

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Емельянов Сергей Геннадьевич  
Должность: ректор  
Дата подписания: 13.06.2022 21:41:22  
Уникальный программный ключ:  
9ba7d3e34c012eba476ffd2d064cf2781953be730df2374d16f3c0ce536f0fc6

МИНОБРАЗОВАНИЯ И НАУКИ РОССИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра электроснабжения

УТВЕРЖДАЮ

Проректор по учебной работе  
О.Г. Локтионова

« 13 » 06

2022 г.



## ОСНОВЫ АСУ ЭЛЕКТРОУСТАНОВКАМИ ЭЛЕКТРОСТАНЦИЙ И ПОДСТАНЦИЙ

Методические указания к выполнению лабораторных работ  
для студентов направления подготовки 13.03.02  
Электроэнергетика и электротехника

Курск 2022

УДК 621.31

Составители: В.И.Бирюлин, Д.В.Куделина

Рецензент:

Кандидат технических наук, доцент кафедры  
«Электроснабжение» *В.Н. Алябьев*

**Основы АСУ электроустановками электростанций и подстанций:** методические указания к лабораторным работам/ Юго-Зап. гос. ун-т; сост.: В.И.Бирюлин, Д.В.Куделина. – Курск, 2022. – 31 с.: – Библиогр.: с.32.

Содержат сведения по выполнению лабораторных работ по дисциплине «Основы АСУ электроустановками электростанций и подстанций», в том числе использованию программируемых реле для построения этих систем. Рассматриваются способы составления программ для этих реле на примерах применения данных реле для решения различных производственных задач.

Предназначены для направления подготовки 13.03.02  
Электроэнергетика и электротехника всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ.л. 1,8. Уч.–изд.л.1,84 . Тираж 100 экз. Заказ

.Бесплатно.

Юго-Западный государственный университет.

305040, г.Курск, ул.50 лет Октября, 94

# ЛАБОРАТОРНАЯ РАБОТА № 1

## ОСНОВЫ РАБОТЫ С ПРОГРАММИРУЕМЫМИ РЕЛЕ

*Цель работы:* ознакомление с основами работы с программируемыми реле типа ПР-110, применяемыми для автоматизации различных производственных процессов.

### Краткие методические указания

Программируемое реле ПР110 предназначено для построения простых автоматизированных систем управления, а также для замены релейных систем защиты и контроля.

Логика работы реле ПР110 определяется пользователем в процессе программирования с помощью OwenLogic – разработки программы и записи ее в память реле.

На рисунке 1 приведено расположение контактов реле ПР110-220.

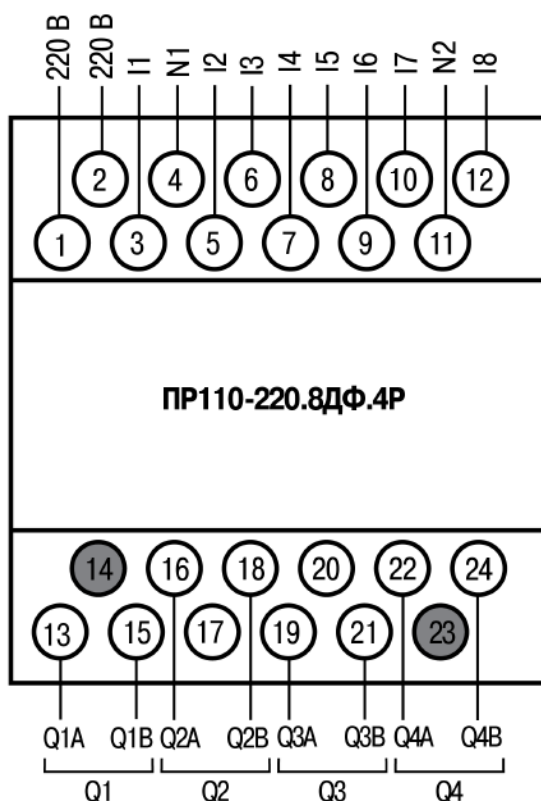


Рисунок 1 – Расположение контактов на клеммниках реле ПР110-220

Управление работой реле ПР110-220 осуществляется от внешних устройств – кнопок, датчиков и т.д. Эти устройства обеспечивают подачу напряжения 220 В при своем срабатывании,

например, замыкании кнопок «Пуск» или «Стоп». Внешние устройства подключаются двумя группами – группа 1 и группа 2 – рис.2.

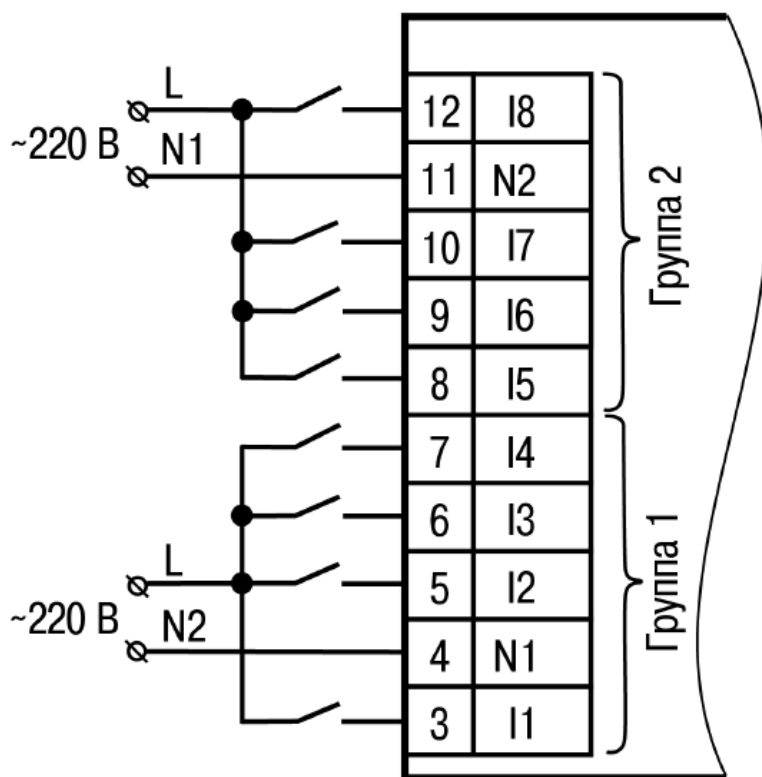


Рисунок 2 – Подключение внешних устройств управления к реле ПР110-220

Нагрузка подключается к электрической сети с помощью замыкающихся контактов, образующих выходы реле Q1 – Q4. Эти выходные контакты срабатывают в соответствии с логикой, определяемой программой, загруженной в память прибора. Схема приведена на рис.3.

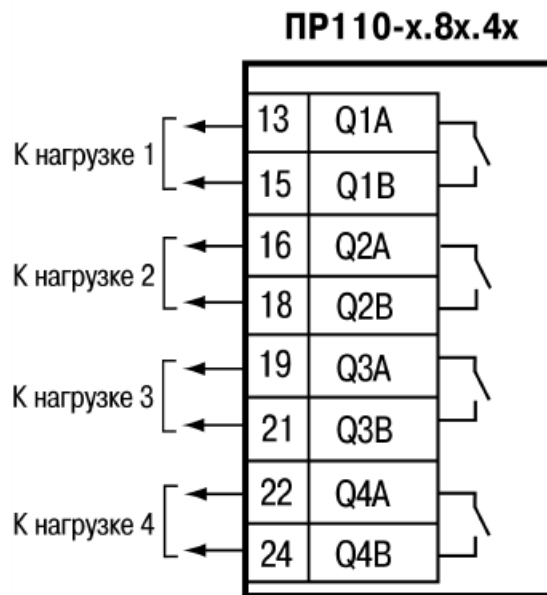


Рисунок 3 – Выходные цепи реле ПР110-220

Реле ПР110-220 является устройством со свободно-программируемой логикой. Логика работы реле программируется на ПК в OwenLogic (среда разработки программ).

Пользовательская программа записывается в энергонезависимую Flash-память реле. По окончании записи ПР110-220 автоматически перезагрузится, и записанная новая пользовательская программа запустится на выполнение.

