



Программа OWEN EasyLogic

Руководство пользователя

Содержание

Введение	3
Аббревиатуры и термины	3
1. Общие сведения о программе	4
1.1. Назначение	4
1.2. Основные характеристики	4
1.3. Особенности коммутационной программы	4
1.4. Требования к программному обеспечению и техническим средствам	5
2. Установка и запуск программы	6
2.1. Установка	6
2.2. Запуск	6
2.3. Настройка связи с программируемым прибором	6
3. Работа с программой	7
3.1. Графический интерфейс	7
3.2. Создание нового проекта и его сохранение	9
3.3. Открытие проекта для редактирования	9
3.4. Загрузка проекта в программируемый прибор	10
3.5. Последовательность работы над проектом	10
4. Быстрый старт. Создание управляющих программ	11
4.1. Управление нагрузкой в режиме ПУСК/СТОП	11
4.2. Включатель света с автоматическим отключением	15
4.3. Автоматическое управление электромотором мешалки	18
Приложение А. Логические элементы программы	23
Повторитель сигнала	23
Функция «НЕ» (NOT)	24
Функция «И» (AND)	24
Функция «ИЛИ» (OR)	25
Функция «исключающее ИЛИ» (XOR)	25
Функция «И-НЕ» (NAND)	26
Функция «ИЛИ-НЕ» (NOR)	27
Приложение Б. Функциональные блоки программы	28
Генератор прямоугольных импульсов	28
Инкрементный счетчик с автосбросом	28
Кнопка с фиксацией включения	29
Кнопка с фиксацией отключения	29
Кнопка с задержкой включения	29
Кнопка с задержкой отключения	30
Формирователь импульса включения	30
Формирователь импульса отключения	31

Введение

Данный документ представляет собой руководство пользователя программы «OWEN EasyLogic», предназначенной для программирования работы приборов серии ПР1хх производства ПО ОВЕН.

Настоящее руководство предназначено для ознакомления пользователя с технологией программирования коммутационных приборов без применения машинных языков высокого уровня.

Программа «OWEN EasyLogic» поставляется на компакт-диске, входящем в комплект поставки программатора ПР-КП110 (последние версии доступны на сайте www.owen.ru).

Аббревиатуры и термины

Аббревиатуры и термины, используемые в тексте данного документа для компактного описания:

OWEN EasyLogic – специализированная среда программирования коммутационных и управляющих приборов, разработанная компанией «ПО ОВЕН».

ПР – программируемое реле (программируемый прибор для которого составляется проект).

ПК – персональный компьютер.

Переменная – величина, которая может использоваться для хранения в энергозависимой памяти прибора промежуточных результатов выполнения логических операций (значение логических состояний входов или выходов функциональных блоков программы). Каждая переменная имеет идентификатор (например, V1, ..., Vn).

Прибор – программируемое устройство, для которого составляется коммутационная программа, например, программируемое реле ПР110.

Проект – коммутационная (управляющая) программа, созданная пользователем в среде «OWEN EasyLogic» для последующей записи в программируемый прибор.

Преобразователь – устройство, через которое подключается программируемое реле к ПК. Схема подключения преобразователя к ПР приведена в руководстве по эксплуатации на прибор.

Коммутационная программа – созданная пользователем программа, записываемая в ПР, предназначенная для управления внешними устройствами.

Логический сигнал (или просто «сигнал») – дискретная физическая величина (напряжение или ток), принимающая только два значения: включено – соответствует логической «1» (**лог. «1»**), и отключено – логический «0» (**лог. «0»**).

Цикл – время рабочего цикла прибора (зависит от количества выполняемых операций в программных цепях).

1. Общие сведения о программе

1.1. Назначение

Программа «OWEN EasyLogic» (далее по тексту – «программа» или «среда программирования») – является прикладным программным обеспечением, предназначенным для создания алгоритма работы приборов, относящихся к группе программируемых реле. Эти приборы применяются для построения автоматизированных систем управления, а также для замены релейных систем защиты и контроля. При использовании программируемых приборов потребуются меньше переключающих устройств для решения ряда задач малой автоматизации, что снижает затраты на проектирование и изготовление систем, а также повышает их надежность.

Перечень приборов, для программирования которых может использоваться программа, доступен на сайте www.owen.ru.

1.2. Основные характеристики

Программа «OWEN EasyLogic» позволяет сформировать схему автоматизации – соответствующую конкретному комплексу процедур, управление которыми должен выполнять программируемый прибор при подаче сигналов на его входы.

Управление внешними устройствами выполняют коммутационные элементы на выходах прибора по загруженной в его энергонезависимую память программе.

Управляющая программа для прибора, создаваемая пользователем, называется «проектом».

Одновременно среда программирования может работать только с одним проектом и одним программируемым прибором. При создании проекта не требуется подключения прибора.

Последовательность операций при работе следующая:

- 1) установка среды программирования на ПК;
- 2) запуск среды программирования;
- 3) создание нового проекта или запуск существующего для редактирования;
- 4) сохранение проекта в виде файла;
- 5) загрузка проекта в программируемый прибор.

В программе используются принципы функционирования релейной автоматики, в которой схемы выполняются на основе коммутационных элементов.

В создаваемой при помощи программы схеме можно использовать логические операторы (И, ИЛИ и другие), а также специальные функциональные блоки (их перечень приведен в приложениях А и Б).

1.3. Особенности коммутационной программы

Коммутационная (управляющая) программа составляется для ПР с учетом количества имеющихся у него входов и выходов. Общая структура для таких программ показана на рисунке 1.1.

Работу ПР можно представить в виде последовательно выполняемых шагов:

Шаг 1 – логическое состояние входов автоматически записывается в ячейки памяти входов (количество ячеек равно числу входов, например, I1, ..., I8).

Шаг 2 – коммутационная программа, записанная в виде последовательности **программных цепей**, считывает значения из ячеек памяти входов и выполняет над ними логические операции, указанные для цепи 1.

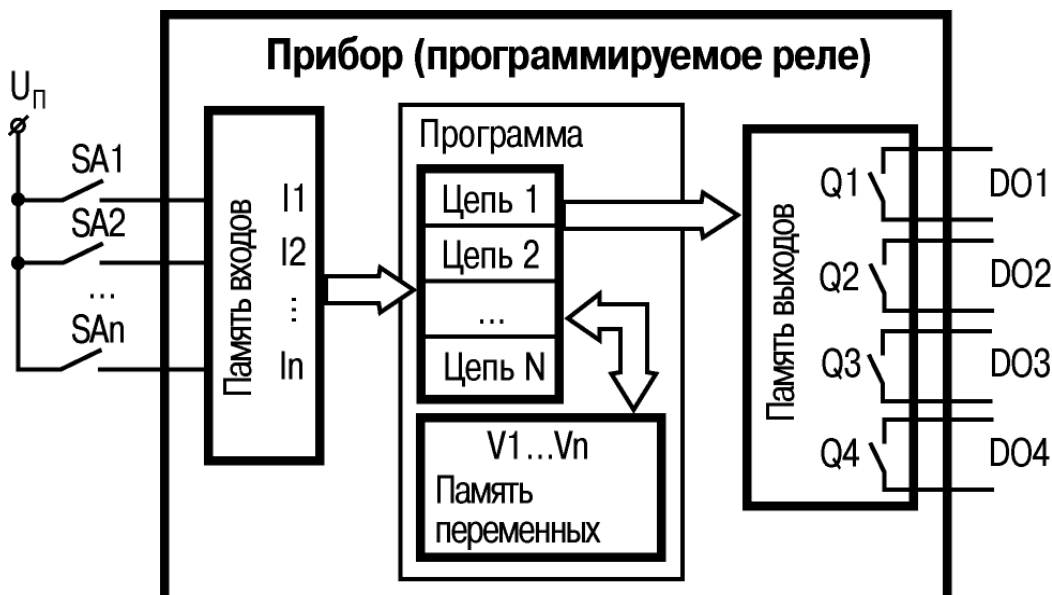


Рисунок 1.1. Функциональная схема работы программируемого прибора

Шаг 3 – повторяются действия, указанные в шаге 2, но для второй цепи, и так до обработки всех остальных программных цепей.

Шаг 4 – после обработки всех программных цепей, производится запись результатов на выходы.

Каждая программная цепь завершается одним из двух логических действий: выдачей управляющего сигнала на физический выход ПР (например, включением контактов выходных реле Q1, ..., Q4) или передачей его в ячейку памяти (V1, ..., Vn), называемую «переменной». Значение переменной может использоваться в качестве входного сигнала для любых последующих цепей.

Внимание! Следует учитывать, что логические операции всегда выполняются последовательно по порядку расположения программных цепей. Это значит, что если несколько цепей используют один и тот же выход, то на выходе будет выведен результат действий в последней цепи.

Время выполнения всех шагов (с 1 по 4) называется рабочим циклом прибора (зависит от количества выполняемых операций в программных цепях).

Шаг N – переход на Шаг 1 (после выполнения всех предыдущих шагов обработки программных цепей, цикл работы ПР повторяется с первого шага). Скорость повторения определяется внутренней тактовой частотой генератора ПР.

1.4. Требования к программному обеспечению и техническим средствам

Программа «OWEN EasyLogic» работает на ПК под управлением операционных систем MS Windows 98/2000/XP. Для ее работы должна быть предустановлена среда .NET Framework 2.0 (при отсутствии приложения запрос на его установку появляется автоматически).

Минимальные технические требования к ПК:

- процессор Pentium 2 и выше;
- оперативная память 64 Мб и более;
- минимальное свободное пространство на жестком диске 30 Мб;
- оптический привод CD-ROM;
- стандартный COM-порт;

- клавиатура и мышь;
- монитор с разрешением экрана 800х600 пикселей.

Примечание. COM-порт используется для записи созданного пользователем проекта в прибор. Если на используемом компьютере есть только интерфейс USB, то для подключения кабеля к этому порту нужен преобразователь интерфейсов (USB/RS-232) с соответствующим драйвером. При установке драйвера для преобразователя необходимо следовать указаниям, появляющимся на экране.

Для исключения сбоев при записи проекта компьютер и программируемый прибор должны иметь бесперебойное питание.

2. Установка и запуск программы

2.1. Установка

Для установки на компьютер программы «OWEN EasyLogic» следует запустить файл «Setup EasyLogic.exe» (находится на компакт-диске) и следовать инструкциям, появляющимся на экране.

После успешной установки на рабочем столе появится иконка «OWEN EasyLogic» (рисунок 2.1).



Рисунок 2.1. Иконка программы

2.2. Запуск

Для запуска программы следует выбрать команду «Пуск ▶ Программы ▶ OWEN ▶ OWEN EasyLogic ▶ OWEN EasyLogic» или щелкнуть левой кнопкой мыши на соответствующей иконке.

