 1 – Схема САР

В данном задании, для моделирования системы управления, нам необходимо рассмотреть только тепловые процессы, протекающие в объекте.

Данный объект представляет собой аппарат с идеальным перемешиванием потока (температура во всех точках аппарата одинакова). Поэтому математическая модель –

модель с сосредоточенными параметрами. Также будем считать, что теплофизические параметры от температуры не зависят. Отсюда, система допущений:

- теплофизические параметры считаем величинами постоянными;
- теплоемкостью материала реактора пренебрегаем;
- инерционность канала регулирования считаем пренебрежимо малой по сравнению с инерционностью объекта;
- запаздыванием при передаче управляющего воздействия пренебрегаем;
- пар конденсируется полностью;
- считаем толщину стенки змеевика бесконечно малой.

2. Анализ САР

Возмущающим воздействием является изменение расхода воды на входе в объект, регулируемый параметр – температура воды в проточной емкости, управляющее воздействие – изменение расхода греющего пара на входе в змеевик за счет изменения степени открытия клапана.

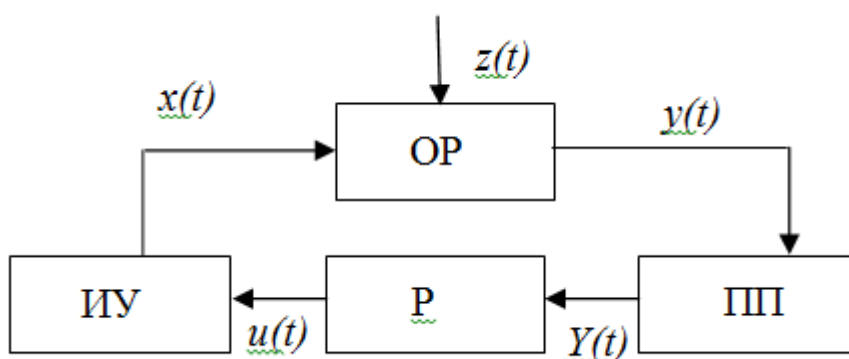


Рисунок 2 – Структурная схема САР

ОР – объект регулирования (проточная ёмкость);

ПП – первичный преобразователь;

Р – регулятор (ПИ-регулятор);

ИУ – исполнительное устройство (клапан);

$x(t)$ – расход греющего пара на входе в змеевик;

$y(t)$ – температура жидкости (регулируемый параметр);

$Y(t)$ – приведенная температура жидкости (безразмерная величина $0...1$)

$u(t)$ – управляющее воздействие ($0...1$);

$z(t)$ – расход жидкости на входе в объект (возмущающее воздействие).

3. Составление структурной схемы и математической модели объекта

В соответствии с принятой системой допущений структурная схема нашего объекта будет выглядеть следующим образом:

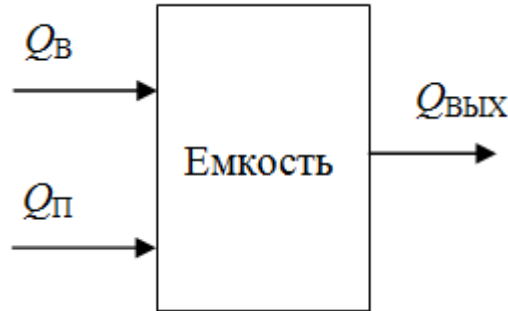


Рисунок 3 – Структурная схема объекта

где $Q_{в}$ – входящий тепловой поток воды, Дж/с;

$Q_{п}$ – входящий тепловой поток пара, Дж/с;

$Q_{вых}$ – выходной тепловой поток, Дж/с.

В проточной емкости происходит перенос тепла от греющего пара к воде, протекающей через емкость. Балансовое соотношение в общем виде выглядит следующим образом:

$$\sum n_{прих} - \sum n_{ух} = \frac{d}{dt} (кол - во) \quad (1)$$

где: $\sum n_{прих}$ – количество вещества или энергии, входящей в объект;

$\sum n_{ух}$ – количество вещества или энергии, уходящей из объекта;

$\frac{d}{dt} (кол - во)$ – производная по времени от количества вещества или энергии, находящейся в объекте.

Уравнение материального баланса может быть заменено тепловым балансом:

$$Q_{в} + Q_{п} - Q_{вых} = \frac{dQ}{dt} \quad (2)$$

Входящий тепловой поток воды рассчитываем по формуле [1]:

$$Q_{в} = m_{в} \cdot c_{в} \cdot T_{вх}, \quad (3)$$

где $m_{п}$ – массовый расход воды, кг/с;

$$m_B = \frac{40}{60} = \frac{2}{3} \text{ кг/с};$$

c_B – удельная теплоемкость воды [1], Дж/кг·К;

$$c_B = 4190 \text{ Дж/кг·К};$$

T_{BX} – температура воды, поступающей в проточную емкость °С;

$$T_{BX} = 20^\circ\text{C}.$$

Приходящий тепловой поток пара рассчитывается по формуле:

$$Q_{\Pi} = m_{\Pi} \cdot r, \quad (4)$$

где r – удельная теплота парообразования [1], Дж/К;

$$r = 2310000 \frac{\text{Дж}}{\text{К}};$$

m_{Π} – массовый расход пара, кг/с, который определяется из модели статики объекта.

Уходящий тепловой поток с водой рассчитывается по формуле:

$$Q_{\text{ВЫХ}} = m_B \cdot c_B \cdot T_3, \quad (5)$$

где m_B – массовый расход воды, кг/с;

c_B – удельная теплоемкость воды, Дж/кг·К;

T_3 – температура воды, уходящей из емкости, °С;

$$T_3 = 80^\circ\text{C}.$$

Производная от количества тепла, находящегося в емкости:

$$\frac{dQ}{dt} = \frac{dT \cdot M_B \cdot c_B}{dt} = M_B \cdot c_B \frac{dT}{dt}, \quad (6)$$

где M_B – масса воды, находящейся в емкости, кг;

$$M_B = 100 \text{ кг};$$

c_B – удельная теплоемкость воды [1], Дж/кг·К;

$\frac{dT}{dt}$ – производная от температуры по времени.

Подставив выражения (3), (4), (5), (6) в уравнение теплового баланса (2), получим:

$$m_B \cdot c_B \cdot (T_{BX} - T_3) + m_{\Pi} \cdot r = M_B \cdot c_B \frac{dT}{dt}. \quad (7)$$

Запишем начальное условие – значение температуры на выходе из емкости в момент времени равный нулю:

$$T|_{e=0} = 80^{\circ}C \quad (8)$$

Начальный массовый расход пара m_{Π}^0 в змеевике определяем из модели статики объекта:

$$m_B \cdot c_B \cdot (T_{BH} - T_3) + m_{\Pi}^0 \cdot r = 0, \quad (9)$$

откуда

$$m_{\Pi}^0 = \frac{m_B \cdot c_B \cdot (T_3 - T_{BH})}{r}, \quad (10)$$

где r – удельная теплота парообразования, Дж/К;

m_B – массовый расход воды, кг/с;

c_B – удельная теплоемкость воды, Дж / кг · К ;

T_3 – температура воды, уходящей из емкости, °С;

T_{BH} – температура воды, поступающей в проточную емкость °С.