Пользовательская программа, хранящаяся в памяти реле, начинает выполняться после подачи напряжения питания. Время установления рабочего режима после записи в реле пользовательской программы или после подачи напряжения питания не более 0,5 с

Работу реле ПР110-220 можно представить в виде последовательно выполняемых шагов (рабочий цикл):

1. Логическое состояние входов автоматически записывается в ячейки памяти входов (количество ячеек равно числу входов –  $I1...In$ ).

2. Программа считывает значения из ячеек памяти входов и выполняет над ними логические операции в соответствии с алгоритмом работы.

3. После обработки всей программы результаты записываются на физические выходы прибора (для включения выходных элементов  $Q1...Q4$ ).

4. Переход к Шагу 1 (после выполнения всех предыдущих шагов обработки программы цикл работы прибора повторяется с первого шага).

Время выполнения всех шагов зависит от сложности алгоритма программы.

### **Порядок выполнения работы**

1. Ознакомиться с устройством программируемого реле ПР110.

2. Изучить основные характеристики реле типа ПР110.

3. Для предложенной преподавателем структурной схемы составить схему подключения к входам реле управляющих устройств.

4. Для предложенной преподавателем структурной схемы составить схему подключения к выходам реле нагрузочных устройств.

5. Повторить п.3 и 4 для других структурных схем.

### **Содержание отчета**

1. Название работы

2. Цель работы

3. Схемы подключения к входам реле управляющих устройств.

4. Схемы подключения к выходам реле нагрузочных устройств.

5. Выводы по работе.

### **Контрольные вопросы**

1. Для чего применяются программируемые реле?

2. Как расшифровываются обозначения реле типа ПР110?

3. Как подключаются датчики к реле ПР110?

4. Как подключается нагрузка к реле ПР110?

## ЛАБОРАТОРНАЯ РАБОТА № 2

### ОСНОВЫ РАБОТЫ С ЯЗЫКОМ FBD

*Цель работы:* ознакомление с основами работы с языком функциональных блоков и диаграмм FBD, применяемым для создания программ автоматизации различных производственных процессов.

#### Краткие методические указания

Программирование реле ПР110 производится на языке FBD. Развитие идеи программной реализации электрических схем привело к появлению языка функциональных блоков и диаграмм (ФБД), FBD (Function Block Diagram). Диаграмма FBD очень напоминает принципиальную схему электронного устройства на микросхемах.

Выходы экземпляров функциональных блоков могут быть поданы на входы других блоков либо непосредственно на выходы ПЛК. Сами блоки, представленные на схеме как функциональные модули, могут выполнять стандартные и специальные функции.

Программируемые FBD схемы ясно отражают взаимосвязь входов и выходов диаграммы. Если алгоритм хорошо описывается с позиции сигналов, то его FBD – представление всегда получается значительно нагляднее, чем в текстовых языках.

При программировании используются наборы библиотечных блоков. Блок (элемент) – это подпрограмма, функция или функциональный блок (И, ИЛИ, НЕ, триггеры, таймеры, счетчики, математические операции и др.).

Каждая отдельная цепь блоков представляет собой выражение, составленное графически при разработке программы. В этой цепи к выходу отдельного блока подключается вход другого блока, образуя тем самым эту цепь. Внутри этой цепи блоки выполняются строго в порядке их соединения. Результат вычисления цепи может записываться во внутреннюю переменную программы, либо подаваться на выход реле.

Далее рассмотрим составление программ в среде OWEN Logic, она является средой программирования, предназначенной для создания алгоритмов работы коммутационных приборов,

относящихся к классу программируемых реле, в частности, серий ПР110 производства компании OWEN.

OWEN Logic позволяет пользователю разработать программу автоматизации системы по собственному алгоритму и записать ее в энергонезависимую память реле. Для составления программы используется графический язык FBD, который применяется в цифровых электрических схемах.

После запуска OWEN Logic на мониторе ПК открывается главное окно, содержащее следующие элементы (рис.1):

- главное меню: Файл, Вид, Прибор, Сервис, Расширения, Помощь;
- панели инструментов;
- панели Библиотека компонентов, Свойства и Переменные (до открытия или создания проекта в них нет информации);
- рабочую область проекта – поле редактирования программы (до открытия или создания проекта пустое);
- строку состояния в нижней части главного окна, показывающую информация о доступных ресурсах прибора и подключении к OWEN Logic.
- менеджер экранов (доступен только для проектов).

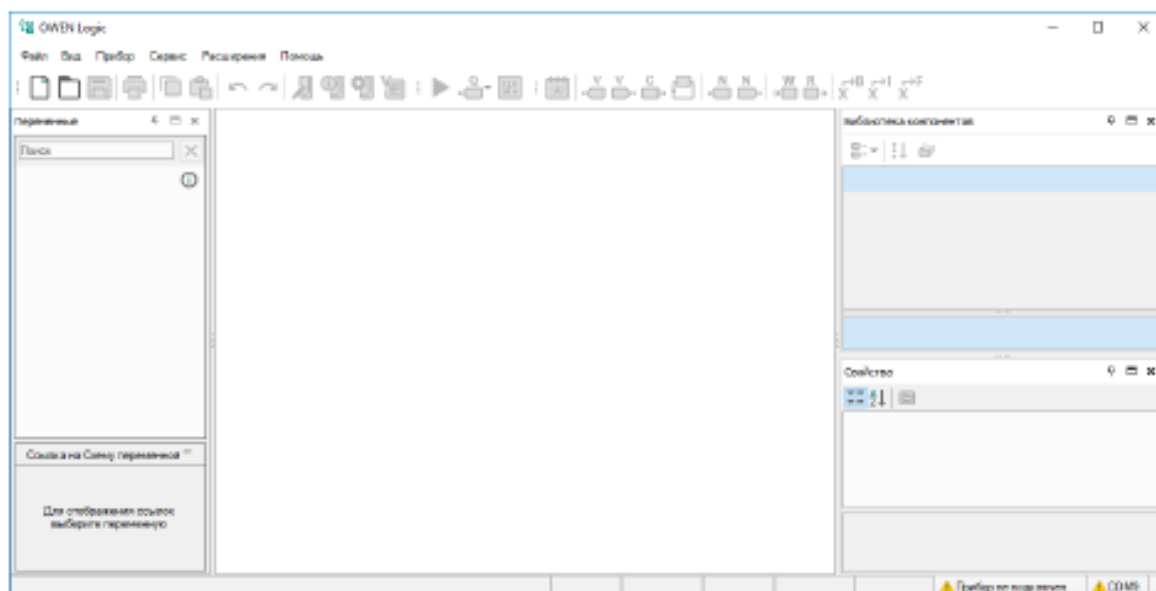


Рисунок 1 – Главное окно среды OWEN Logic

Приведем более подробное описание элементов главного окна.



Посередине окна есть поле для работы с нашими программами. Размеры данного поля можно увеличивать и уменьшать по своему желанию, а заранее расставленные значки входов и выходов того реле, для которого разрабатывается программа, перемещать по своему усмотрению так, чтобы всё было наглядно для всех пользователей, включая и тех, кто после вас будет разбираться в созданном проекте.

Сверху располагается панель инструментов или кнопок, включая и работы с переменными различного типа и преобразование типов значений. С их помощью можно вставить переменную (обычную, сетевую, из функционального блока), константу и обратную связь (задержку на один цикл программы).

Справа находятся списки всех доступных программных компонентов, функций и макросов проекта. Нужные элементы из этих списков перетаскиваются при помощи мыши на поле в схему.

Справа снизу находится окно свойств того элемента, который выбран на поле из схемы рассматриваемой программы — самое поле (можно изменить его размеры), переменная, функциональный блок, экран, группа экранов, и т.д.

Слева от поля работы с программами расположено окно доступных переменных проекта. Оно используется для того, чтобы можно было быстро выбрать необходимую переменную и переместить её в окно разработки программы в нужное место, а не активировать панель «Переменная», а потом задавать там нужную переменную. Также данное окно помогает контролировать, какие переменные уже используются (если выбрать переменную в этом окне, то снизу выводится список ссылок на неё).

В самом низу главного окна есть строка состояния, которая показывает информацию, сколько памяти занимает наша программа и переменные.

### **Порядок выполнения работы**

1. Ознакомиться с средой программирования OWEN Logic.
2. Изучить основные элементы рабочего окна среды программирования OWEN Logic.
3. Изучить назначение основных элементов, находящиеся в библиотеке.