2.3. Настройка связи с программируемым прибором

Для настройки связи с ПР сначала выполняется подключение его к компьютеру через любой свободный COM-порт. Для соединения используется кабель программирования, входящий в комплект поставки преобразователя ПР-КП10.

Внимание! Электрическое подключение к компьютеру допускается выполнять только при выключенном питании ПР.

После электрического соединения устройств, настраиваются параметры связи в программе.

Настройка канала соединения с ПР производится в окне «Настройка порта» – оно вызывается из основного меню командой

Сервис ▶ Настройка порта....

Первоначально это окно имеет вид, показанный на рисунке 2.2.

Настройка состоит в выборе номера используемого COM-порта (остальные настройки фиксированные и выводятся для справки).

Выбор подтверждается нажатием кнопки ОК.

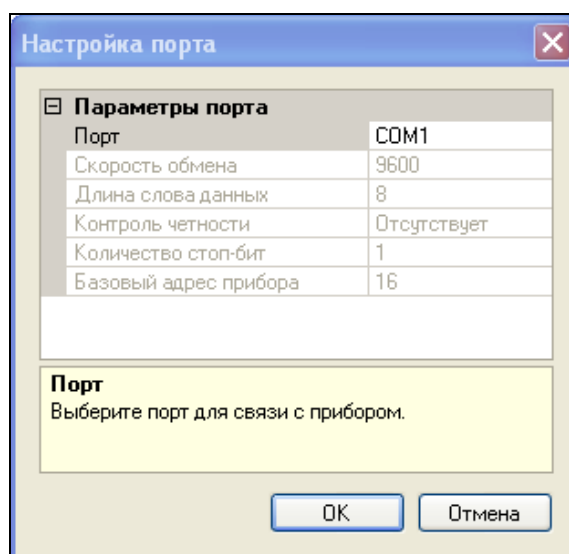


Рисунок 2.2. Окно настройки порта

3. Работа с программой

3.1. Графический интерфейс

После запуска программы «OWEN EasyLogic» на мониторе открывается окно, приведенное на рисунке 3.1.

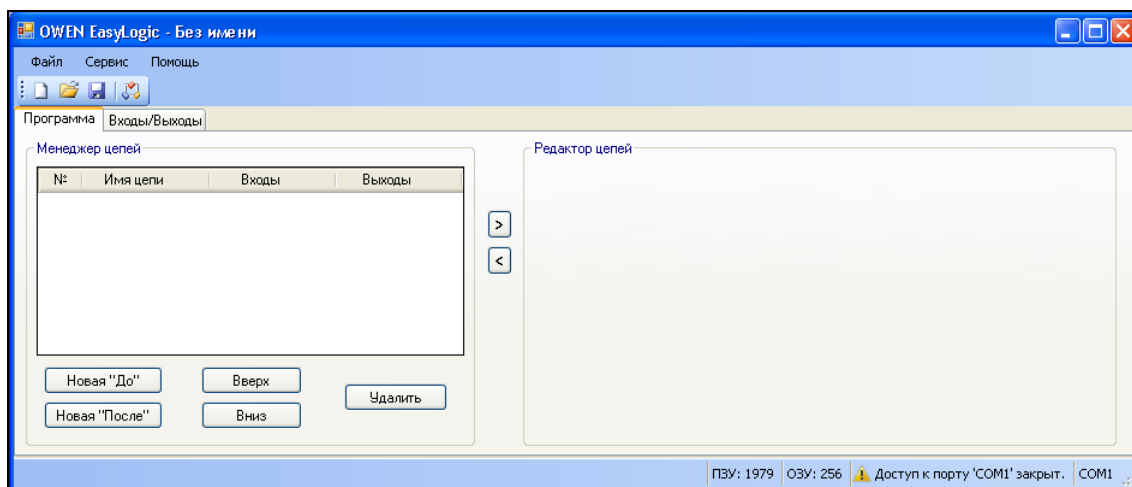


Рисунок 3.1. Главное окно программы на закладке «Программа»

Главное окно программы включает:

- заголовок окна (верхняя строка), содержащий информацию об имени файла проекта;
- основное меню: **Файл, Сервис, Помощь**;
- панели инструментов для быстрого вызова часто используемых действий:
 - создание нового проекта (📁);
 - поиск и открытие ранее сохраненного проекта (📁);
 - сохранение настоящего проекта (💾);
 - запись программы проекта в прибор (🔌);
- рабочую область, отображающую структуры текущего проекта с закладками: Программа, Входы/Выходы (на каждой закладке имеются элементы управления);
- нижнюю статусную строку, показывающую состояние программы и наличие подключения к ней программируемого устройства (рисунок 3.2).

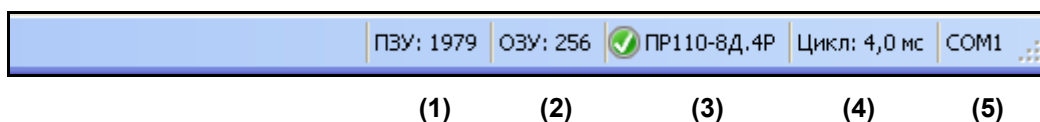


Рисунок 3.2. Статусная строка окна программы

В строке статуса отображается информация:

- (1) и (2) – доступная память прибора: «ПЗУ: 1979» и «ОЗУ: 256». Программа «OWEN EasyLogic» автоматически рассчитывает доступную память прибора, и, в случае критического значения, выводит предупреждение;
- (3) – информация о наличии связи между программой «OWEN EasyLogic» и ПР, при наличии связи указывается тип подключенного программируемого реле;
- (4) – ячейка появляется только при установлении связи с программируемым прибором и информирует о длительности рабочего цикла ПР (рассчитывается и передается в среду программирования прибором ПР);
- (5) – номер выбранного пользователем порта.

Рабочая область вкладки «Программа» поделена на две части: «Менеджер цепей» и «Редактор цепей».

В первой части ведутся работы с цепями:

- создание цепи до или после выделенной курсором;
- удаление цепи;
- переименование цепи;
- изменение порядка расположения цепей по списку.

Для создания новой цепи следует нажать кнопку или .

Для удаления выделенной курсором цепи следует нажать кнопку .

Вторая часть предназначена для ввода и редактирования коммутационной программы (рисунок 3.3).

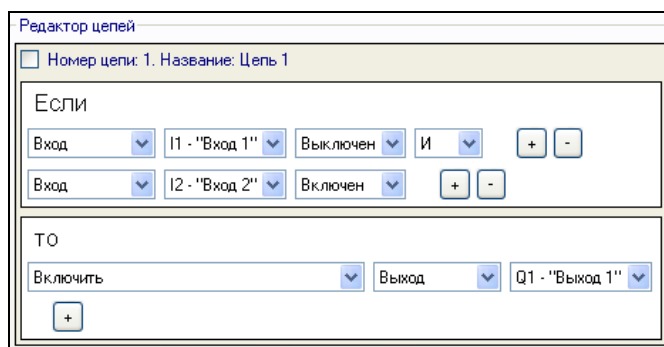


Рисунок 3.3. Создание цепи программы

Для редактирования при помощи курсора мыши пользователь выбирает необходимую цепь и в соответствующем окне (в раскрывающихся списках) нужный параметр или вводит его значение при помощи клавиатуры. В этом же окне производится присвоение (или изменение) названия входа/выхода.

Для каждой цепи есть сектор условий (Если), в котором выбираются номера физических входов (добавляется нужное количество входов при помощи кнопки) и выбирается тип логических действий с входными сигналами:

И – логическая операция «И», **ИЛИ** – логическая операция «ИЛИ».

В секторе действий (то) указывается результат выполнения условий, описанных в секторе «Если», при котором выход (номер его выбирается) должен изменить свое состояние. При этом в качестве выхода может использоваться переменная, имеющая свой идентификатор. Эта же переменная может также использоваться и в качестве источника входного логического сигнала для любых последующих цепей программы.

Результатом выполнение блока условий может быть не только одно действие, но и целый ряд действий, например, добавляя при помощи кнопки логические операции, можно включить сразу нужное число выходов.

Рабочая область вкладки «Входы/выходы» поделена на три зоны: «Переменные», «Входы» и «Выходы», в каждой из которых, при необходимости, можно добавить комментарии, поясняющий назначение сигнала (рисунок 3.4).

В зоне «Переменные» при помощи кнопки «Новая» вводятся переменные, необходимые для использования в проекте. Основные действия над переменными: создание, переименование, удаление.

По умолчанию созданной переменной дается имя «Переменная X», где X – порядковый номер. Для удобства использования имеется возможность переименовать переменную.

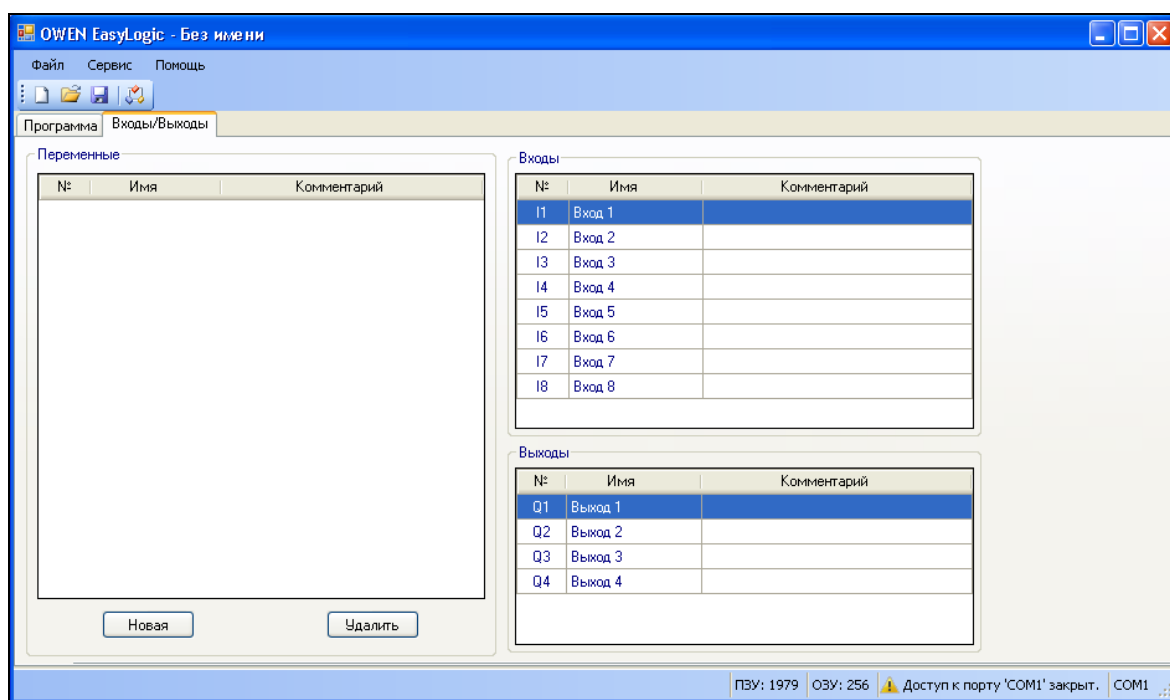


Рисунок 3.4. Окно программы на закладке «Входы/Выходы»

В зоне «Входы» и «Выходы» пользователь может изменять имена (названия) Входов/Выходов и записывать поясняющие работу программы комментарии.