Тогда модель динамики объекта регулирования выглядит следующим образом:

$$\begin{cases} m_B \cdot c_B \cdot (T_{BH} - T_3) + m_{\Pi} \cdot r = M_B \cdot c_B \frac{dT}{dt} \\ m_{\Pi}^0 = \frac{m_B \cdot c_B \cdot (T_3 - T_{BH})}{r} \\ T|_{e=0} = 80^{\circ}C \end{cases} \quad (11)$$

4. Составление математической модели САР температуры

Кроме объекта регулирования САР температуры содержит первичный преобразователь, ПИ-регулятор и исполнительное устройство в виде клапана (см. рисунок 1).

4.1 Модель первичного преобразователя (ПП)

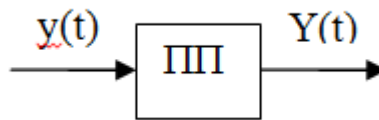


Рисунок 4 – Структурная схема ПП

где $y(t)$ – температура жидкости (регулируемый параметр);

$Y(t)$ – выходной сигнал с ПП (0...1).

Инерционность первично преобразователя бесконечно мала по сравнению с инерционностью объекта. На выходе первичного преобразователя имеется электрический сигнал. Электрический сигнал может быть по току, по напряжению, с разными диапазонами, цифровой и т.д., но в любом случае минимальному значению измеряемой величины соответствует минимальное значение выходного сигнала, а максимальному – максимальное значение выходного сигнала. Для единообразия модели выходной сигнал в модели представляется безразмерной переменной, изменяющейся в пределах от 0 до 1.

$$\begin{cases} Y(t) = \frac{1}{y_{\max} - y_{\min}} \cdot y(t), & y_{\min} \leq y(t) \leq y_{\max} \\ Y(t) = 1, & y(t) > y_{\max} \end{cases} \quad (12)$$

y_{\max} , y_{\min} – пределы измерения конкретного преобразователя.

В качестве первичного преобразователя выбираем термометр сопротивления медный ТСМ-9623 с диапазоном измерения 0...120°C.

4.2 Модель регулятора

Зависимость, по которой выходной сигнал ПП $Y(t)$ преобразуется в регулирующее воздействие, U называется законом регулирования.

Управляющее воздействие регулятора определяется законом регулирования.

Для ПИ-закона регулирования:

$$U = K_y \varepsilon + \frac{1}{T_i} \int \varepsilon dt, \quad (13)$$

где K_y – коэффициент усиления регулятора;

T_i – время интегрирования;

ε – ошибка регулирования.

Условимся, что в начальный момент времени регулирующее воздействие равно нулю.

$$U|_{t=0} = 0. \quad (14)$$

Ошибка регулирования или рассогласование ε находится по следующей формуле:

$$\varepsilon = \frac{T - T_3}{(T_{\max} - T_{\min})}. \quad (15)$$

4.3 Модель исполнительного устройства (ИУ)

Допущения: пренебрегаем инерционностью ИУ.

Степень открытия клапана считаем:

$$A = A_0 + U, \quad (16)$$

где U – регулирующее воздействие;

A_0 – начальная степень открытия клапана. Принимаем $A_0=0,5$.

Расходную характеристику в нашем случае будем считать линейной.

$$m_{II} = k \cdot A, \quad (17)$$

где A – степень открытия клапана;

k – коэффициент передачи клапана. Находим из начальных условий:

$$k = m_{II}^0 / A_0, \quad (18)$$

4.4 Модель динамики САР температуры

Учитывая уравнения (11), (12), (13), (14), (15), (16) и (18) получим модель динамики САР температуры:

$$\left\{ \begin{array}{l} m_B \cdot c_B \cdot (T_{BX} - T_3) + m_{II} \cdot r = M_B \cdot c_B \frac{dT}{dt} \\ m_{II}^0 = \frac{m_B \cdot c_B \cdot (T_3 - T_{BX})}{r} \\ \varepsilon = \frac{(T - T_3)}{(T_{\max} - T_{\min})} \\ U = K_y \varepsilon + \frac{1}{T_u} \int \varepsilon dt \\ A = A_0 + U \\ k = m_{II}^0 / A_0 \\ T|_{t=0} = 80^\circ C \\ U|_{t=0} = 0 \end{array} \right. \quad (19)$$

5 Создание модели САР температуры в MatLab

5.1 Определение параметров модели

Для определения всех констант создаем M-файл «Variable.m»:

```
1 - tz=80 % Заданная температура воды, уходящая из емкости, С
2 - tmin=0 % Минимальный предел измерения датчика, С
3 - tmax=120 % Максимальный предел измерения датчика, С
4 - Tvn=20 % Температура воды, поступающая в проточную емкость, С
5 - cv=4190 % Удельная теплоемкость воды, Дж/(кг*К)
6 - M=100 % Масса воды, находящейся в емкости, кг
7 - mv= 40/60 % Массовый расход воды, кг/с
8 - rp=2310000 % Удельная теплота парообразования, Дж/К
9 - mp=(cv*mv*(tz-Tvn))/rp % Массовый расход пара, кг/с
10 - A0=0.5 % Начальная степень открытия клапана
11 - k1=mp/A0 % Коэффициент передачи клапана
```

Рисунок 5 – Создание M-файла

В этом файле описываем все заданные константы, а также начальные значения, найденные из моделей статики. Запускаем файл на исполнение, чтобы переменные внеслись в память.

5.2 Создание модели объекта

В Simulink создаем отдельно объект, используя блоки Constant и Fcn (пользовательская функция). При этом в интеграторе необходимо задать Initial condition равное tz (заданное значение температуры).