4. Для заданного преподавателем элемента рассмотреть его свойства и изучить способы изменения этих свойств.
5. Повторить п.4 для других элементов из библиотеки.

### **Содержание отчета**

1. Название работы
2. Цель работы
3. Схемы изученных в ходе выполнения работы элементов.
4. Списки свойств для рассмотренных в ходе выполнения работы элементов.
5. Выводы по работе.

### **Контрольные вопросы**

1. Назначение языка FBD.
2. Порядок применения языка FBD.
3. Основные элементы рабочего окна среды программирования OWEN Logic
4. Как изменяются свойства элементов из библиотеки?

## ЛАБОРАТОРНАЯ РАБОТА № 3

### ОСНОВЫ ПРОГРАММИРОВАНИЯ РЕЛЕ ПР110

*Цель работы: ознакомление с основами разработки программ на языке функциональных блоков и диаграмм FBD для реле ПР110, предназначенных для создания программ автоматизации различных производственных процессов.*

#### Краткие методические указания

Работа над программой начинается с активации инструмента «Новый проект». Это приводит к отображению разделов в Библиотеке компонентов и появлению в рабочей области холста (незаполненной схемы реле с входами и выходами) для разработки программы на языке FBD – рис.1.

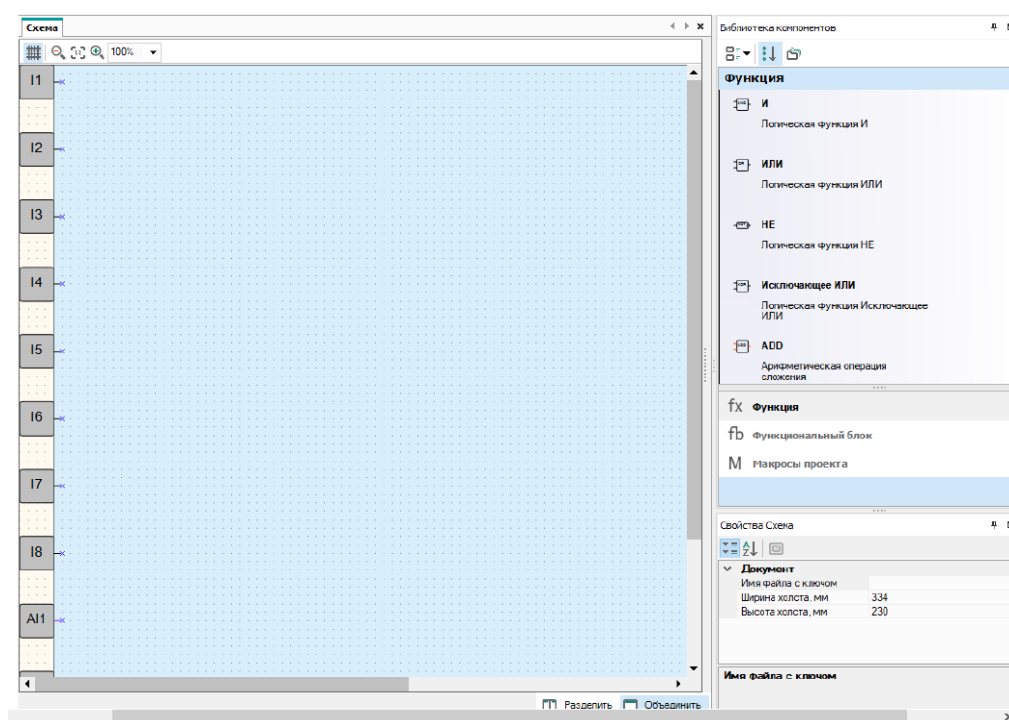


Рисунок 1 – Вид рабочей области с холстом (слева) и Библиотекой компонентов (справа)

На холсте слева находятся входы реле (I1 – I8, рассматриваем только для реле ПР110-220 с 8 входами), справа располагаются выходы Q1 – Q4 (на данном рисунке не видны).

Входы и выходы прибора обозначены в виде квадратов с маркировкой номеров:

- вдоль левого края холста расположены входы ( $I_x$  — дискретные,  $AI_x$  — аналоговые);
- вдоль правого края холста расположены выходы ( $Q_x$  — дискретные,  $AO_x$  — аналоговые,  $F_x$  — светодиоды).

Цифры в обозначениях соответствуют номерам физических входов и выходов прибора. Входы и выходы можно перетаскивать вверх/вниз вдоль холста методом drag & drop, удалять и добавлять для удобства компоновки схемы.

Программа для прибора составляется с учетом количества имеющихся у него входов, выходов и наличия часов реального времени. Функциональная схема работы реле приведена на рис.2.

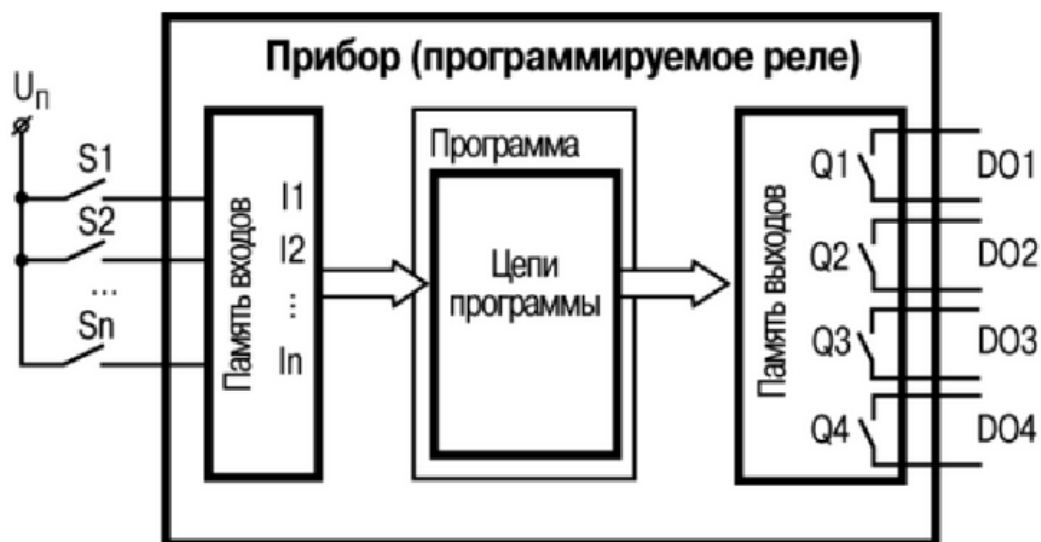


Рисунок 2 - Функциональная схема работы реле

Для создания нового проекта (программы) следует:

1. Нажать кнопку на панели инструментов или выбрать в главном меню Файл → Новый проект....

2. В открывшемся окне (рис.3) выбрать модификацию реле и нажать ОК. Меню выбора реле поддерживает фильтрацию по модели (если реле уже подключено через программатор, то оно опознается и отображается автоматически). При отсутствии подключенного реле производится выбор из меню.

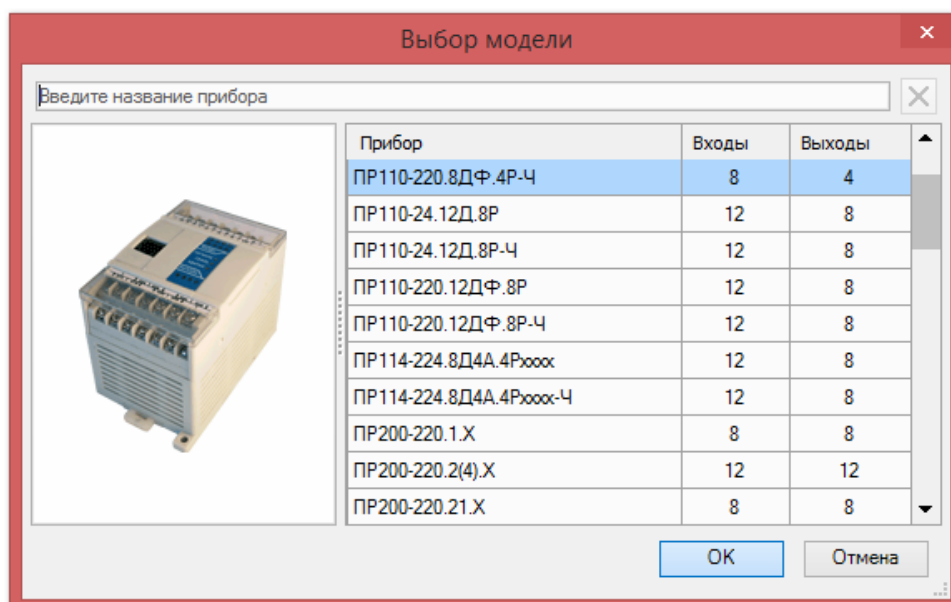


Рисунок 3 - Окно выбора модели реле (прибора)

В главном окне OWEN Logic отобразится холст проекта, станут активными строка состояния, панели Библиотека компонентов, Свойства и Переменные.