3.2. Создание нового проекта и его сохранение

При запуске программы «OWEN EasyLogic» по умолчанию открывается временный файл «Без имени.xml». Для присвоения проекту идентификационного имени следует нажать кнопку «Сохранить проект» (📁) панели инструментов, или выбрать команду «**Файл** ▶ **Сохранить как...**» – откроется окно «Сохранить как» в котором можно присвоить имя файлу (при этом допустимо использовать русские буквы).

При первом сохранении или при переименовании файла проекта – в открывшемся стандартном окне выбора файла – выбрать либо задать наименование и расположение файла. Расширение файла проекта: «*.xml». Таким образом, название файла проекта выглядит, например, так:

«Автомат-1.xml».


3.3. Открытие проекта для редактирования

Для того чтобы внести изменения в проект, ранее сохраненный в файл, этот файл следует открыть из программы «OWEN EasyLogic».

Для того чтобы открыть требуемый файл, следует нажать кнопку «Открыть проект» (📁) панели инструментов, или выбрать команду «**Файл** ▶ **Открыть проект**» главного меню. В открывшемся стандартном окне выбора файла – выбрать требуемый файл. Текущий проект будет закрыт, а в окне configurатора – открыт выбранный проект.

3.4. Загрузка проекта в программируемый прибор

После создания проекта его записывают в энергонезависимую память ПР. Для записи необходимо:

- 1) соединить ПР с COM-портом компьютера при помощи коммуникационного кабеля;
- 2) подать питание на ПР согласно его руководству по эксплуатации;
- 3) настроить параметры соединения (см. п. 2.3);
- 4) записать проект в ПР при помощи кнопки  панели инструментов, или выбрать команду «Сервис ▶ Запись ПК > Прибор...».

Примечание. Сразу после записи ПР переходит в рабочий режим и коммутационная программа запускается автоматически.

Внимание! Если в подключенном приборе уже есть ранее записанная коммутационная программа, то при записи в него нового проекта она заменяется.

3.5. Последовательность работы над проектом

Составление управляющей программы рекомендуется начинать с планирования. План должен описывать все возможные состояния ПР при функционировании (в виде диаграммы режимов, таблицы состояний, электрической или функциональной схемы и/или др.).

После того как продуманы все задачи, которые должны выполняться, необходимо составить коммутационную программу на основе логических элементов и функциональных блоков, приведенных в приложении А и Б.

Работа над проектом включает:

- 1) создание (открытие) проекта – весь проект будет храниться в одном файле, которому следует присвоить идентификационное имя;
- 2) формирование структуры текущего проекта (добавление или удаление программных цепей);
- 3) загрузка проекта в прибор (ПР) и проверка его работы на макете. При проверке правильности работы программы в приборе последовательно включают и выключают каждый вход, контролируя состояние выходов на соответствие нужным условиям;

Примечание. На компьютере не предусмотрен режим эмуляции работы коммутационной программы с визуализацией состояний.

4) по результатам макетирования – редактирование проекта с целью устранения ошибок (добавление новых элементов и программных цепей, или их удаление, редактирование параметров работы функциональных блоков и т. д.);

5) загрузка проекта в прибор и проверка его работы на макете;

6) после устранения всех ошибок подготовка проекта завершается сохранением его в файле.

Процесс создания коммутационной управляющей программы иллюстрируется на конкретных примерах (см. раздел 4).

4. Быстрый старт. Создание управляющих программ

В данном разделе на примере простых задач поясняется, каким образом составляются управляющие коммутационные программы в среде «OWEN EasyLogic». Все приведенные примеры имеются на компакт-диске (в папке «Примеры для ПР110»), поставляемом со средой программирования.

4.1. Управление нагрузкой в режиме ПУСК/СТОП

Довольно часто возникает необходимость дистанционно управлять подачей напряжения в нагрузку от кнопки ПУСК (SB3), а выключать от кнопок СТОП (SB2, SB1). Нагрузка должна отключаться при срабатывании датчика (F1) или при временном отключении питающего напряжения. В качестве нагрузки (Rн) может использоваться, например, электромотор вентилятора, нагреватель или любые другие потребители энергии.

Варианты реализации данной задачи

В релейной автоматике такое устройство обычно выполняется на основе типового кнопочного поста ПУСК/СТОП (триггера на реле). Электрическая схема, в зависимости от выбранной элементной базы, может иметь вид, показанный на рисунках 4.1 и 4.2 (SB1–SB3 – кнопки без фиксации; K1 – электромагнитное реле; F1 – датчик состояния нагрузки, например, срабатывающий при перегреве мотора; G1, G2 – источники питания).

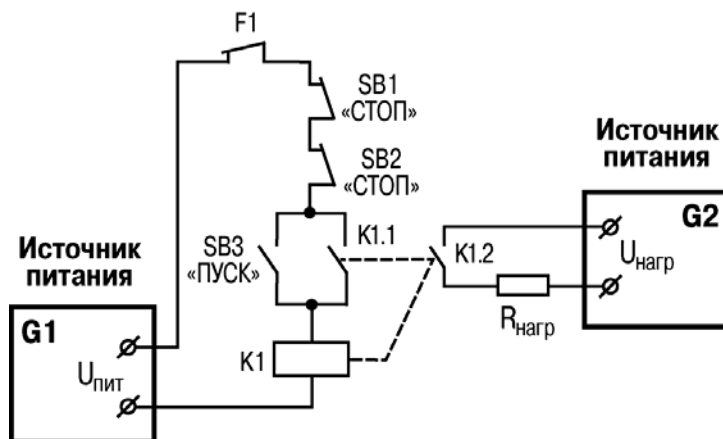


Рисунок 4.1. Электрическая схема на коммутационных элементах релейной автоматики

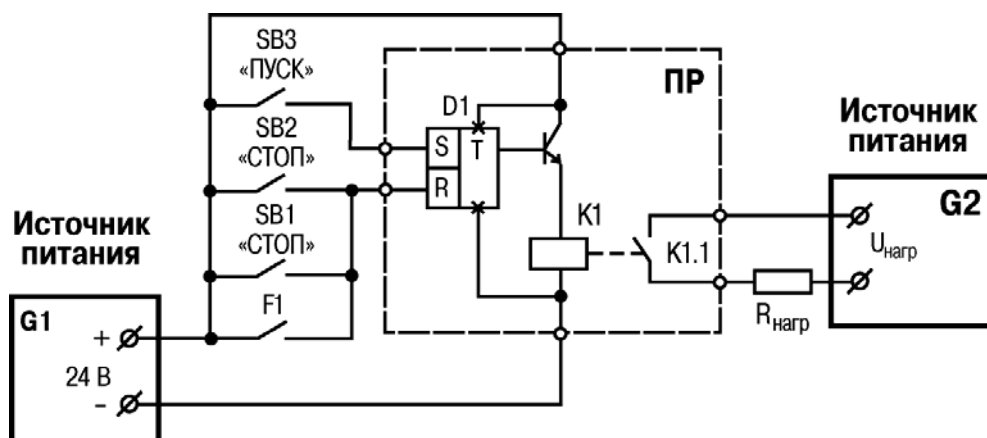


Рисунок 4.2. Электрическая схема на логических функциональных элементах, D1 – RS-триггер, SB1–SB3 – кнопки без фиксации

В этих схемах, за счет имеющейся развязки между цепями управления и выходом, нагрузка ($R_{нагр}$) может питаться напряжением любого вида (постоянным или переменным).

Для составления управляющей программы более удобна схема, приведенная на рисунке 4.2, так как в ней применяются все кнопки одного типа (нормально разомкнутые). Описание работы данной схемы:

- 1) при нажатии кнопки ПУСК (SB3) на входе S (D1) появляется лог. «1», при этом на выходе RS-триггера установится лог. «1» – включатся контакты реле K1;
- 2) при нажатии кнопки СТОП (SB1 или SB2) или замыкании контактов датчика F1, на входе R (D1) появится лог. «1» – на выходе RS-триггера установится лог. «0» – выходные контакты реле K1отключатся.

Примечание. Следует отметить, что RS-триггер возвращается в исходное состояние (лог. «0» на выходе) при любом состоянии входа S – сигнал на входе R имеет приоритет.

Данное описание можно представить в виде логических условий, приведенных в таблице 4.1.

Таблица 4.1

Цепь 1	
Если	кнопка SB3 включена И кнопка SB1* отключена
то	включить выход Q.
Цепь 2	
Если	кнопка SB1* включена
то	отключить выход Q.
* Так как кнопка SB2 и контакты датчика F1 подключены к одному входу параллельно с кнопкой SB1, отдельно описывать для них действий не требуется (логика работы ИЛИ реализована за счет параллельного электрического соединения).	

Это же описание, но для конкретного ПР будет иметь подобный вид, с несущественными отличиями, касающимися наименований используемых в схеме подключений входов и выходов.

Использование программируемого реле

Программируемое реле может применяться с двумя входами и одним выходом. Исходя из необходимого числа входов и выходов, выбирается тип программируемого прибора.

При реализации данной задачи на основе ПР110-24.8Д.4Р, можно воспользоваться схемой подключения, приведенной на рисунке 4.3.

Примечание. Номера входов и выходов ПР выбраны произвольно – именно для этих цепей будет составляться коммутационная программа.

В среде «OWEN EasyLogic» коммутационная программа будет иметь вид, приведенный на рисунке 4.4.