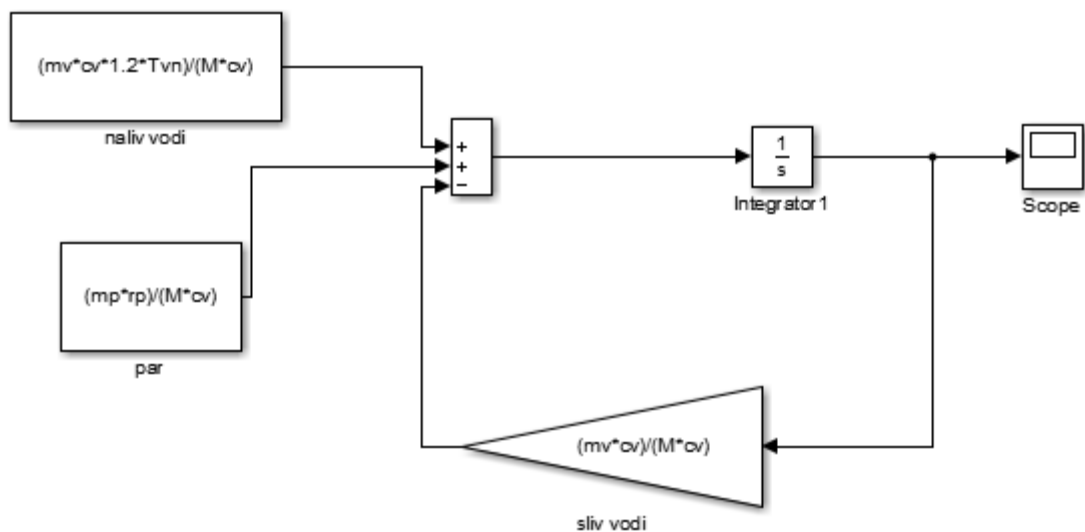


Рисунок 6 – Создание модели объекта

Возмущающим воздействием в нашей системе является изменение расхода поступающей в проточную емкость воду.

Переходная характеристика объекта при ступенчатом изменении расхода воды на 20% будет выглядеть следующим образом:

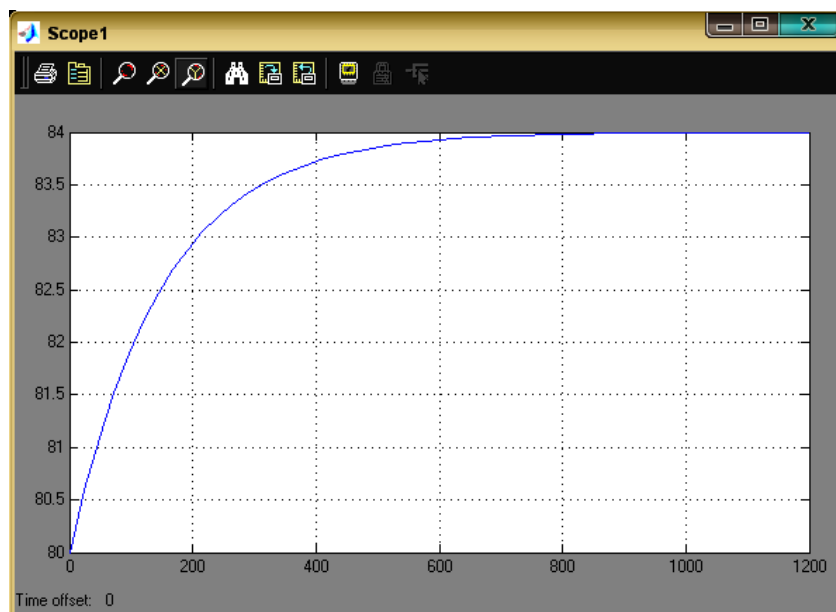


Рисунок 7 – Переходная характеристика объекта при ступенчатом изменении расхода воды

Созданный нами объект объединяем в подсистему (предварительно создав усилительные блоки вместо констант на входе). Входами подсистемы будут возмущающее воздействие и массовый расход пара (управляющее воздействие).

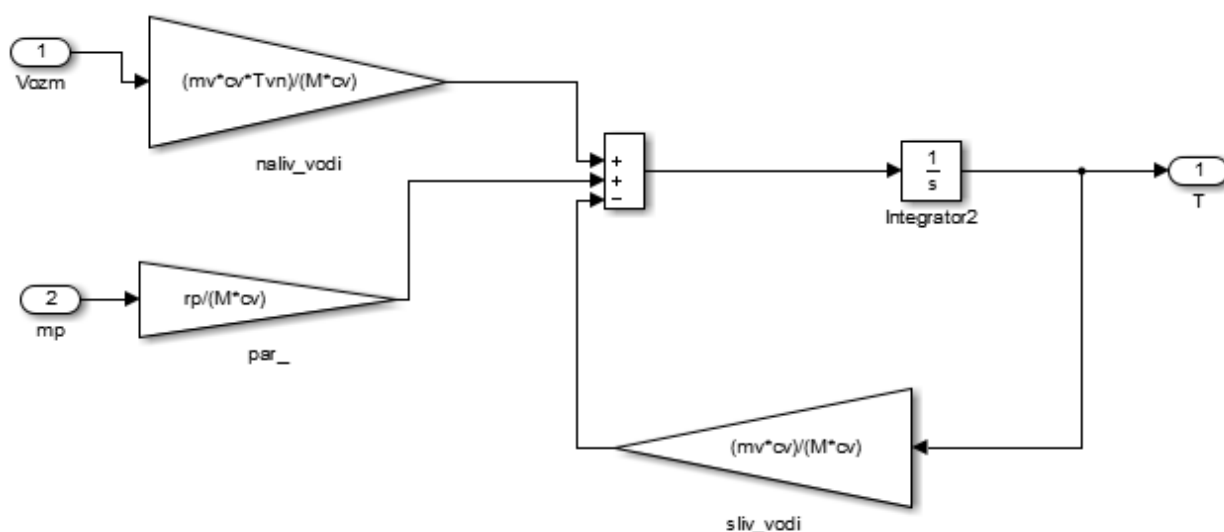


Рисунок 8 – Подсистема «Объект»

Вход «Vozm» необходим для подачи возмущения.

На вход «mp» поступает сигнал от исполнительного устройства, изменяющий расход греющего пара.

Выход «T» служит для передачи сигнала, выходного параметра, температуры в контур регулирования.

5.3 Создание модели ПИ-регулятора, ИУ, ПП

Аналогично создаем модель ПИ-регулятора и объединяем в подсистему «ПИ-Регулятор»:

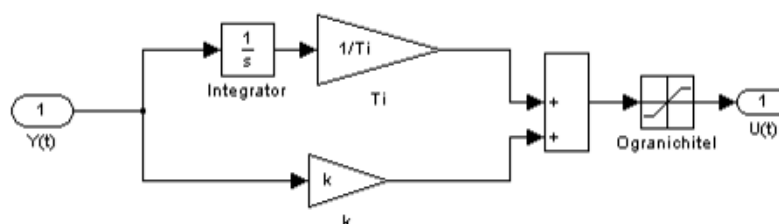


Рисунок 9 – Подсистема «ПИ-Регулятор»

где блоки: «k» – для умножения ошибки регулирования на коэффициент усиления;

«Ti» – для учета времени интегрирования;

«Ogranichitel» – необходим для предотвращения выхода значения величины управляющего воздействия за допустимые границы(0...1);

«Integrator» – в свойствах задаем начальное регулирующее воздействие равное нулю.

Модель исполнительного устройства создаем по аналогии.