Порядок программирования реле в проекте:

1. Разработка программы.
2. Отладка программы.
3. Подключение реле к ПК.
4. Загрузка разработанной программы в реле.

Сама программа создается на холсте проекта на языке FBD с помощью блоков и элементов вставки из Библиотеки компонентов.

Разработку программы рекомендуется начать с планирования. План должен описывать все возможные состояния прибора во время функционирования (в виде диаграммы режимов, таблицы состояний, электрической или функциональной схемы и/или др.).

После того как продуманы все задачи, которые должны выполняться прибором, следует составить программу на холсте проекта с помощью элементов Библиотеки компонентов – выбирается пункт fx Функция и далее соответствующие компоненты.

Входы и выходы реле и компонентов следует соединить, проведя между ними линии связи. Для создания соединительной линии следует:

- нажать левую клавишу мыши на выход первого компонента. Линия присоединится к нему и потянется вслед за курсором мыши;
- для создания изгиба линии следует нажать на холсте для отметки точки поворота линии;
- протянуть линию до входа второго компонента и нажать по нему. Линия связи создана.

### **Порядок выполнения работы**

1. Ознакомиться с порядком выбора модели реле в среде программирования OWEN Logic.
2. Изучить основные разделы библиотеки Компонентов среды программирования OWEN Logic.
3. Изучить порядок применения элементов из библиотеки Компонентов.
4. Для заданных преподавателем схемы и элементов из библиотеки Компонентов составить блок-схему программы.
5. Повторить п.4 для других схем и элементов.

### **Содержание отчета**

1. Название работы
2. Цель работы
3. Схемы составленных программ.
4. Выводы по работе.

### **Контрольные вопросы**

1. В каком порядке составляется программа на языке FBD?
2. В каком порядке применяются элементы (компоненты)?
3. Основные правила составления схемы программы в среде программирования OWEN Logic.
4. Как изменяются свойства элементов из библиотеки

## ЛАБОРАТОРНАЯ РАБОТА № 4

### СОЗДАНИЕ ПРОГРАММ ДЛЯ РЕЛЕ ПР110 НА С ИСПОЛЬЗОВАНИЕМ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ

*Цель работы:* ознакомление с разработкой программ на языке функциональных блоков и диаграмм FBD для реле ПР110 с применением логических элементов.

#### Краткие методические указания

Рассмотрим составление программ для схемы подключения реле, приведенной на рис.1.

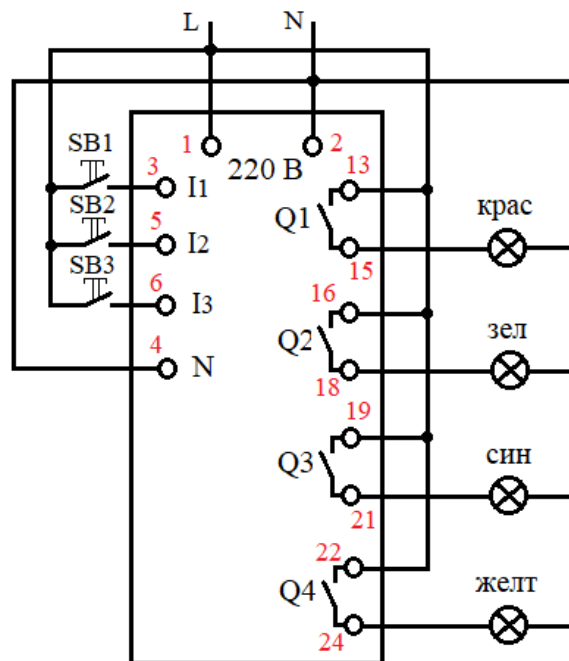


Рисунок 1 – Схема подключения реле ПР110

К трем входам реле подключены три кнопки с замыкающимися контактами без фиксации включенного положения SB1 – SB3. К четырем выходам реле (Q1 – Q4) подключены четыре сигнальные лампы (красная, зеленая, синяя и желтая) для индикации срабатывания контактов выходов реле.

Программа будет выполнять следующие функции:

- включение красной сигнальной лампы нажатием одной из кнопок SB1, SB2;

- включение зеленой сигнальной лампы нажатием сразу двух кнопок SB1 и SB2 одновременно;

- синяя сигнальная лампа должна быть включена в исходном (отключенном) состоянии кнопки SB3, ее нажатие должно отключать синюю сигнальную лампу и включать желтую.

При составлении программы в виде совокупности функциональных элементов и связей между ними следует учитывать, что они работают с двумя видами сигналов – «0» (низкий уровень) и «1» (высокий уровень). Так при нажатии любой кнопки соответствующий ей вход реле переходит из состояния «0» в состояние «1». Выходные контакты реле будут замыкаться, если на данный выход поступает сигнал «1», при сигнале «0» выход не активен – контакты разомкнуты.

Для первой функции составляем следующую таблицу состояний: красная сигнальная лампа должна загораться, если нажата ИЛИ кнопка SB1 ИЛИ SB2. Этой формулировке соответствует элемент ИЛИ (OR). Данный элемент имеет два входа и один выход.

Для второй функции программы составляем следующую таблицу состояний: зеленая сигнальная лампа должна загораться, если нажата И кнопка SB1 И SB2. Этой формулировке соответствует элемент И (AND). Данный элемент имеет два входа и один выход.

Для третьей функции программы составляем следующую таблицу состояний: синяя сигнальная лампа должна гореть, если не нажата кнопка SB3, при ее нажатии эта лампа должна погаснуть и загореться желтая. Этой формулировке соответствует элемент НЕ (NOT). Данный элемент имеет один вход и один выход. Блок используется для инвертирования сигнала.

Для реализации первой функции используем элемент ИЛИ (OR), его входы подключаются к входам реле I1 и I2 (внешние кнопки SB1 и SB2), выход подключается к выводу реле Q1 (управление красной лампой). Это показано на рис.2, на котором приведен фрагмент холста проекта с размещенным элементом ИЛИ (OR) и связями с двумя входами реле I1 и I2, к которым подключены кнопки, и выходом Q1, к которому подключена красная лампа.



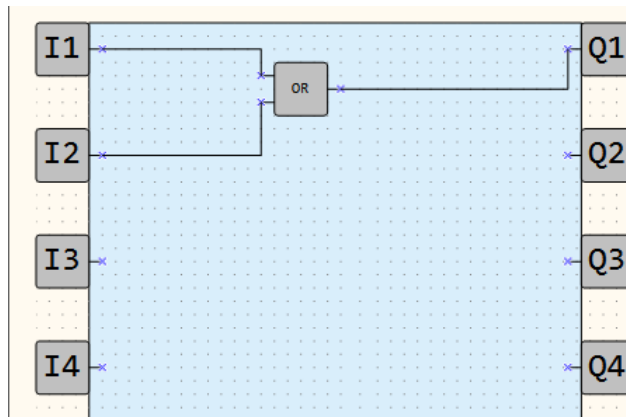


Рисунок 2 – Схема первой функции программы

Вторая функция программы обеспечивается элементом И (AND). Его вводы подключаются к вводам реле I1 и I2 (внешние кнопки SB1 и SB2), выход подключается к выводу реле Q2 (управление зеленой лампой) – рис.3.