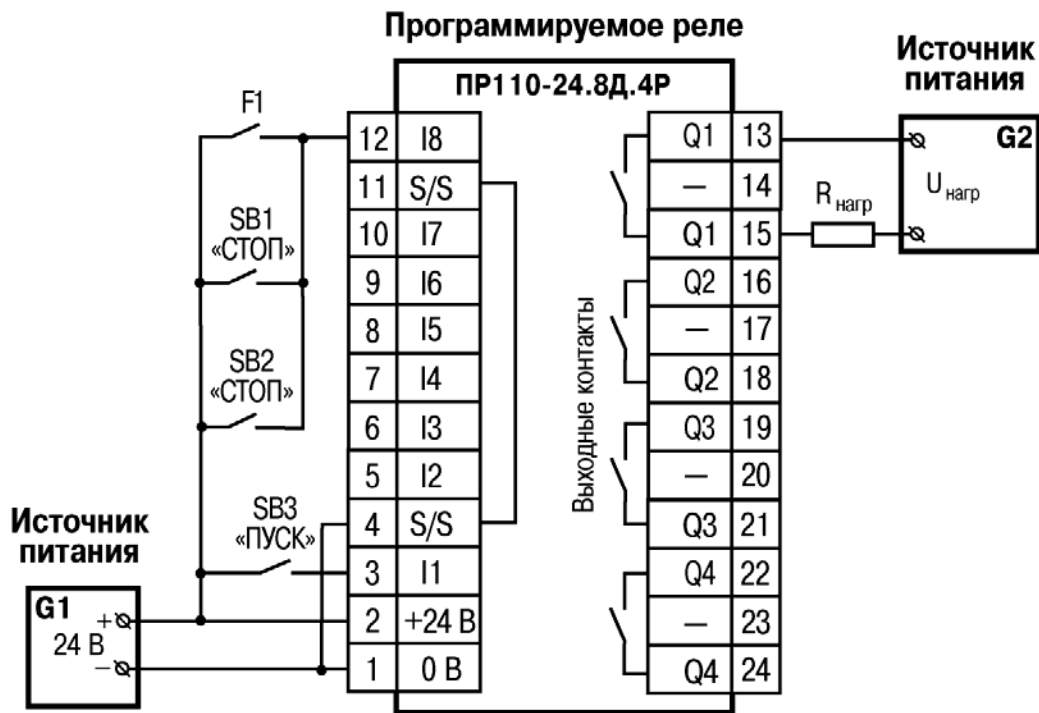


Рисунок 4.3. Электрическая схема подключения программируемого реле, SB1–SB3 – кнопки без фиксации

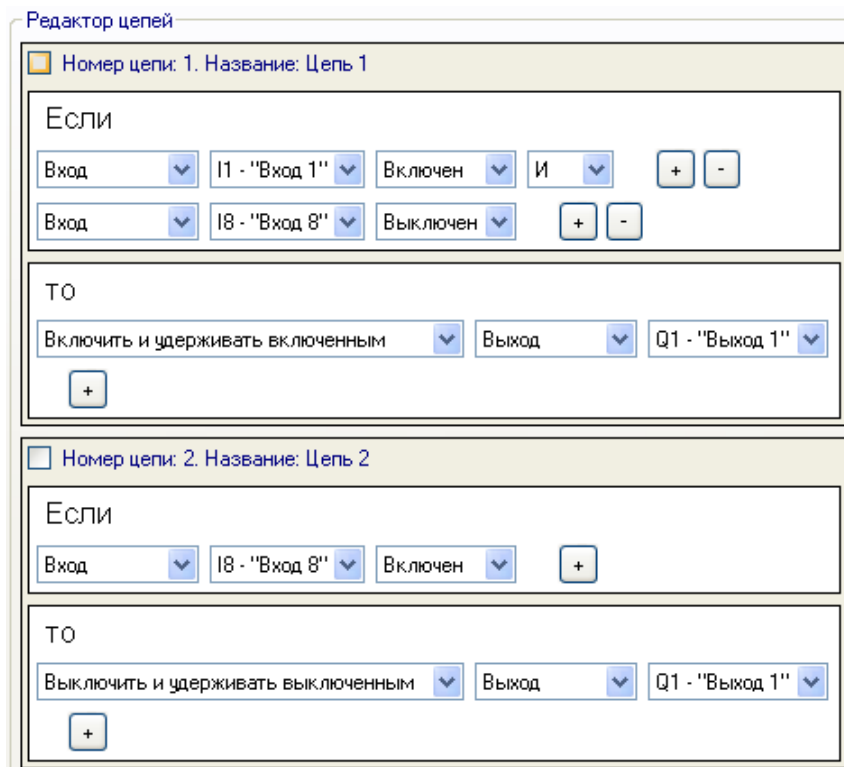

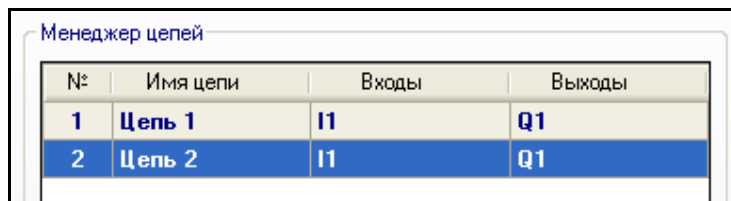


Рисунок 4.4. Управляющая коммутационная программа «ПУСК-СТОП»

Последовательность действий для ввода коммутационной программы (пошаговая инструкция) следующая:


- 1) **открытому проекту присваивается имя:** в основном меню выбирается команда «**Файл** ▶ **Сохранить как...**» – откроется окно «Сохранить как» в котором можно присвоить имя файлу, например «ПУСК-СТОП.xml»;

2) добавляются в проект две цепи (Цепь 1 и Цепь 2): на закладке «Программа» в части «Менеджер цепей» при помощи кнопки  (рисунок 4.5);



№	Имя цепи	Входы	Выходы
1	Цепь 1	I1	Q1
2	Цепь 2	I1	Q1

Рисунок 4.5. Вид окна с добавленными цепями программы

3) добавляется дополнительный вход для программной цепи 1: в части «Редактор цепей» в секторе условий (Если) при помощи кнопки ;

4) устанавливаются параметры дополнительного входа:

– выбирается используемый вход: **I8 – «Вход 8»** (в секторе условий (Если) в окне выбора номера входа, при помощи контекстного меню, показанного на рисунке 4.6). Номер входа должен соответствовать схеме, приведенной на рисунке 4.3;

– указывается состояние входа: **Выключен** (в окне выбора состояния входа при помощи контекстного меню);

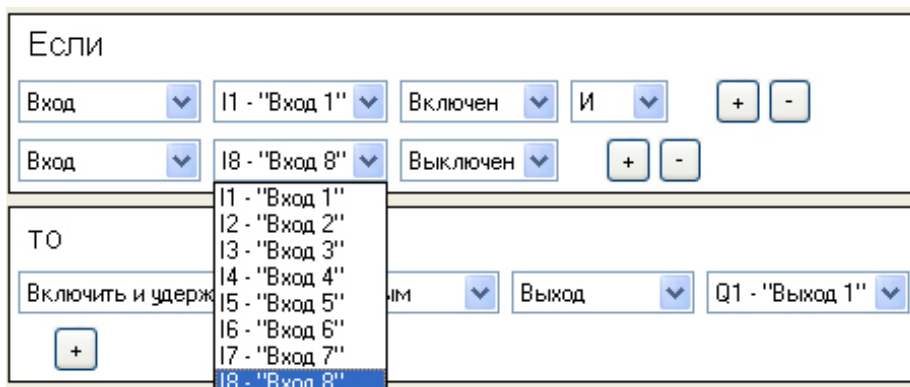


Рисунок 4.6. Вид окна с открытой вкладкой выбора номера входа

5) выбирается логическая операция с входными сигналами цепи 1: операция **И** (в данном случае она установлена по умолчанию в секторе условий (Если));

6) указывается нужное состояние выхода программной цепи 1: **Включить и удерживать включенным** – в секторе действий (то) выбирается из контекстного меню;

7) устанавливаются параметры входа для программной цепи 2:

– выбирается используемый вход: **I8 – «Вход 8»** (в секторе условий (Если) в окне выбора номера входа, при помощи контекстного меню);

– указывается состояние входа: **Включен** (в окне выбора состояния входа при помощи контекстного меню);

8) указывается нужное состояние выхода программной цепи 2:

Выключить и удерживать выключенным – в секторе действий (то) выбирается из контекстного меню;

9) сохранение проекта:

выполняется кнопкой  на панели инструментов.

4.2. Включатель света с автоматическим отключением

В подсобных помещениях и коридорах свет нужен ограниченное время. После включения освещения, его нередко забывают отключить, что приводит к излишнему расходу электроэнергии.

Требуется обеспечить включение света на заданный интервал времени, например, у входной двери в квартиру, по следующему алгоритму:

- 1) перед входной дверью установлен датчик освещения (F1) и кнопка включения света ТАЙМЕР (SB1);
- 2) при кратковременном нажатии на кнопку ТАЙМЕР, при недостаточном естественном освещении, светильник должен включаться на интервал времени 1 мин – этого времени достаточно чтобы найти замочную скважину и ключом открыть дверь;
- 3) при удерживании нажатой кнопки ТАЙМЕР в течение 2 с светильник должен включаться на интервал времени 3 мин независимо от внешнего освещения – этот режим может потребоваться при уборке коридора;
- 4) предусмотреть возможность управления работой светильника по командам от внешних управляющих устройств или при помощи выключателя СВЕТ (SA1), независимо от внешнего освещения. Этот режим удобен во время приема гостей или для дальнейшей автоматизации квартиры в рамках программы «умный дом».

Использование программируемого реле

Для реализации данной задачи управляющий автомат должен иметь три входа и один выход.

При выполнении автомата на основе ПР110-24.8Д.4Р можно воспользоваться схемой, приведенной на рисунке 4.7.

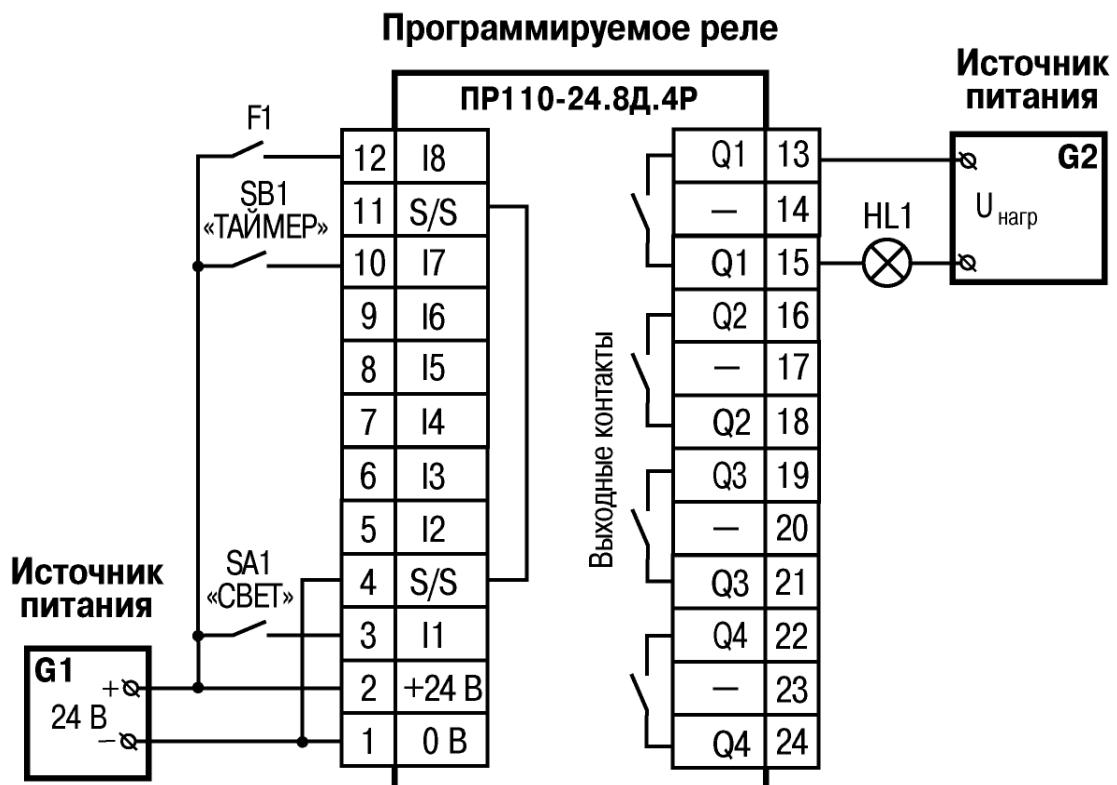


Рисунок 4.7. Электрическая схема подключения программируемого реле, SA1 – переключатель ON-OFF; SB1 – кнопка без фиксации; HL1 – световая лампа

Создание управляющей программы

На этапе планирования составляется схема реализации поставленной задачи на основе логических элементов и функциональных блоков (см. приложения А и Б).