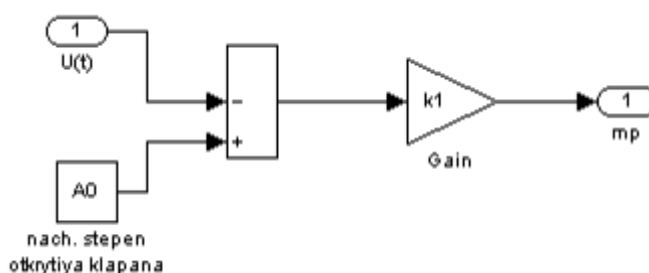


Рисунок 10 – Модель исполнительного устройства

Выходной сигнал ИУ – новый расход греющего пара G_{II} в змеевике при уточненной (новой) степени открытия регулирующего органа A .

При помощи функций блока «Fsp» создаем модель первичного преобразователя.

5.4 Модель САР температуры

После реализации всех созданных нами подсистем, объединяем их в соответствии со структурной схемой САР температуры (Рисунок 2).

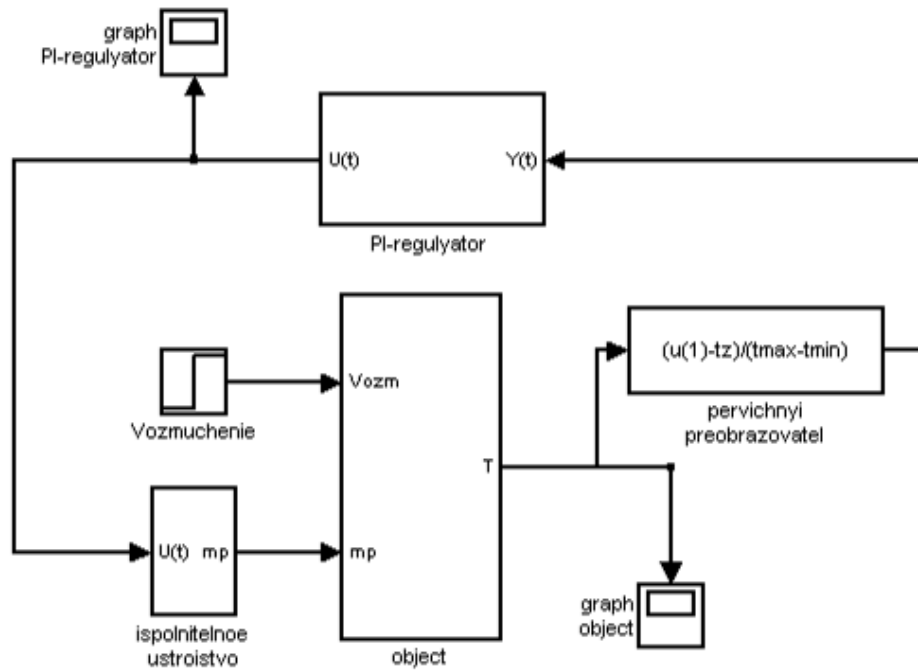


Рисунок 11 – Модель САР температуры

Процесс моделирования проводим в интервале времени от 0 до 1000 с.

В результате получаем следующие графики переходного процесса при настройках регулятора $K_y=1$ и $T_i=1$ (рисунок 12, 13):

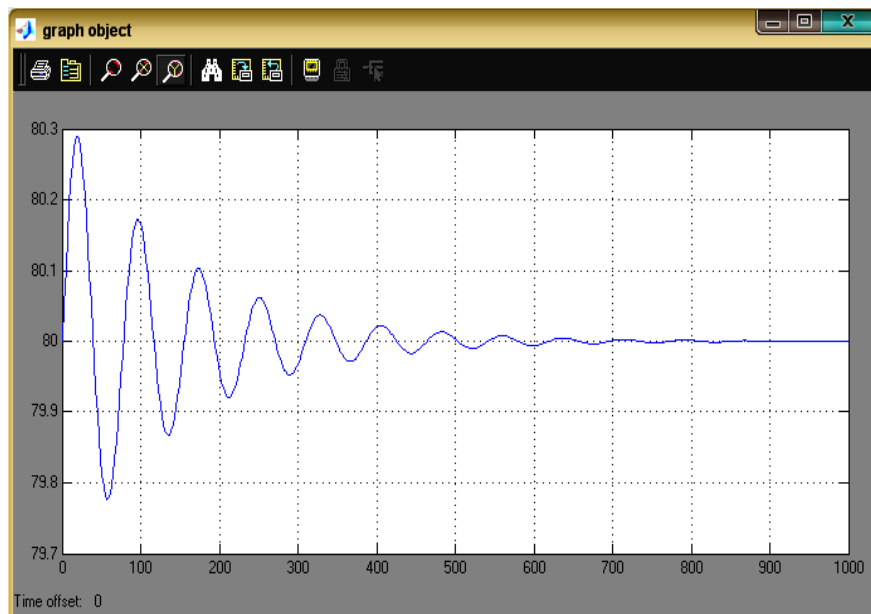


Рисунок 12 – Переходный процесс в САР температуры

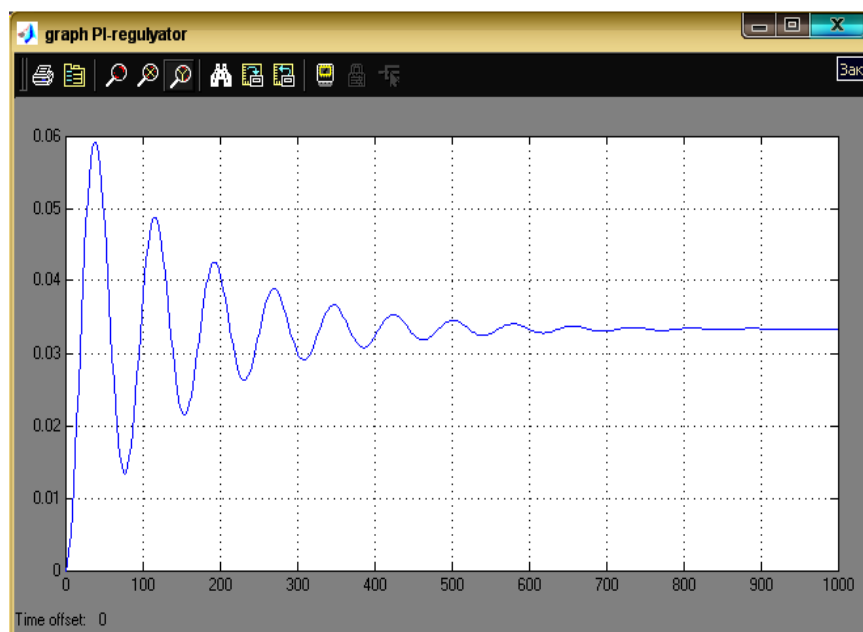


Рисунок 13 – Регулирующее воздействие ПИ-регулятора

Далее необходимо, пользуясь алгоритмом ручной настройки, определить оптимальные параметры ПИ-регулятора, получив переходный процесс при минимальном перерегулировании.