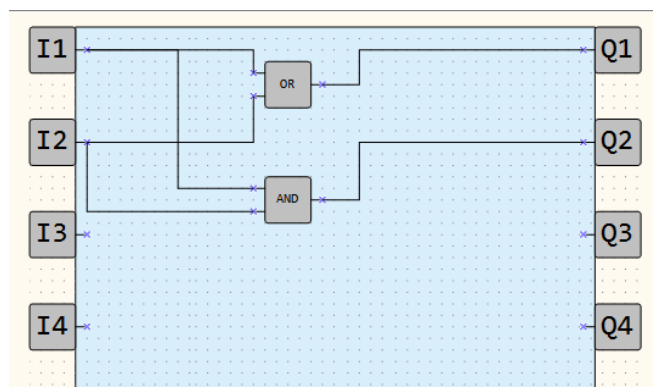


Рисунок 3 – Схема программы с добавлением второй функции

Третья функция программы создается двумя элементами НЕ (NOT). Ввод одного из них подключается к вводу I3 (внешняя кнопка SB3), выход подключается к выводу реле Q3 (управление синей лампой) и вводу второго элемента НЕ (NOT). Выход второго элемента соединяется с выводом реле Q4 (управление желтой лампой) – рис.4.

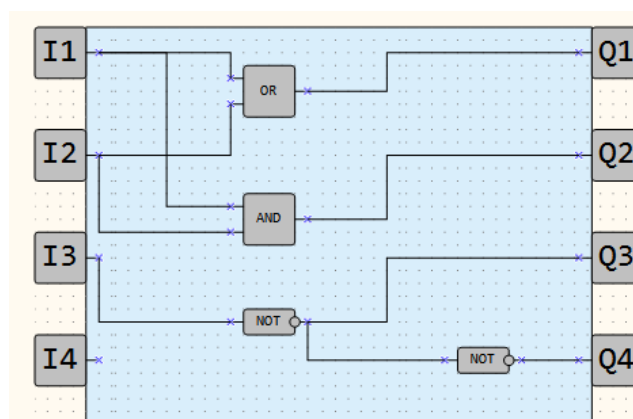


Рисунок 4 - Схема программы с добавлением третьей функции

После окончательной сборки схемы программы, ее следует сохранить, используя для этого главное меню Файл – Сохранить, или же соответствующую кнопку из панели инструментов. Если сохранение программы (проекта) осуществляется в первый раз, то на главное окно среды OwenLogic выводится окно сохранения программы, в котором задается место сохранения и имя разработанной программы (проекта).

Далее созданную программу следует записать в память реле (прибора), например, через главное меню. Выбирается пункт главного меню – Прибор и в появившемся списке активируется - Записать программу в прибор (рис.5).

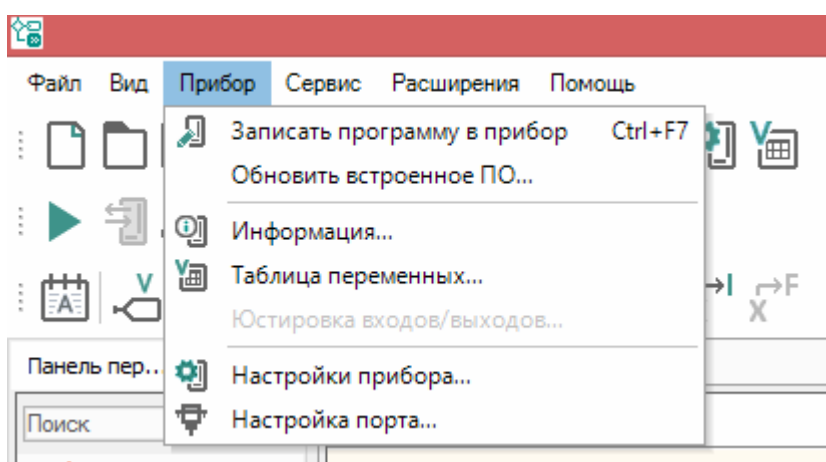


Рисунок 5 – Производство записи программы в память реле (прибора)

Во время записи программы в главном окне появляется небольшое окно, в котором находится полосовой индикатор загрузки программы, на котором после окончания процесса появляется сообщение об окончании данного процесса – рис.6.

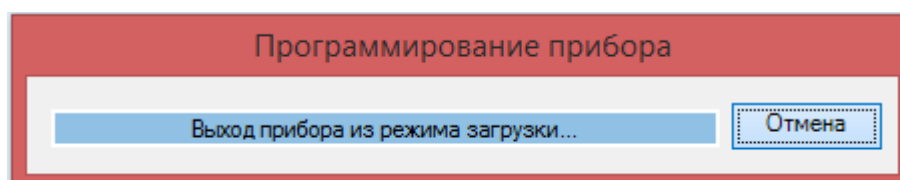


Рисунок 6 – Окно индикации записи программы

После завершения процесса записи программы реле автоматически переходит к выполнению новой, только что записанной в память программы.

### **Порядок выполнения работы**

1. Повторить приведенные схемы программы для реле ПР110.
2. Составить таблицы состояния для использованных в схеме программы логических элементов.
3. Произвести изменения в схеме программы согласно заданию преподавателя.
4. Для нового задания преподавателя составить схему программы.
5. Повторить п.3 и п.4 для других схем и элементов.

### **Содержание отчета**

1. Название работы
2. Цель работы
3. Схемы составленных программ.
4. Выводы по работе.

### **Контрольные вопросы**

1. Что такое таблицы состояния?
2. Как выполнить элемент ИЛИ с тремя входами и одним выходом?
3. Как выполнить элемент И с тремя входами и одним выходом?
4. Как выполнить элемент ИЛИ-НЕ?

## ЛАБОРАТОРНАЯ РАБОТА № 5

### СОЗДАНИЕ ПРОГРАММ ДЛЯ РЕЛЕ ПР110 НА С ИСПОЛЬЗОВАНИЕМ ТАЙМЕРОВ И ГЕНЕРАТОРОВ

*Цель работы:* ознакомление с разработкой программ на языке функциональных блоков и диаграмм FBD для реле ПР110 с применением таймеров и генераторов.

Рассмотрим теперь программу с использованием таймеров и генераторов (пункт библиотеки компонентов fb Функциональный блок). Вторая программа будет выполнять следующие функции:

- включение красной сигнальной лампы на некоторое время, определяемое пользователем при составлении программы, нажатием кнопки SB1;

- включение зеленой сигнальной лампы через некоторый промежуток времени на определенное время нажатием кнопки SB1, длительность задержки включения и включения зеленой лампы задаются пользователем при составлении программы;

- включение синей сигнальной лампы в мигающем режиме на определенный период времени нажатием кнопки SB2, параметры мерцающего режима и продолжительность этого режима устанавливаются пользователем при составлении программы;

- нажатие кнопки SB3 должно блокировать работу (включение) всех трех задействованных в предыдущих действиях сигнальных ламп.

Для программирования первой функции используем таймер (TP), формирующий импульс включения выхода на заданный интервал времени – рис.1.

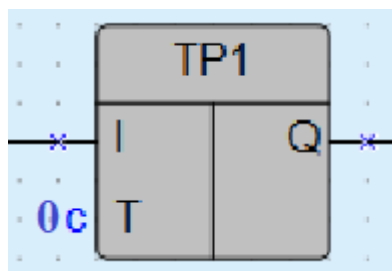


Рисунок 1 – Схема таймера, формирующий импульс включения выхода на заданный интервал времени

На вход I данного блока подается управляющий сигнал, по фронту которого (переключение из состояния «0» в состояние «1») на выходе Q блока появляется сигнал логической «1». После запуска этого таймера выход Q не реагирует на изменение значения входного сигнала в течение интервала  $T_{имп}$ . По истечении интервала  $T_{имп}$  выходной сигнал сбрасывается в логический «0».

Допустимый диапазон значений  $T_{имп}$  от 0 до 4147200000 мс или 48 дней. Длительность импульса и единицы измерения времени настраиваются на панели свойств этого блока – рис.2.