Схема может иметь вид, приведенный на рисунке 4.8.

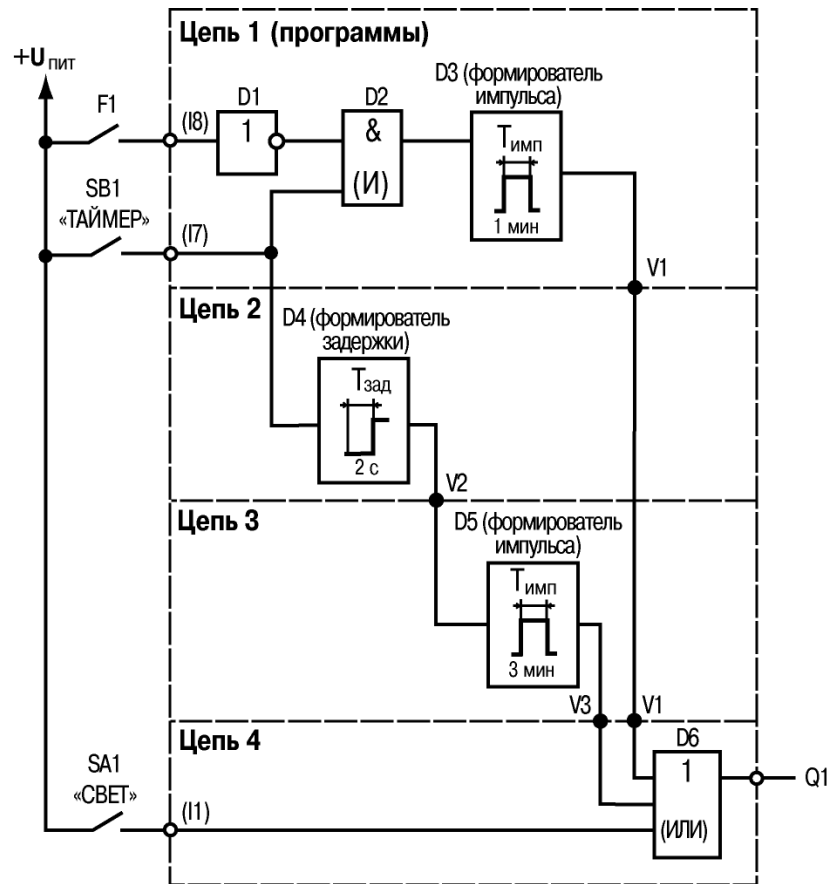


Рисунок 4.8. Цепи управляющей программы «Свет»

Описание работы схемы:

1) при кратковременном (менее 2 с) нажатии на кнопку ТАЙМЕР (SB1), лог. «1» поступает на вход логического элемента И (D2). Так как на втором входе D2 также присутствует лог. «1», на выходе D2 появится лог. «1», которая запустит формирователь импульса (D3) длительностью 1 мин. Этот импульс через элемент ИЛИ (D6) поступит на выход Q1 – реле включит своими контактами светильник на интервал действия импульса;

2) контакты датчика F1 замыкают цепь только при хорошем естественном освещении – в этом случае лог. «1» на входе D1 преобразуется в лог. «0» на выходе и поступает на вход логического элемента И (D2), что блокирует работу D2. При этом нажатие кнопки ТАЙМЕР не включит выход Q1;

3) при удержании нажатой кнопки ТАЙМЕР более 2 с на выходе блока D4 появится лог. «1», которая запустит формирователь импульса (D5) длительностью 3 мин. Этот импульс через элемент ИЛИ (D6) поступит на выход Q1 – реле включится на интервал действия импульса (независимо от состояния датчика F1);

4) при включении тумблера «СВЕТ» (SA1) лог. «1» через элемент ИЛИ (D6) поступит на выход Q1 – реле включится на время включения SA1.

Коммутационная программа, обеспечивающая работу ПР, будет иметь вид, приведенный на рисунке 4.9.

Редактор цепей

Номер цепи: 1. Название: Цель 1

Если

Вход I8 - "Вход 8" Выключен И

Вход I7 - "Вход 7" Включен

ТО

Выключить на время T, с 60 Переменная

V1 - "Переменная 1"

Номер цепи: 2. Название: Цель 2

Если

Вход I7 - "Вход 7" Включен

ТО

Включить с задержкой T, с 2 Переменная

V2 - "Переменная 2"

Номер цепи: 3. Название: Цель 3

Если

Переменная V2 - "Переменная 2" Включен

ТО

Включить на время T, с 90 Переменная

V3 - "Переменная 3"

Номер цепи: 4. Название: Цель 4

Если

Вход I1 - "Вход 1" Включен ИЛИ

Переменная V1 - "Переменная 1" Включен ИЛИ

Переменная V3 - "Переменная 3" Включен

ТО

Включить Выход Q1 - "Выход 1"

Рисунок 4.9. Управляющая коммутационная программа «Свет»

4.3. Автоматическое управление электродвигателем мешалки

При производстве пищевых продуктов часто требуется перемешивать компоненты (например, молоко или сливки на молочной ферме) в течение определенного времени.

Требуется обеспечить работу установки по следующему алгоритму:

- 1) необходимы два режима работы: «Автоматический» и «Ручной», переключаемых тумблером «РЕЖИМ» (SA1);
- 2) в режиме «Автоматический», при включении оператором установки кнопкой ПУСК (SB1), производится автоматическое включение и отключение электродвигателя через заданные интервалы времени (15 с – включен, 10 с – отключен). Отключение установки производится через интервал 5 мин или оператором при помощи кнопки СТОП (SB2);
- 3) в режиме «Ручной» производится прямое управление работой электродвигателя от кнопок ПУСК и СТОП (без временных интервалов отключения);
- 4) при перегрузке электродвигателя (на котором устанавливается соответствующий датчик – F1) должно срабатывать автоматическое отключение установки с индикацией режима «Неисправность» лампой (HL1) и звуковым прерывистым сигналом (интервал повторения звукового сигнала 3 с);
- 5) звуковой сигнал должен отключаться кнопкой СБРОС (SB3);
- 6) с помощью кнопки КОНТРОЛЬ (SB4) проверяется исправность элементов сигнализации – работа лампы и звукового сигнала.

Использование программируемого реле

Для реализации данной задачи управляющий прибор должен иметь шесть входов и три выхода.

При выполнении установки на основе ПР110-24.8Д.4Р можно воспользоваться схемой, приведенной на рисунке 4.10.

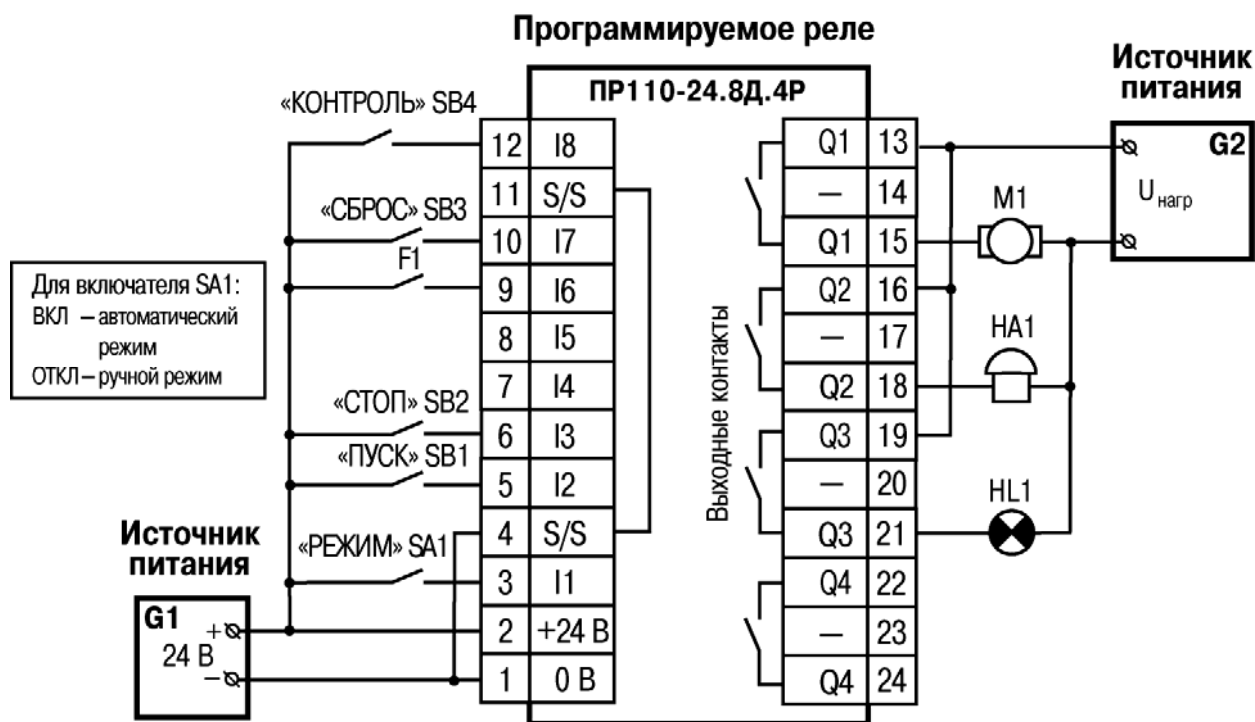


Рисунок 4.10. Электрическая схема подключения программируемого реле, SA1 – переключатель; SB1–SB4 – кнопки без фиксации; M1 – электродвигатель; HA1 – звонок; HL1 – индикаторная лампа

Работу установки можно представить в виде цепей, приведенных на рисунке 4.11.