5.5 Алгоритм ручной настройки ПИ-регулятора

Расчёт параметров по формулам может не дать оптимальной настройки регулятора, поскольку аналитически полученные результаты основываются на сильно упрощённых моделях объекта.

В частности, в них не учитывается всегда присутствующая нелинейность типа «ограничение» для управляющего воздействия. Кроме того, модели используют параметры, идентифицированные с некоторой погрешностью. Поэтому после расчёта параметров регулятора желательно сделать его подстройку. Подстройку можно выполнить на основе правил, которые используются для ручной настройки.

Эти правила получены из опыта, теоретического анализа и численных экспериментов. Они сводятся к следующему:

- увеличение пропорционального коэффициента увеличивает быстродействие и снижает запас устойчивости;
- с уменьшением интегральной составляющей ошибка регулирования с течением времени уменьшается быстрее;
- уменьшение постоянной интегрирования уменьшает запас устойчивости;
- увеличение дифференциальной составляющей увеличивает запас устойчивости и быстродействие.

Практическое применение правил настройки не всегда возможно. Например, при регулировке тепловых процессов настройка по правилам может занять недопустимо много времени.

Отметим, что применение правил возможно только после предварительной настройки регулятора по формулам. Попытки настроить регулятор без начального приближённого расчёта коэффициентов могут быть безуспешными.

Сформулированные правила справедливы только в окрестности оптимальной настройки регулятора. Вдали от неё эффекты могут быть иными.

Если приближенные к оптимальным параметры настройки регулятора еще неизвестны, то для удержания регулирующего контура в стабильном состоянии применяются следующие установки (перед началом настройки):

- коэффициент пропорциональности $K_p = 0,1$;
- время интегрирования $T_i = 10000$ сек (отключено);
- время дифференцирования $T_d = 0$ (отключено).

После определения исходных условий настройки для ПИ-регулятора выполняют последовательность действий (***алгоритм ручной настройки***):

- Медленно увеличивать K_p пока регулирующий контур при малых изменениях заданной величины не начнет клониться к колебаниям.
- Незначительно уменьшить K_p для устранения колебаний.
- Уменьшать T_i до тех пор, пока в регулирующем контуре снова не начнут проявляться колебания.
- Медленно увеличивать T_i до тех пор, пока тенденция к колебаниям не будет устранена.
- Повторить шаги до получения хороших показателей качества (3-4 итерации).

5.6 Варианты индивидуальных заданий

№ варианта	T_{vn}	T_z
1	10	90
2	15	75
3	10	80
4	15	90
5	20	75
6	20	90
7	15	80
8	10	70
9	25	70

10	25	90
11	10	60
12	25	80
13	15	60
14	10	60
15	20	80

Практическая работа №6

Моделирование САР уровня воды в парогенераторе в MATLAB Simulink

1. Автоматическое регулирование уровня воды в парогенераторе

Регулирование питания в парогенераторе (ПГ) сводится к поддержанию материального баланса между отводом пара, продувкой и подачей питательной воды. Параметром, характеризующим материальный баланс, является уровень воды в ПГ.

К стабилизации уровня предъявляются довольно жесткие требования. Для ПГ с ВВЭР-1000 номинальный уровень $L_{ном}$ составляет 2450 мм от внутренней образующей корпуса. Точность поддержания уровня в статических режимах составляет ± 50 мм от $L_{ном}$, в динамике ± 150 мм от $L_{ном}$ (с учетом нечувствительности регулятора). Повышение уровня воды от $L_{ном}$ не допускается из-за затопления и нарушения работы сепарационных устройств (заброс воды в турбину), а снижение уровня - из-за оголения поверхности нагрева.

Возмущающими воздействиями на уровень являются:

- расход пара (нагрузка);
- изменение расхода питательной воды;
- изменение температуры питательной воды;
- изменение расхода продувки;
- изменение теплоподвода со стороны первого контура (изменение средней температуры первого контура или отключение ГЦН).

При возмущении расходом пара или отключении ГЦН проявляется явно выраженное «набухание» уровня, т.е. изменение его в начальные моменты времени в сторону, не соответствующую знаку возмущающего воздействия. Явление «набухания» можно объяснить так.

При изменении расхода пара, например увеличении, давление в ПГ уменьшается, происходит дополнительное вскипание воды, увеличение уровня. В дальнейшем уровень начинает падать, поскольку расход пара увеличился, а расход питательной воды остался прежним. При отключении ГЦН теплоподвод в ПГ резко уменьшается, интенсивность кипения и объем пузырьков пара уменьшаются, что приводит к снижению уровня. Однако уменьшение парообразования приводит к снижению давления и некоторому увеличению парообразования. Тем не менее первый фактор более существенный. В дальнейшем уровень увеличивается, поскольку расход питательной воды остался неизменным.

Рассмотренные динамические свойства ПГ являются крайне неблагоприятными с точки зрения стабилизации уровня воды. Этим объясняется тот факт, что для автоматического регулирования уровня не пригодны обычные одноконтурные системы регулирования.

Динамику ПГ по уровню по всем входным возмущениям можно описать произведением передаточных функций интегрального и интегро-дифференцирующих звеньев. Эту передаточную функцию можно представить суммой передаточных функций инерционного и интегрального звеньев.

2. Выбор схемы и закона регулирования уровня

Применение пропорционально-интегрального (ПИ) закона регулирования для астатического объекта с явлением «набухания» не обеспечивает требуемого качества регулирования (длительные слабо затухающие колебания уровня при ступенчатом входном возмущении). Интегральный (И) закон также дает плохую устойчивость системы. Пропорциональный (П) закон не допустим из-за статической ошибки регулирования. Поэтому для регулирования уровня в ПГ применяют комбинированную САР: регулирование по отклонению с П-регулятором и контуром инвариантности по основному возмущающему воздействию - расходу пара.

2-импульсная схема регулирования не применяется по следующим причинам:

- 1) расход питательной воды через регулирующий питательный клапан зависит не только от положения клапана, но и от перепада давления на нем, который в процессе эксплуатации может изменяться;
- 2) в дифманометрах-расходомерах прежних лет выпуска выходной сигнал был пропорционален корню квадратному из перепада давления.

Указанные недостатки 2-контурной САР устраняются введением в регулятор третьего импульса по расходу питательной воды от дифманометра-расходомера. Такая 3-импульсная САР изображена на рисунке 1.

Принцип работы САР следующий. Сигналы по расходу пара и питательной воды вводятся в регулятор с противоположными знаками. В установившемся состоянии эти сигналы равны, противоположны по знаку и, следовательно, компенсируют друг друга.

Сигнал по уровню воды в ПГ компенсируется сигналом задания. При изменении расхода пара мгновенно изменяется соответствующий сигнал на входе в регулятор и последний пропорционально изменяет расход питательной воды, не дожидаясь изменения уровня.