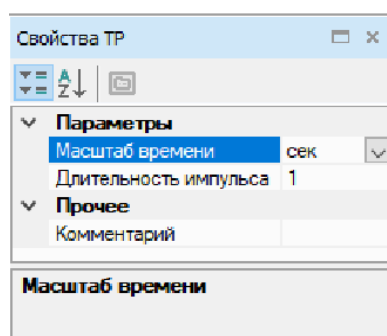


Рисунок 2 – Окно свойств блока таймер, формирующий импульс включения выхода на заданный интервал времени

Для выполнения первой функции рассматриваемой программы используем таймер TP1, на вход I которого подается сигнал от ввода реле I1. К выводу Q подключается выход реле Q1 – рис.3. Интервал времени существования сигнала логической «1» на выводе Q установлен в окне свойств блока, равным 10 с.

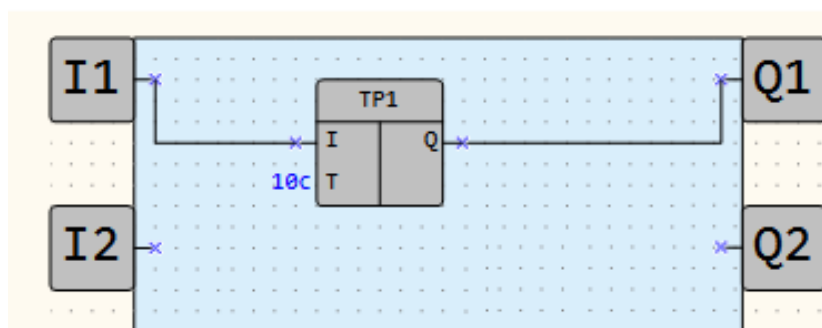


Рисунок 3 – Схема включения таймера TP

В исходном положении схемы (кнопка SB1 не нажата) сигнал на вводе I блока TP1 равен логическому «0». При нажатии кнопки

SB1 на выводе Q блока TP1 появляется сигнал логической «1», что приводит к включению вывода реле Q1, его контакт замыкается и красная сигнальная лампа включается. После истечения 10 секунд (интервал времени действия импульса), на выводе Q блока TP1 появляется сигнал логической «0», выход реле Q1 отключается, контакт размыкается, и красная сигнальная лампа гаснет.

Для выполнения второй функции используем два блока - таймер (TP) и таймер с задержкой включения (TON) используется для операции задержки передачи сигнала. На выходе Q таймера появится сигнал логической «1» с задержкой относительно фронта входного сигнала I (переключение из состояния «0» в состояние «1») продолжительностью не менее длительности  $T_{on}$  и выключится по спаду входного сигнала – рис.4.



Рисунок 4 – Таймер с задержкой включения

Допустимый диапазон значений  $T_{зад}$  от 0 до 4147200000 мс или 48 дней. Время задержки выключения и единицы измерения времени настраиваются на панели свойств этого блока.

Схема выполнения второй функции программы приведена на рис.5.

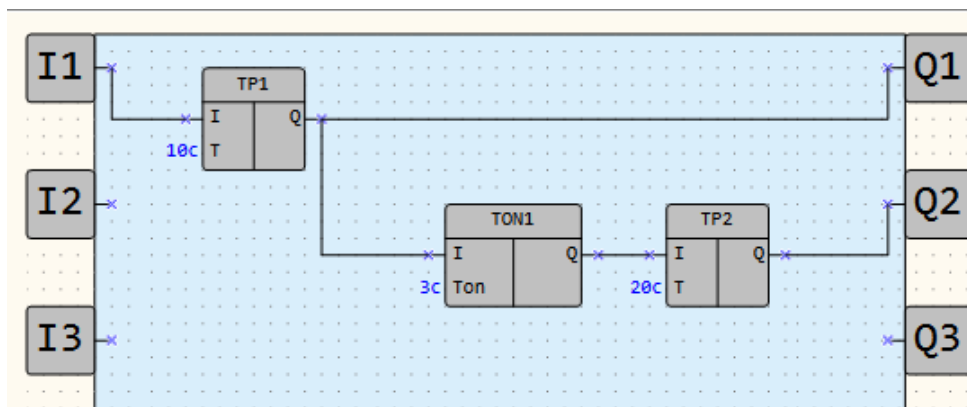


Рисунок 5 – Схема программирования второй функции второй программы

На этой схеме показаны все установленные интервалы времени. При нажатии кнопки SB1, на вход таймера TP1 подается фронт управляющего сигнала (переключение из состояния «0» в состояние «1»). По этому фронту на выводе Q данного таймера формируется импульс сигнала логической «1» длительностью 10 с – на такой период времени включится выход Q1 и загорится красная сигнальная лампа.

Через 3 с после этого на выводе таймера TON1 появится фронт управляющего сигнала (переключение из состояния «0» в состояние «1»), что приведет к срабатыванию таймера TP2 и возникновению на его выводе Q импульса сигнала логической «1» длительностью 20 с – на такой период времени включится выход Q2 и загорится зеленая сигнальная лампа.

Для программирования третьей функции (включения синей сигнальной лампы в мигающем режиме) применяем генератор прямоугольных импульсов (BLINK). Он используется для формирования прямоугольных импульсов пульсации – рис.6.

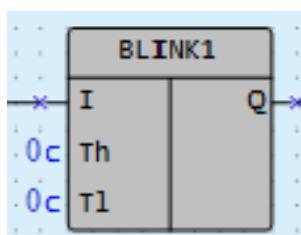


Рисунок 6 - Генератор прямоугольных импульсов

На выходе Q генератора формируются импульсы с заданными параметрами длительности включенного ( $T_{\text{вкл}}$  – сигнал логической «1») и отключенного ( $T_{\text{откл}}$  – сигнал логического «0») состояния на время действия управляющего сигнала на входе I (сигнал логической «1»).

Допустимый диапазон значений  $T_{\text{вкл}}$  и  $T_{\text{откл}}$ : от 0 до 423360000 мс или 49 дней. Длительность включенного и отключенного состояний настраиваются на панели свойств данного блока. Параметр Th соответствует  $T_{\text{вкл}}$ . Параметр Tl соответствует  $T_{\text{откл}}$ .

Схема выполнения третьей функции программы включает в себя таймер TP3 и генератор прямоугольных импульсов BLINK1 – рис.7.

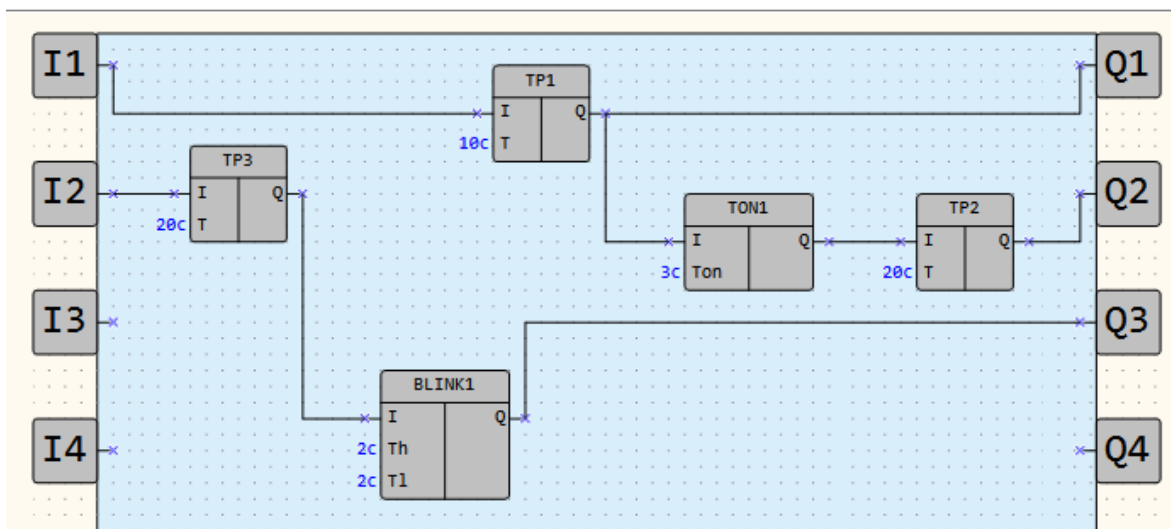


Рисунок 71 – Схема программирования третьей функции второй программы

Нажатие кнопки SB2 приводит к подаче на ввод таймера TP3 фронта управляющего сигнала (переключение из состояния «0» в состояние «1»). По этому фронту на выводе Q данного таймера формируется импульс сигнала логической «1» длительностью 20 с – на такой период времени включится в работу генератор прямоугольных импульсов BLINK1. Он генерирует прямоугольные импульсы с  $T_{\text{вкл}} = 2 \text{ с}$  (на выводе Q генератора сигнал логической «1») и  $T_{\text{откл}} = 2 \text{ с}$  (на выводе Q генератора сигнал логической «0»).