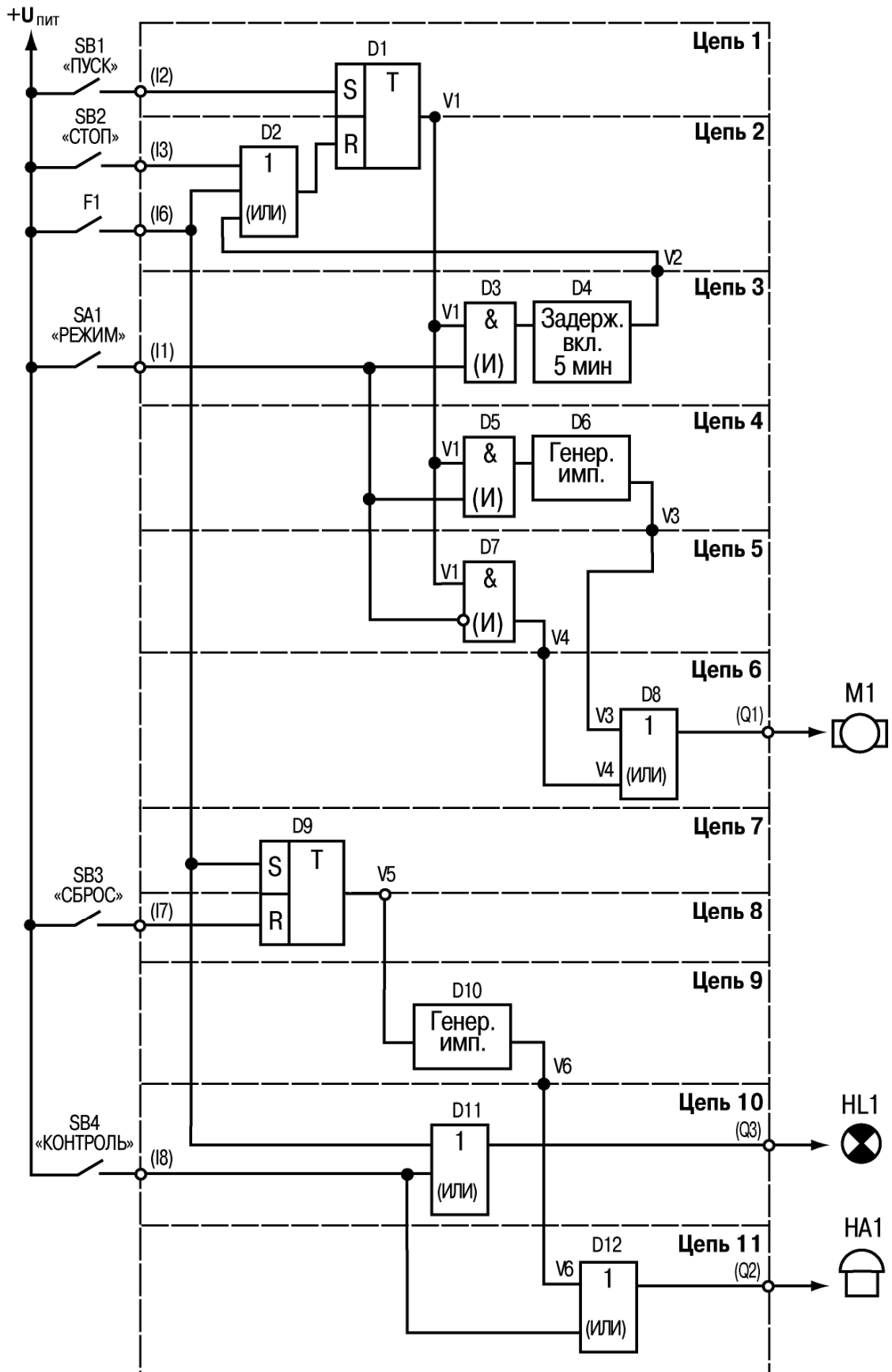


Рисунок 4.11. Цепи управляющей программы «Мешалка»

Описание работы схемы:

Цепь 1 (включение установки): при нажатии кнопки ПУСК (SB1) на входе S (D1) появляется лог. «1» – на выходе RS-триггера D1 установится лог. «1» (передается в переменную V1).

Этот сигнал дальше поступает в цепи 3–6:

– при разомкнутых контактах включателя SA1 (РЕЖИМ – Ручной), лог. «1» проходит через элементы D7, D8 и поступает на выход Q1 (электромотор M1 включится);

– при замкнутых контактах включателя SA1 (РЕЖИМ – Автоматический), лог. «1» проходит через цепи 3 и 4.

Цепь 2 (отключение установки): при нажатии кнопки СТОП (SB2) или срабатывании датчика F1 на входе R (D1) появляется лог. «1» – на выходе RS-триггера (D1) установится лог. «0» (передается в переменную V1).

Цепь 3 (формирование интервала автоматического отключения): при замкнутых контактах включателя SA1 (РЕЖИМ – Автоматический) и при включенном триггере D1, сигнал лог. «1» проходит через элемент D3 и поступает на D4 – формирователь импульса с задержкой 5 мин. Этот импульс, проходя через элемент ИЛИ (D2), поступит на вход R триггера D1 – установка отключится.

Цепь 4 (формирование интервалов работы электромотора): при замкнутых контактах включателя SA1 (РЕЖИМ – Автоматический) и при включенном триггере D1, сигнал лог. «1» проходит через элемент D5 и поступает на D6 – генератор импульсов с параметрами: лог. «1» – 15 с, лог. «0» – 10 с. Эти импульсы проходят через элемент ИЛИ (D6) и поступают на выход Q1 для управления работой электромотора M1.

Цепь 5 (передача сигнала при ручном управлении электромотором): логический элемент И (D7) обеспечивает появление лог. «1» на выходе (V4) только при разомкнутых контактах SA1 и лог. «1» на выходе D1. Сигнал с выхода D7 через элемент D8 поступает на выход Q1 для управления работой электромотора M1.

Цепь 6 (передача управляющих сигналов на выход Q1): логический элемент ИЛИ (D8) обеспечивает появление лог. «1» на выходе Q1 при появлении лог. «1» на любом из входов.

Цепь 7 (включение звуковой сигнализации): при срабатывании датчика F1 на входе S (D9) появляется лог. «1» – на выходе RS-триггера D9 установится лог. «1» (передается в переменную V5).

Цепь 8 (отключение звуковой сигнализации): при нажатии кнопки СБРОС (SB3) на входе R (D1) появляется лог. «1» – на выходе RS-триггера установится лог. «0» (передается в переменную V5).

Цепь 9 (формирование интервалов работы звукового сигнала): при включенном триггере D9 сигнал лог. «1» с его выхода поступает на генератор импульсов D10, с параметрами: лог. «1» – 3 с, лог. «0» – 3 с. Эти импульсы проходят через элемент ИЛИ (D12) и поступают на выход Q2 для управления работой звонка HA1.

Цепь 10 (контроль работы лампы сигнализации): при нажатой кнопке КОНТРОЛЬ (SB4) лог. «1» через элемент ИЛИ (D11) поступает на включение выхода Q3 – лампа HL1 будет непрерывно светиться. Выход Q3 включится также, если контакты датчика F1 замкнуты.

Цепь 11 (контроль работы звонка сигнализации): при поступлении на вход элемента ИЛИ (D12) импульсов от генератора D10, они передаются на выход Q2 – звонок HA1 будет прерывисто работать. При нажатой кнопке КОНТРОЛЬ (SB4) лог. «1» через элемент ИЛИ (D12) поступает на включение выхода Q2 – звонок HA1 будет непрерывно работать.

Коммутационная программа, обеспечивающая работу ПР, будет иметь вид, приведенный на рисунке 4.12.

Редактор цепей

Номер цепи: 1. Название: Цепь 1

Если

Вход I2 - "ПУСК" Включен

ТО

Включить и удерживать включенным Переменная

V1 - "Переменная 1"

Номер цепи: 2. Название: Цепь 2

Если

Вход I3 - "СТОП" Включен ИЛИ

Вход I6 - "Датчик" Включен ИЛИ

Переменная V2 - "Переменная 2" Включен

ТО

Выключить и удерживать выключенным Переменная

V1 - "Переменная 1"

Номер цепи: 3. Название: Цепь 3

Если

Вход I1 - "РЕЖИМ" Включен И

Переменная V1 - "Переменная 1" Включен

ТО

Включить с задержкой T, с 300,00 Переменная

V2 - "Переменная 2"

Номер цепи: 4. Название: Цепь 4

Если

Вход I1 - "РЕЖИМ" Включен И

Переменная V1 - "Переменная 1" Включен

ТО

Включить пульсацию Tмин Tмакс, с 15,00 10,00

Переменная V3 - "Переменная 3"

а)

Рисунок 4.12. Управляющая программа «Мешалка»

Номер цели: 5. Название: Цель 5
 Если
 Вход I1 - "РЕЖИМ" Выключен И
 Переменная V1 - "Переменная 1" Включен
 ТО
 Включить Переменная
 V4 - "Переменная 4"

Номер цели: 6. Название: Цель 6
 Если
 Переменная V3 - "Переменная 3" Включен ИЛИ
 Переменная V4 - "Переменная 4" Включен
 ТО
 Включить Выход
 Q1 - "МОТОР"

Номер цели: 7. Название: Цель 7
 Если
 Вход I6 - "Датчик" Включен
 ТО
 Включить и удерживать включенным Переменная
 V5 - "Переменная 5"

Номер цели: 8. Название: Цель 8
 Если
 Вход I7 - "СБРОС" Включен
 ТО
 Выключить и удерживать выключенным Переменная
 V5 - "Переменная 5"

Номер цели: 9. Название: Цель 9
 Если
 Переменная V5 - "Переменная 5" Включен
 ТО
 Включить пульсацию Тмин Тмакс, с 3,00 3,00
 Переменная V6 - "Переменная 6"

Номер цели: 10. Название: Цель 10
 Если
 Вход I6 - "Датчик" Включен ИЛИ
 Вход I8 - "КОНТРОЛЬ" Включен
 ТО
 Включить Выход
 Q3 - "СВЕТ"

Номер цели: 11. Название: Цель 11
 Если
 Вход I8 - "КОНТРОЛЬ" Включен ИЛИ
 Переменная V6 - "Переменная 6" Включен
 ТО
 Включить Выход
 Q2 - "ЗВОНК"

б)

Продолжение рисунка 4.12

Примечание. Оставшиеся свободные два входа и один выход, при необходимости, можно использовать для введения дополнительных функций. Например, переключать четыре разных интервала длительности автоматической работы электромотора или же изменять другие рабочие параметры установки.