В регуляторе используется ПИ-закон регулирования, однако вследствие ввода в регулятор практически безинерционной отрицательной обратной связи по расходу питательной воды в нем реализуется П-закон регулирования (аналогия жесткой обратной связи по положению регулирующего органа). Статическая неравномерность П-регулятора устраняется корректирующим сигналом по расходу пара.

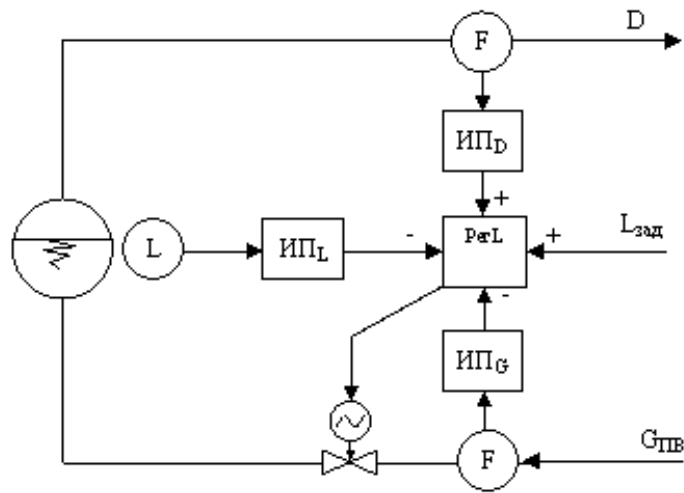


Рисунок 1 – Принципиальная схема 3-импульсной САУ уровня воды в парогенераторе
3. Идентификация технологического объекта по кривой разгона

Экспериментальным путем были получены разгонные кривые парогенератора по уровню при подаче на него возмущений расходом питательной воды и расходом пара (рисунок 2, 3).

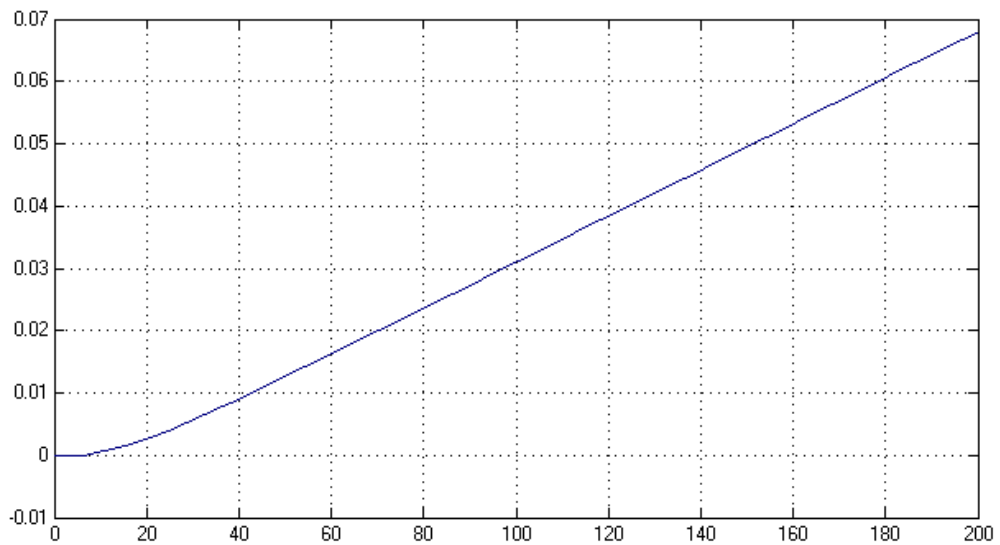


Рисунок 2 - Кривая разгона ПГ по уровню при подаче скачкообразного возмущения расходом питательной воды $DG = 10 \text{ кг/с}$

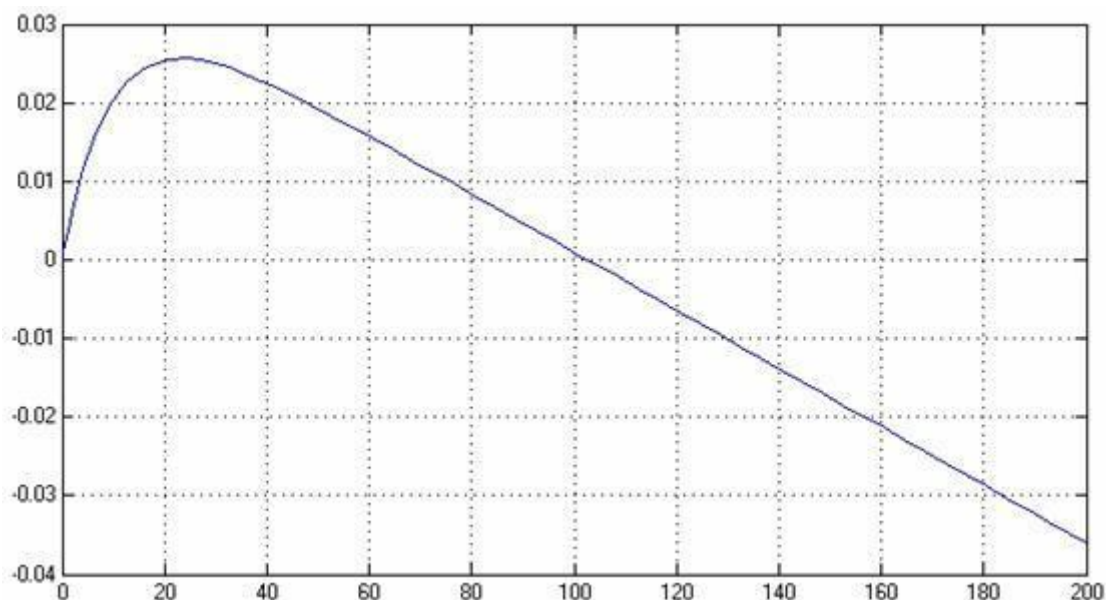


Рисунок 3 - Кривая разгона ПГ по уровню при подаче скачкообразного возмущения расходом пара $DD = 10$ кг/с

Данные свойства парогенератора можно описать суммой передаточных функций инерционного и интегрального звеньев, а коэффициенты определить экспертно-опытным путем при помощи программы Simulink, входящей в программный пакет Matlab.

Структурная схема модели представлена на рисунке 4.

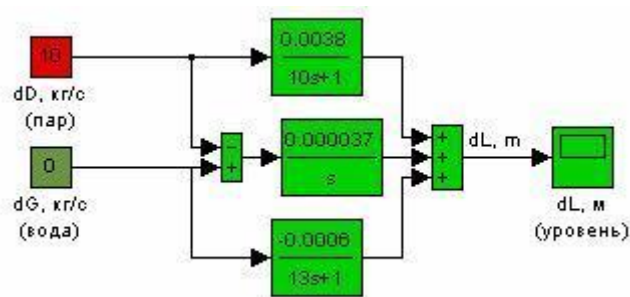


Рисунок 4 - Структурная схема модели парогенератора

4. Идентификация измерительных преобразователей уровня, расходов пара и воды

Уровень в ПГ измеряется датчиком типа Сапфир-22ДД по малому уровнемеру с пределом измерения 0-1000 мм, используя двухкамерный уравнительный сосуд. Статическая характеристика уровнемера приведена на рисунке 5-а.

Расход воды измеряется по перепаду давления на сужающем устройстве типа «диафрагма» датчиком типа Сапфир-22ДД с выходным токовым сигналом 4 - 20 мА с пределами измерения 0 - 555 кг/с (0 - 2000 т/ч). Статическая характеристика расходомера приведена на рисунке 5-б.

Расход пара оценивается по температуре и давлению в трубопроводе I контура до и после ПГ. Сигнал расхода пара подается в виде нормированного токового сигнала 4 - 20 мА с пределом измерения 0 - 555 кг/с (0 - 2000 т/ч). Статическая характеристика расходомера приведена на рисунке 5-б.

Таким образом, измерительные преобразователи можно описать передаточными функциями пропорциональных звеньев с коэффициентами передачи, которые равняются:

$$K_{ипЛ} = 16, \text{ мА/м} - \text{ для измерения уровня воды в ПГ};$$

$K_{ипГ} = K_{ипД} = 0.0288, \text{ мА/(кг/с)}$ - для измерения расходов питательной воды и пара соответственно.

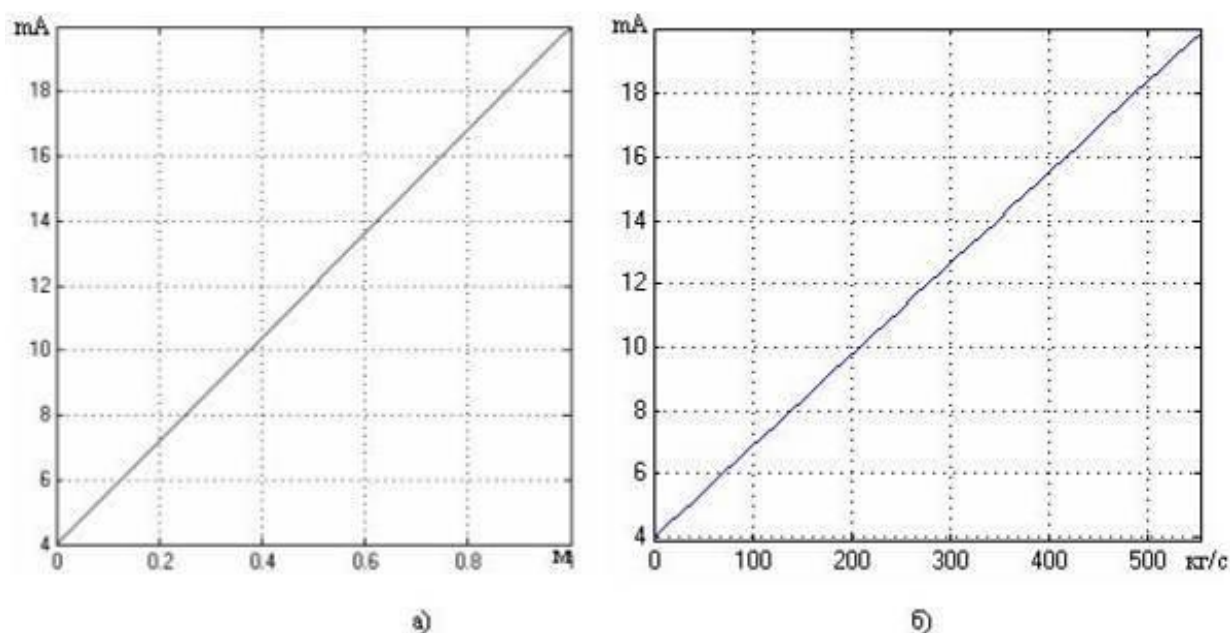


Рисунок 5 - Статические характеристики ИП типа «Сапфир -22ДД» для измерения уровня (а) и расходов питательной воды и пара (б)

5. Идентификация регулирующего клапана

Регулирование уровня воды осуществляется путем изменения расхода питательной воды через регулирующий питательный клапан, который осуществляет регулирование в пределах 0..100%.

Позиционер управляет приводом электродвигателя постоянного тока регулирующего питательного клапана. На позиционер поступает сигнал задания открытия клапана в виде нормированного токового сигнала 0..20мА. Максимальная скорость перемещения штока составляет 5%/с.

Расходная характеристика регулирующего питательного клапана приведена на рисунке 6.

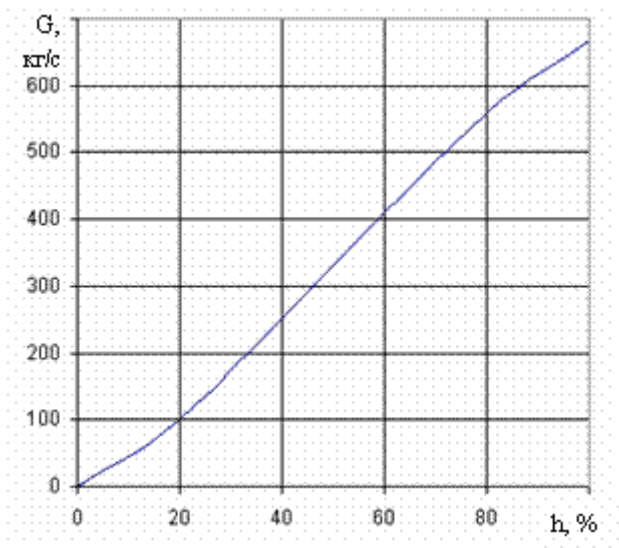


Рисунок 6 - Расходная характеристика регулирующего питательного клапана

Как видно, расходная характеристика клапана близка к линейной. Таким образом, для моделирования расхода питательной воды через клапан по каналу «Степень открытия, % - расход на клапане, кг/с» можно применить пропорциональное звено с коэффициентом пропорциональности $K_{p0}=6.67$ (кг/с)/%. На вход звена должен подаваться сигнал положения штока в процентах хода РО (0-100 %xРО).

Трубопровод от клапана до парогенератора моделируется инерционным звеном 1-го порядка с коэффициентом передачи $K_{tp}=1$ и постоянной времени $T_{tp}=2$ с.

Структурная схема модели РО с позиционером, и трубопроводом до ПГ изображена на рисунке 7.