В те моменты времени, когда на выводе Q генератора существует сигнал логической «1», выход реле Q3 включается, контакт реле замыкается и загорится синяя сигнальная лампа. В другое время, когда на выводе Q генератора существует сигнал логической «0», выход реле Q3 отключается, контакт реле размыкается и синяя сигнальная лампа гаснет. После окончания периода действия импульса сигнала логической «1» длительностью 20 с, генератор прекращает свою работу, его вывод Q переключается из состояния логической «1» в состояние логического «0». Выход реле выход реле Q3 отключается и останавливает свою работу.



## **Порядок выполнения работы**

1. Повторить приведенные схемы программы для реле ПР110.
2. Составить временные диаграммы работы для использованных в схеме программы элементов.
3. Произвести изменения в схеме программы согласно заданию преподавателя.
4. Для нового задания преподавателя составить схему программы.
5. Повторить п.3 и п.4 для других схем и элементов.

## **Содержание отчета**

1. Название работы
2. Цель работы
3. Схемы составленных программ.
4. Выводы по работе.

## **Контрольные вопросы**

1. Чем отличаются элементы TON и TR?
2. По какому фронту управляющего сигнала переключается элемент TR?
3. Как изменить время существования импульса сигнала логической «1» на выходе элемент TR?
4. Назначение элемента BLINK.

## ЛАБОРАТОРНАЯ РАБОТА № 6

### СОЗДАНИЕ ПРОГРАММ ДЛЯ РЕЛЕ ПР110 НА С ИСПОЛЬЗОВАНИЕМ ТРИГГЕРОВ

*Цель работы:* ознакомление с разработкой программ на языке функциональных блоков и диаграмм FBD для реле ПР110 с применением триггеров.

Триггер представляет собой элемент, обладающий способностью длительно находиться в одном из двух возможных устойчивых состояний и менять их под воздействием внешних управляющих сигналов. Каждое состояние триггера легко определяется по значению выходного напряжения или сигнала на его выходе.

Третья программа обладает следующими функциями:

- нажатие кнопки SB1 должно обеспечивать включение красной сигнальной лампы, отключение этой лампы должно осуществляться нажатием кнопки SB3;

- нажатие кнопки SB1 должно обеспечивать включение: красной сигнальной лампы, зеленой сигнальной лампы через 5 с после нажатия кнопки, синей сигнальной лампы через 15 с после нажатия кнопки и желтой сигнальной лампы в мигающем режиме с временами включения и отключения, равными 3 с, отключение всех ламп – нажатие кнопки SB3;

- нажатие кнопки SB2 должно обеспечивать блокировку включения ламп нажатием кнопки SB1.

Для программирования первой функции реле используем RS-триггер с приоритетом выключения - рис.1.

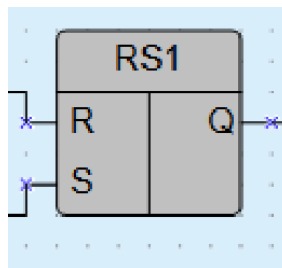


Рисунок 1 - RS-триггер с приоритетом выключения

Он используется для переключения с фиксацией состояния во время поступления коротких импульсов на соответствующий вход. На выходе Q появится логическая «1» по фронту сигнала на входе S (переключение из состояния «0» в состояние «1») - рис.2.

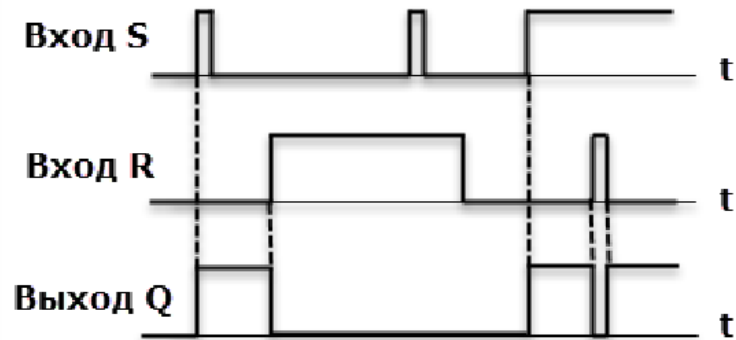


Рисунок 2 – Временная диаграмма работы RS-триггера с приоритетом выключения

В случае одновременного поступления сигналов на оба входа приоритетным является сигнал логической «1» на входе R, переключающий этот триггер в состояние с логическим «0» на выходе триггера Q.

Составим схему выполнения первой функции рассматриваемой программы. В этой схеме используем RS-триггер с приоритетом выключения - рис.3.

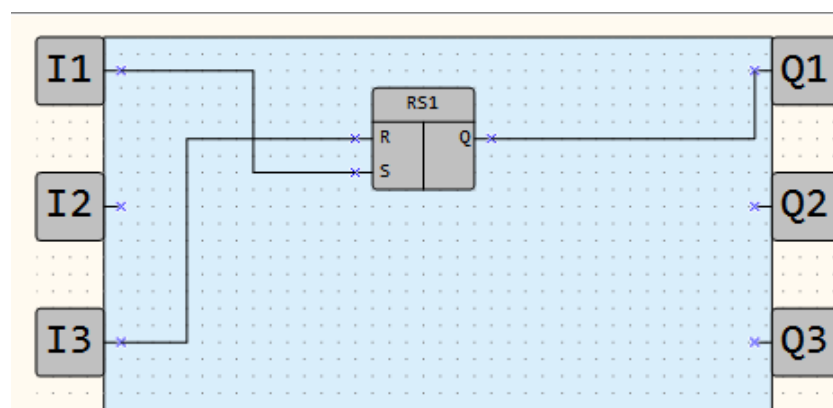


Рисунок 3 - Схема программирования первой функции третьей программы

Его вход S, обеспечивающий появление сигнала логической «1» на выходе триггера Q, подключаем к входу реле I1. Нажатие

кнопки SB1 приведет к появлению фронта управляющего сигнала (переключение из состояния «0» в состояние «1»), по этому фронту произойдет переключение триггера в устойчивое состояние с логической «1» на выходе триггера Q, это приведет к включению вывода Q1 реле, его контакт замкнется и красная сигнальная лампа загорится.

Для отключения этой лампы вход R подключается к входу реле I2. Нажатие кнопку SB3 приведет к появлению сигнала логической «1» на входе R. Поэтому произойдет переключение триггера в состояние с логическим «0» на выходе триггера Q. Вывод Q1 реле отключается, его контакт размыкается, и красная сигнальная лампа погаснет.

Для выполнения второй функции рассматриваемой программы используем таймеры TON1 и TON2, генератор прямоугольных импульсов BLINK1. Работа с этими функциональными блоками была рассмотрена выше, поэтому ее рассматривать подробно не будем и приведем ниже схему – рис.4.