Приложение А. Логические элементы программы

При использовании в одной программной цепи логических элементов **И** и **ИЛИ** следует учитывать порядок выполнения логических операций, по которому приоритетной всегда является операция **И**.

Например, для записей операций, приведенных на рисунке А.1, они выполняются в порядке (1 И 2), (3 И 4), а над результатами этих операций выполняется действие **ИЛИ**.

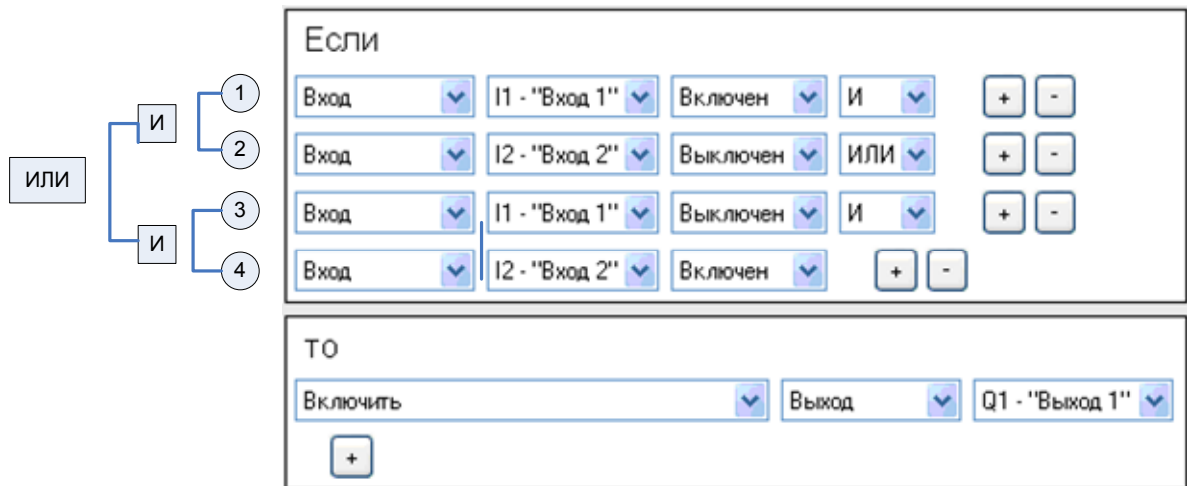
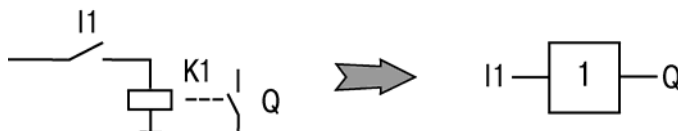


Рисунок А.1. Пример очередности выполнения логических операций

Повторитель сигнала

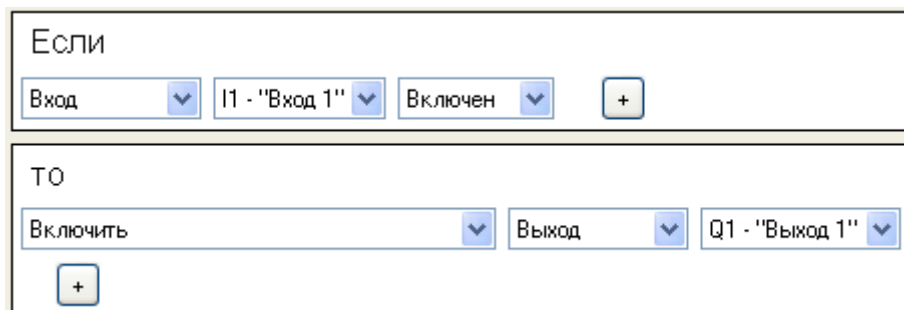


I1	Q
0	0
1	1

Элемент используется для усиления мощности сигнала управления (позволяет выходными контактами ПР коммутировать более мощную нагрузку).

На выходе элемента логическая «1» (выход включен), если на входе логическая «1» (контакты замкнуты) – повторяется сигнал.

Описание действий в программе:



Функция «НЕ» (NOT)



I1	Q
0	1
1	0

Элемент используется для преобразования замыкающего контакта в размыкающий, с усилением по мощности.

На выходе элемента логическая «1» (выход включен), если на входе логический «0» (контакты разомкнуты) и наоборот – инвертируется сигнал.

Описание действий в программе:

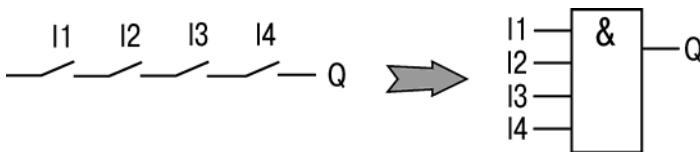
Если

Вход I1 - "Вход 1" Выключен

ТО

Включить Выход Q1 - "Выход 1"

Функция «И» (AND)



Элемент используется для выполнения логических операций.

На выходе элемента логическая «1» (выход включен), если на всех входах логическая «1» (все входы включены – контакты замкнуты).

Работе соответствует приведенная таблица состояний.

Описание действий в программе:

Если

Вход I1 - "Вход 1" Включен И

Вход I2 - "Вход 2" Включен И

Вход I3 - "Вход 3" Включен И

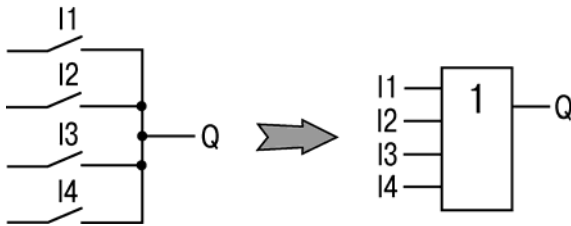
Вход I4 - "Вход 4" Включен

ТО

Включить Выход Q1 - "Выход 1"

I1	I2	I3	I4	Q
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Функция «ИЛИ» (OR)



Элемент используется для выполнения логических операций.

На выходе элемента логическая «1» (выход включен), если хотя бы на одном входе логическая «1» (контакты замкнуты).

Работе соответствует приведенная таблица состояний.

Описание действий в программе:

I1	I2	I3	I4	Q
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Если

Вход	I1 - "Вход 1"	Включен	ИЛИ	+	-
Вход	I2 - "Вход 2"	Включен	ИЛИ	+	-
Вход	I3 - "Вход 3"	Включен	ИЛИ	+	-
Вход	I4 - "Вход 4"	Включен		+	-

ТО

Включить Выход Q1 - "Выход 1"

Функция «исключающее ИЛИ» (XOR)



Элемент используется для выполнения логических операций.

На выходе элемента логическая «1» (выход включен), только если на любом из входов логическая «1».

Работе соответствует приведенная таблица состояний.

Описание действий в программе:

I1	I2	Q
0	0	0
0	1	1
1	0	1
1	1	0

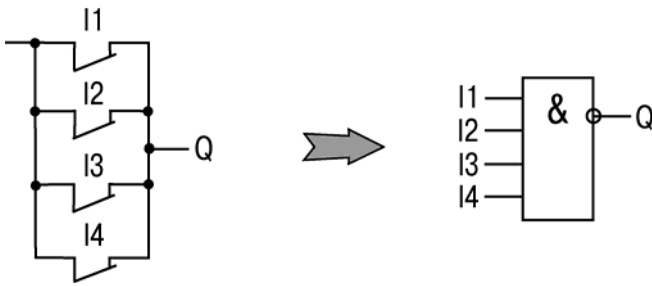
Если

Вход	I1 - "Вход 1"	Включен	И	+	-
Вход	I2 - "Вход 2"	Выключен	ИЛИ	+	-
Вход	I1 - "Вход 1"	Выключен	И	+	-
Вход	I2 - "Вход 2"	Включен		+	-

ТО

Включить Выход Q1 - "Выход 1"

Функция «И-НЕ» (NAND)



Элемент используется для выполнения логических операций.

На выходе элемента логическая «1» (выход включен), если на любом из входов логический «0».

Работе соответствует приведенная таблица состояний.

I1	I2	I3	I4	Q
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Описание действий в программе:

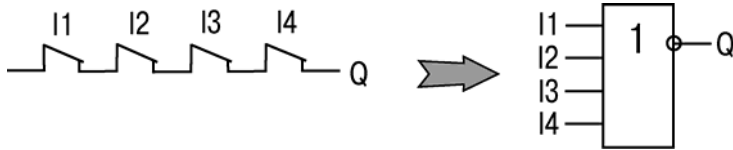
Если

Вход	I1 - "Вход 1"	Выключен	И	+	-
Вход	I2 - "Вход 2"	Выключен	И	+	-
Вход	I3 - "Вход 3"	Выключен	И	+	-
Вход	I4 - "Вход 4"	Выключен		+	-

ТО

Включить	Выход	Q1 - "Выход 1"
+		

Функция «ИЛИ-НЕ» (NOR)



Элемент используется для выполнения логических операций.

На выходе элемента логическая «1» (выход включен), если на всех входах логический «0».

Работе соответствует приведенная таблица состояний.

I1	I2	I3	I4	Q
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Описание действий в программе:

Если

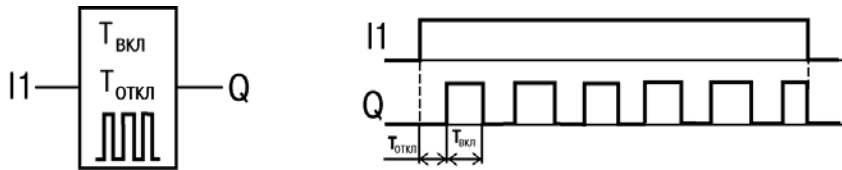
Вход	I1 - "Вход 1"	Выключен	ИЛИ	+	-
Вход	I2 - "Вход 2"	Выключен	ИЛИ	+	-
Вход	I3 - "Вход 3"	Выключен	ИЛИ	+	-
Вход	I4 - "Вход 4"	Выключен		+	-

ТО

Включить	Выход	Q1 - "Выход 1"
+		

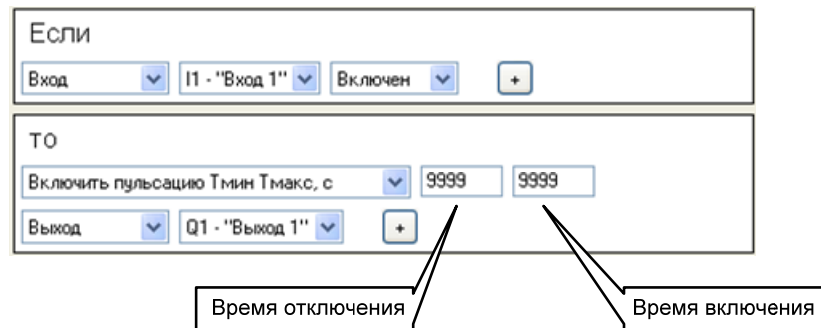
Приложение Б. Функциональные блоки программы

Генератор прямоугольных импульсов

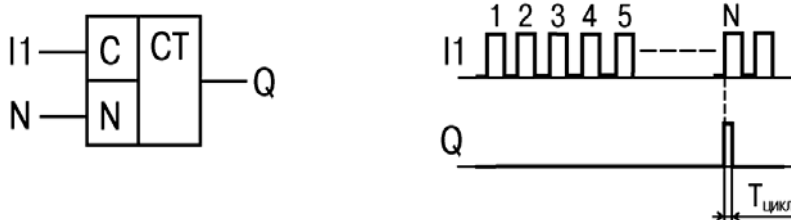


Блок используется для формирования прямоугольных импульсов пульсации. На выходе генератора формируются импульсы с заданными параметрами длительности включенного ($T_{вкл}$ – логическая «1») и отключенного ($T_{откл}$ – логический «0») состояния на время действия входного управляющего сигнала – логической «1». Работу поясняет приведенная на рисунке диаграмма.