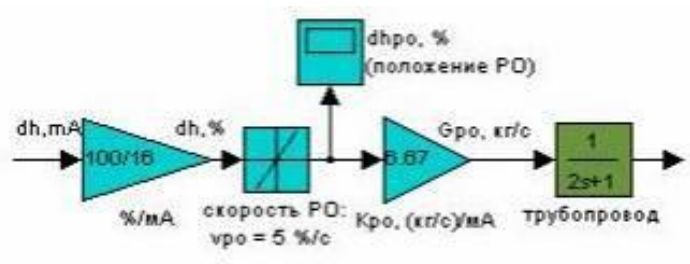


Рисунок 7 - Структурная схема модели РО с двигателем постоянного тока и позиционером, трубопроводом к ПГ

6. Составление структурной схемы САР

По принципиальной схеме 3-импульсной САР уровня, описанной в п.2, составляем структурную схему. Структурная схема модели 3-импульсной САР питания ПГ представлена на рисунке 8. Полученные переходные процессы при скачкообразном возмущении расходом пара представлены на рисунке 9.

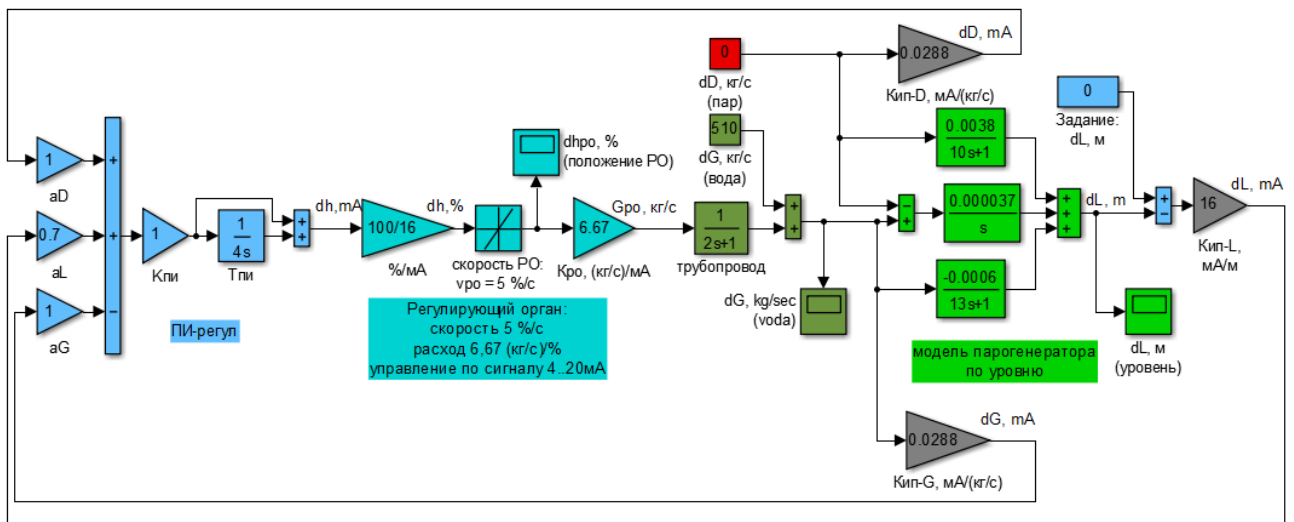


Рисунок 8 - Структурная схема модели 3-импульсной САР питания ПГ

7. Выводы

Использование 3-импульсной САР уровня воды в ПГ с ПИ-регулятором позволяет регулировать объект с эффектом «набухания».

Таким образом, возможно подавление скачкообразных возмущения расходом пара величиной до 18 кг/с без выхода уровня из 50-миллиметровой зоны. Возмущения же расходом питательной воды практически любой величины (до 510 кг/с), подаваемые на ПГ не приводят к выходу уровня из 50-миллиметровой зоны.

При всем этом мы имеем апериодический переходный процесс регулирования уровня со временем регулирования менее чем 200 с.

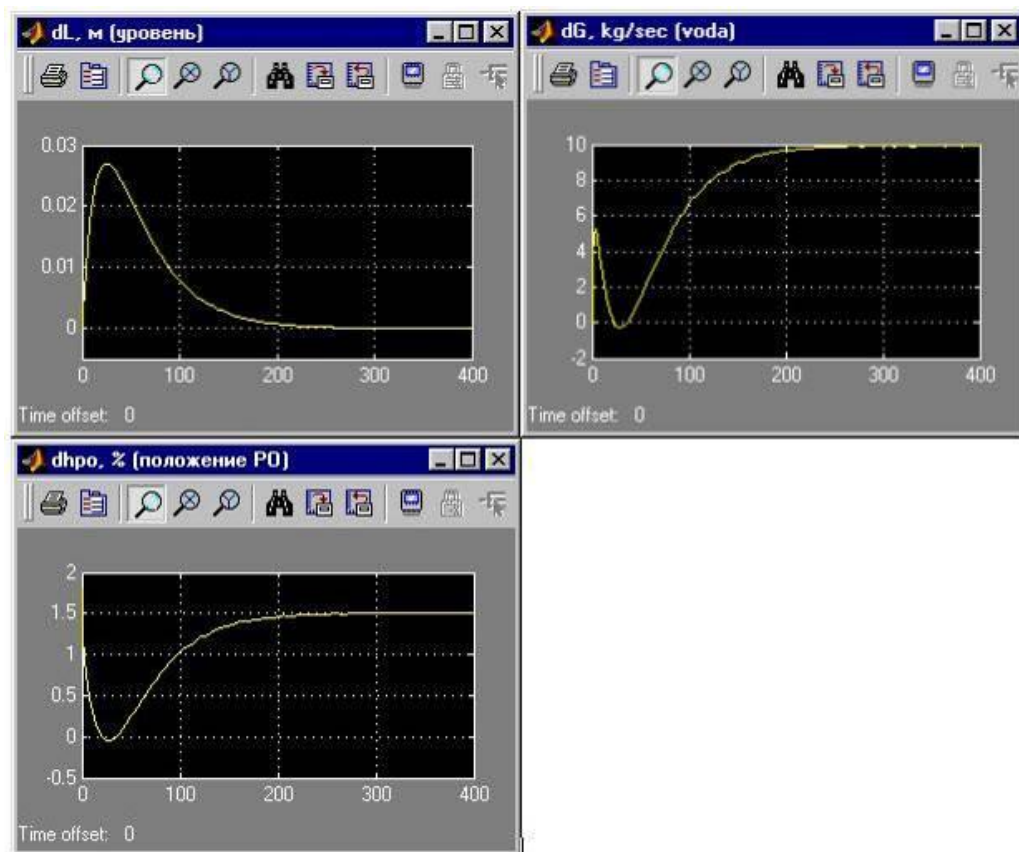


Рисунок 9 – Графики переходных процессов в САР уровня воды в ПГ при подаче скачкообразного возмущения расходом пара ($dD=10$ кг/с)