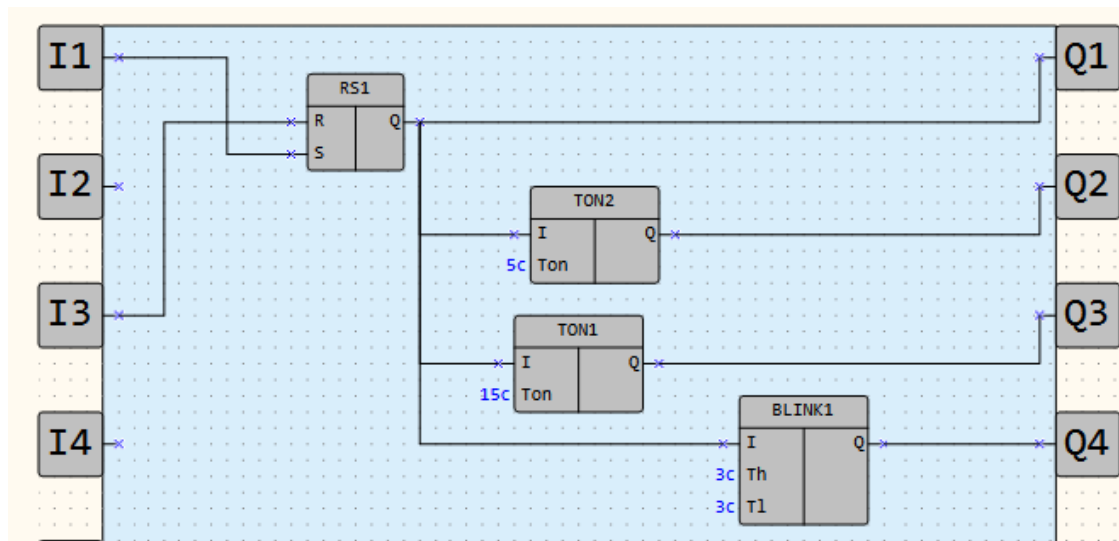


Рисунок 4 - Схема программирования второй функции третьей программы

Для блокирования включения ламп от кнопки SB1 нажатием кнопки SB2 используем блоки функций или элементы И (AND) и НЕ (NOT) – рис.5.

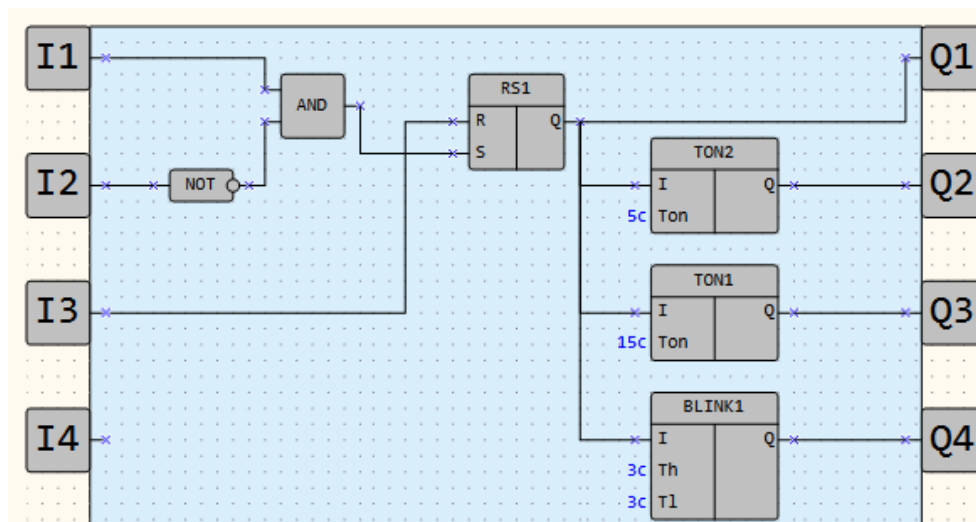


Рисунок 5 - Схема программирования третьей функции третьей программы

При не нажатой кнопке SB2, сигнал на выходе элемента НЕ (NOT) находится в состоянии логической «1». При нажатии кнопки SB1, сигнал на выходе элемента И (AND) примет состояние логической «1», так как на двух входах этого элемента одновременно действуют сигналы с состоянием логической «1». В случае нажатия и удержания кнопки SB2, сигнал на выходе элемента НЕ (NOT) примет состояние логического «0», тем самым обеспечивая блокировку элемента И (AND) в состоянии с сигналом логической «1» на его выводе. Это приведет к отсутствию фронта управляющего сигнала на входе S триггера RS1 и включения ламп не произойдет.

### Порядок выполнения работы

1. Повторить приведенные схемы программы для реле ПР110.
2. Составить таблицы состояния для использованных в схеме программы элементов.
3. Произвести изменения в схеме программы согласно заданию преподавателя.
4. Для нового задания преподавателя составить схему программы.
5. Повторить п.3 и п.4 для других схем и элементов.

## Содержание отчета

1. Название работы
2. Цель работы
3. Схемы составленных программ.
4. Выводы по работе.

## Контрольные вопросы

1. Чем отличаются триггеры с разными приоритетами?
2. Как переключается триггер RS?
3. Как применяются триггеры RS?
4. Назначение входа S триггера RS.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Юсупов, Р. Х. Основы автоматизированных систем управления технологическими процессами : учебное пособие : / Р. Х. Юсупов. – Москва ; Вологда : Инфра-Инженерия, 2018. – 133 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=493900> (дата обращения: 30.01.2022). – ISBN 978-5-9729-0229-3. – Текст: электронный.

2. Трофимов, В. Б. Интеллектуальные автоматизированные системы управления технологическими объектами: учебно-практическое пособие : учебное пособие / В. Б. Трофимов, С. М. Кулаков. – Москва ; Вологда : Инфра-Инженерия, 2016. – 232 с. : ил., табл., схем. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=444175> (дата обращения: 30.01.2022). – Библиогр.: с. 183-193. – ISBN 978-5-9729-0135-7. – Текст : электронный.

3. Глазырин, М. В. Автоматизированные системы управления тепловыми электростанциями: учебное пособие: в 2 частях : / М. В. Глазырин ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2011. – Часть 1. Основы функционирования АСУ ТП ТЭС. – 42 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=228766> (дата обращения: 30.01.2022). – ISBN 978-5-7782-1704-1. – Текст: электронный.

4. Шишов, О. В. Современные средства АСУ ТП : учебник : [16+] / О. В. Шишов. – Москва ; Вологда : Инфра-Инженерия, 2021. – 532 с. : ил., табл., схем. – Режим доступа: по подписке. – URL:

<https://biblioclub.ru/index.php?page=book&id=617234> (дата обращения: 30.01.2022). – Библиогр. в кн. – ISBN 978-5-9729-0622-2. – Текст : электронный.

5. Монтаж, наладка, эксплуатация систем автоматизации: учебное электронное издание : учебное пособие : [16+] / В. Н. Назаров, А. А. Третьяков, И. А. Елизаров, В. А. Погонин ; Тамбовский государственный технический университет. – Тамбов : Тамбовский государственный технический университет (ТГТУ), 2018. – 249 с. : табл., схем., ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=570372> (дата обращения: 30.01.2022). – Библиогр.: с. 243-244. – ISBN 978-5-8265-1932-5. – Текст : электронный.

6. Голубев, А. В. Автоматизированные информационно-управляющие системы электростанций : учебное пособие : [16+] / А. В. Голубев, И. К. Муравьев, Ю. В. Наумов. – Москва ; Вологда : Инфра-Инженерия, 2021. – 180 с. : ил., табл., схем., граф. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=617219> (дата обращения: 30.01.2022). – Библиогр.: с. 169-170. – ISBN 978-5-9729-0756-4. – Текст : электронный.

7. Валеев, И. М. Концепция управления цифровыми подстанциями будущего : учебное пособие : [16+] / И. М. Валеев, В. Г. Макаров ; Казанский национальный исследовательский технологический университет. – Казань : Казанский научно-исследовательский технологический университет (КНИТУ), 2019. – 152 с. : ил., табл., схем. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=612961> (дата обращения: 30.01.2022). – Библиогр. в кн. – ISBN 978-5-7882-2587-6. – Текст : электронный.

8. Настройка и программирование цифровых систем управления с использованием контроллеров, панелей оператора и частотных преобразователей: теория и практика : учебное пособие : [16+] / В. С. Кудряшов, А. В. Иванов, М. В. Алексеев [и др.] ; науч. ред. И. А. Хаустов ; Воронежский государственный университет инженерных технологий. – Воронеж : Воронежский государственный университет инженерных технологий, 2020. – 217 с. : ил., табл., схем., граф. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=612402> (дата обращения: 30.01.2022). – Библиогр. в кн. – ISBN 978-5-00032-459-2. – Текст : электронный.

9. Рысев, Д. В. Информационные основы диспетчерского управления в энергетике : учебное пособие : [16+] / Д. В. Рысев, П. В. Рысев, Л. Д. Федорова ; Омский государственный технический университет. – Омск : Омский государственный технический университет (ОмГТУ), 2020. – 112 с. : ил., табл., схем., граф. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=682335> (дата обращения:

30.01.2022). – Библиогр. в кн. – ISBN 978-5-8149-3032-3. – Текст :  
электронный.