Допустимый диапазон значений $T_{вкл}$ и $T_{откл}$ – от 0,00 до 9999 с.
Описание действий в программе:



Инкрементный счетчик с автосбросом

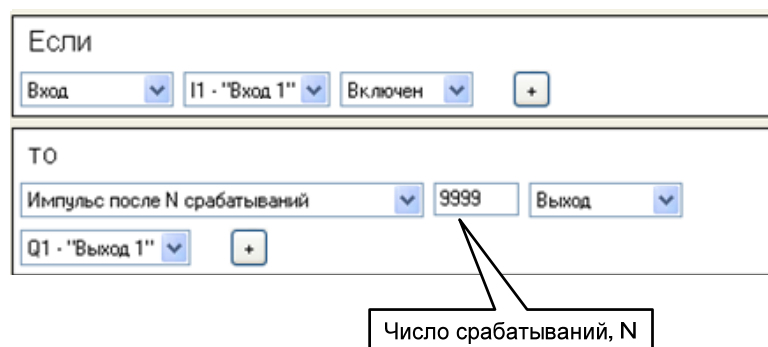


Блок используется для подсчета заданного числа импульсов.

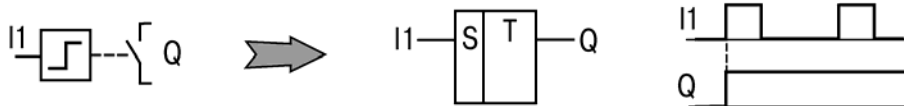
На выходе счетчика появится импульс логической «1» с длительностью $T_{цикл}$ (рабочий цикл прибора), если число приходящих на вход импульсов достигнет установленного значения N . Работу поясняет приведенная на рисунке диаграмма.

Допустимый диапазон значений числа импульсов N – от 0 до 9999.

Описание действий в программе:



Кнопка с фиксацией включения



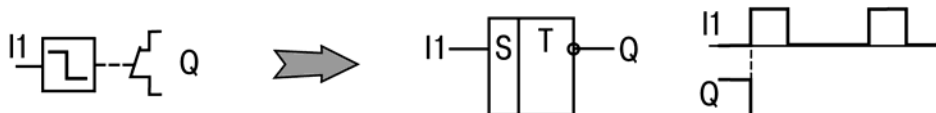
Блок используется для фиксации включенного состояния выхода.

На выходе блока установится логическая «1» (выход включен), если на входе появилась логическая «1». Состояние выхода сохраняется и после отключения входного сигнала. Работу поясняет приведенная на рисунке диаграмма.

Описание действий в программе:

Если			
Вход	I1 - "Вход 1"	Включен	+
ТО			
Включить и удерживать включенным	Выход	Q1 - "Выход 1"	+

Кнопка с фиксацией отключения



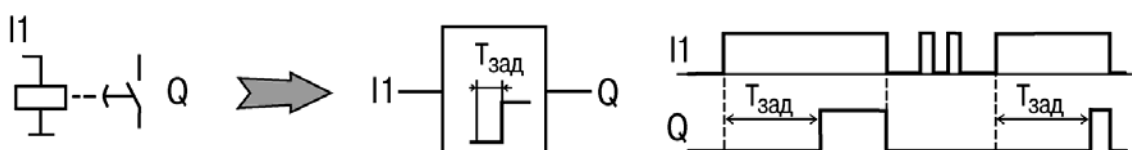
Блок используется для фиксации отключенного состояния выхода.

На выходе блока установится логический «0» (выход отключен), если на входе появилась логическая «1». Состояние выхода сохраняется и после отключения входного сигнала. Работу поясняет приведенная на рисунке диаграмма.

Описание действий в программе:

Если			
Вход	I1 - "Вход 1"	Включен	+
ТО			
Выключить и удерживать выключенным	Выход	Q1 - "Выход 1"	+

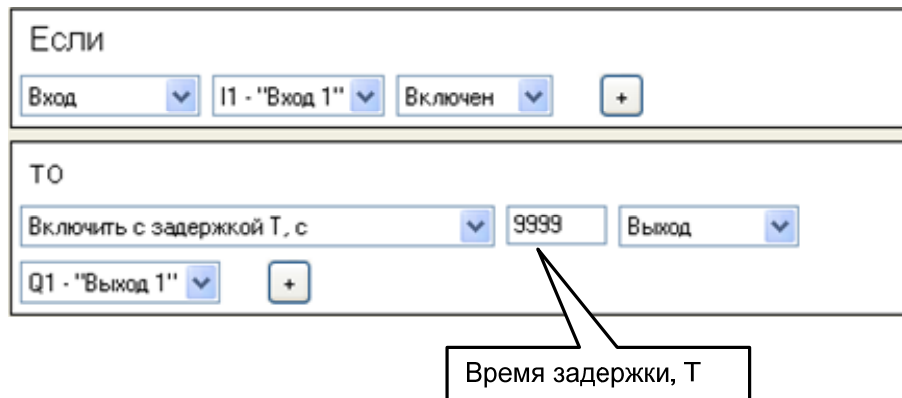
Кнопка с задержкой включения



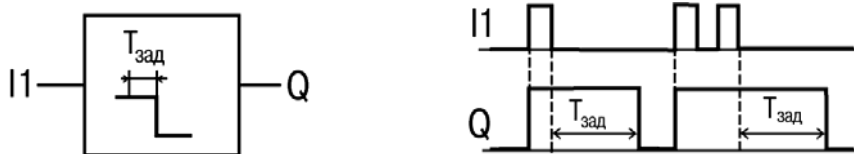
Блок используется для операции задержки передачи сигнала.

На выходе блока появится логическая «1» с задержкой относительно фронта входного сигнала. Выход включается логической «1» на входе продолжительностью не менее длительности $T_{зад}$, а выключается по спаду

входного сигнала. Работу поясняет приведенная на рисунке диаграмма.
 Допустимый диапазон значений $T_{\text{зад}}=T$ – от 0,00 до 9999 с.
 Описание действий в программе:



Кнопка с задержкой отключения



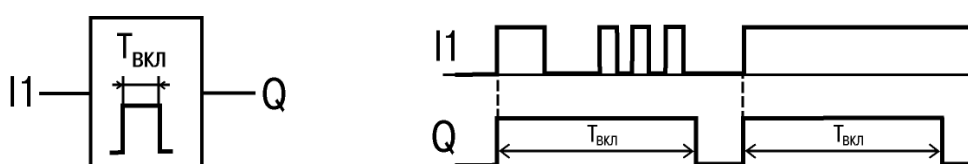
Блок используется для задержки отключения выхода.

На выходе блока появится логическая «1» по фронту входного сигнала, а начало отсчета времени задержки отключения ($T_{\text{зад}}$) происходит по каждому спаду входного сигнала. После отключения входного сигнала на выходе появится логический «0» с задержкой $T_{\text{зад}}$. Работу поясняет приведенная на рисунке диаграмма.

Допустимый диапазон значений $T_{\text{зад}}=T$ – от 0,00 до 9999 с.
 Описание действий в программе:



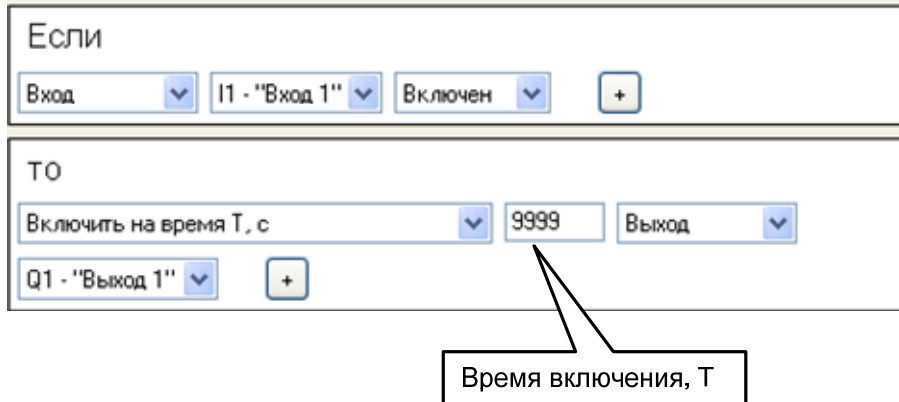
Формирователь импульса включения



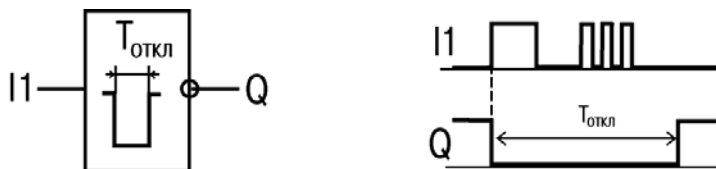
Блок используется для формирования импульса включения выхода.

На выходе блока появится логическая «1» по фронту входного сигнала. После запуска выход Q не реагирует на изменение значения входного сигнала на интервале $T_{вкл}$, а по истечении этого интервала сбрасывается в «0». Работу поясняет приведенная на рисунке диаграмма.

Допустимый диапазон значений $T_{вкл}=T$ – от 0,00 до 9999 с.
Описание действий в программе:



Формирователь импульса отключения



Блок используется для формирования импульса отключения выхода.

На выходе блока появится логический «0» по фронту входного сигнала. После запуска выход Q не реагирует на изменение значения входного сигнала на интервале $T_{откл}$, а по истечении этого интервала переключается в «1». Работу поясняет приведенная на рисунке диаграмма.

Допустимый диапазон значений $T_{откл}=T$ – от 0,00 до 9999 с.
Описание действий в программе:

