

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

КУРГАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Кафедра автоматизации производственных процессов

**ПРОГРАММИРОВАНИЕ КОНТРОЛЛЕРА ZEN
КОМПАНИИ OMRON В ПРОГРАММНОЙ СРЕДЕ
ZEN SUPPORT SOFTWARE**

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторной работы
по дисциплине «Технические средства автоматизации»
для студентов очной и заочной форм обучения специальности 220301
«Автоматизация технологических процессов и производств
(в машиностроении)», по дисциплине «Технические средства автоматизации и
управления» для студентов очной и заочной форм обучения направления
220400.62 «Управление в технических системах»,
по дисциплине «Средства автоматизации и управления»
для студентов очной и заочной форм обучения направления
220700.62 «Автоматизация технологических процессов и производств»

Курган 2012

Кафедра автоматизации производственных процессов

Дисциплины: «Технические средства автоматизации»,
«Технические средства автоматизации и управления»,
«Средства автоматизации и управления»

Составил: канд. техн. наук, доцент Н.Б.Сбродов

Утверждены на заседании кафедры АПП «03» июля 2012 г.

Рекомендованы методическим советом университета «19 » октября 2012 г.

ВВЕДЕНИЕ

Программируемый логический контроллер модели ZEN компании OMRON - это компактный малоканальный моноблочный контроллер, предназначенный для логико-программного управления дискретными технологическими процессами. Данный контроллер имеет 10 программируемых дискретных входов/выходов (6 входов и 4 выхода), что позволяет решать несложные задачи автоматизации. Применение модулей расширения позволяет увеличить количество входов/выходов.

Цель лабораторной работы – изучение основных приемов работы в программной среде ZEN SUPPORT SOFTWARE и приобретение практических навыков программирования контроллера модели ZEN компании OMRON на языке РКС (LD).

1 ПОДГОТОВКА ПРОГРАММНОГО КОМПЛЕКСА ZEN SUPPORT SOFTWARE К РАБОТЕ

После запуска программы ZEN Support Software необходимо выполнить действия, показанные на рисунках 1- 4.

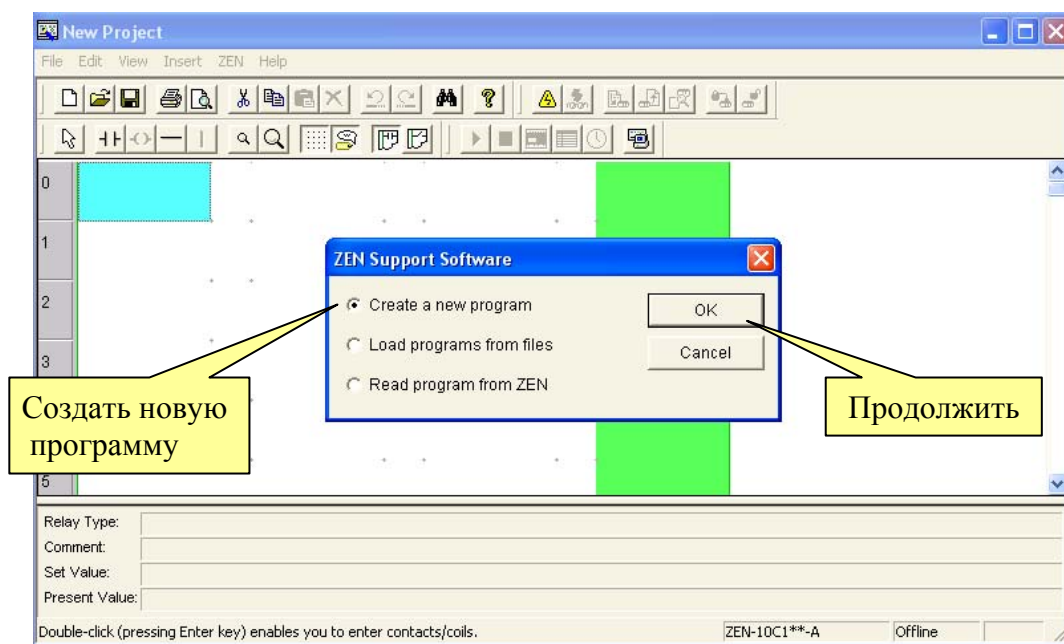


Рисунок 1 – Начало работы комплекса программирования ZEN Support Software

После выполнения всех подготовительных операций на мониторе появляется главное окно комплекса программирования ZEN Support Software, показанное на рисунке 5. Панель инструментов среды программирования показана на рисунке 6.

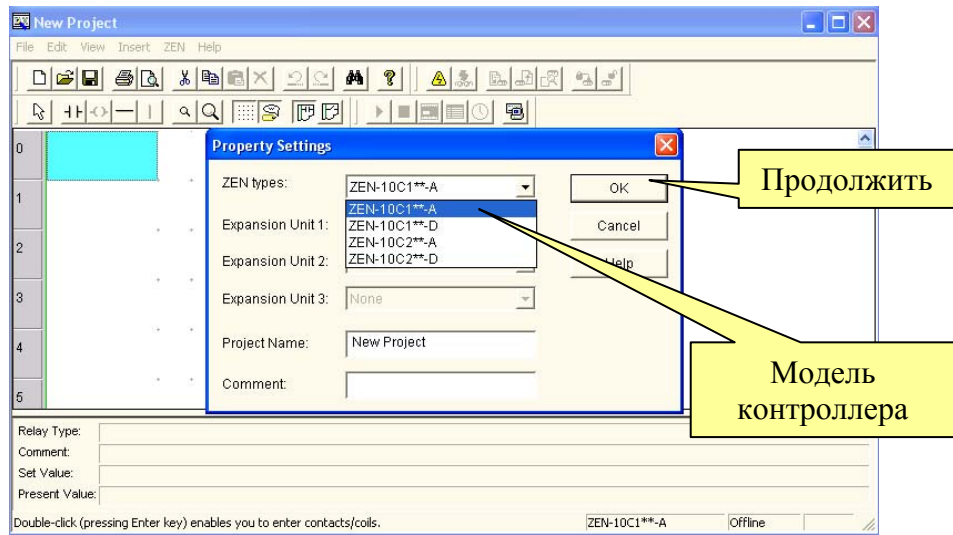


Рисунок 2 – Выбор модели контроллера

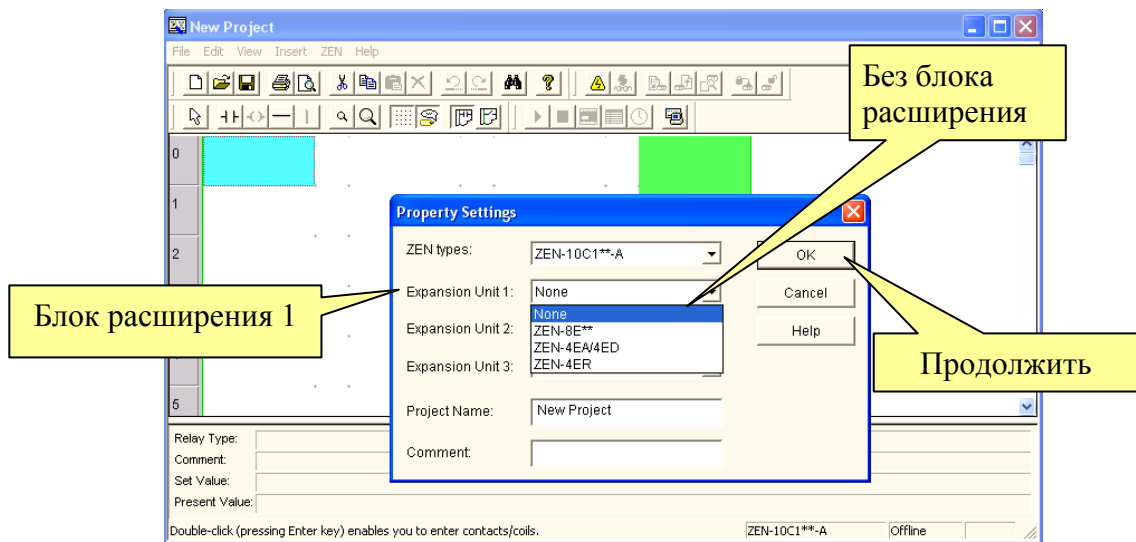


Рисунок 3 – Добавление блоков расширения

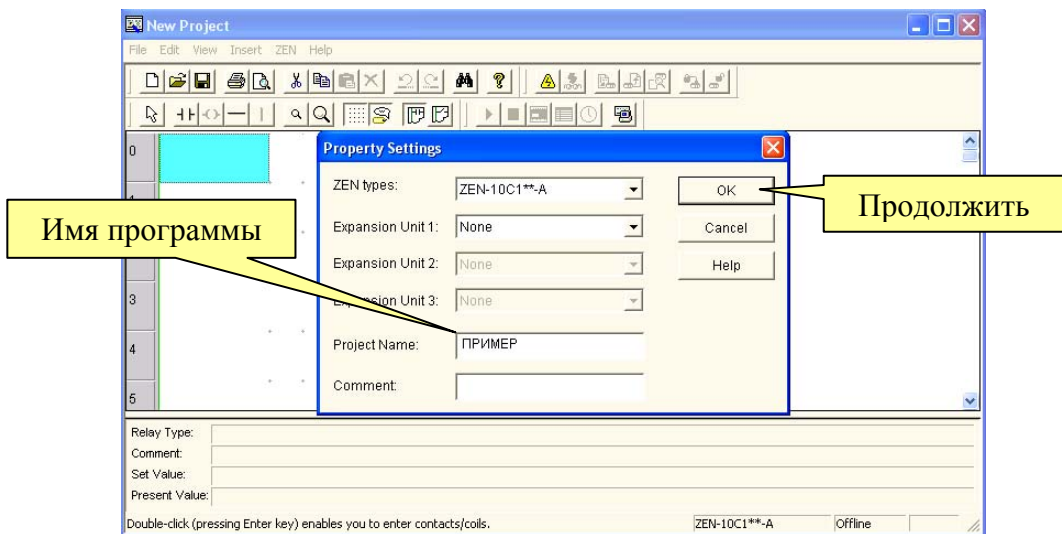


Рисунок 4 – Задание имени программы

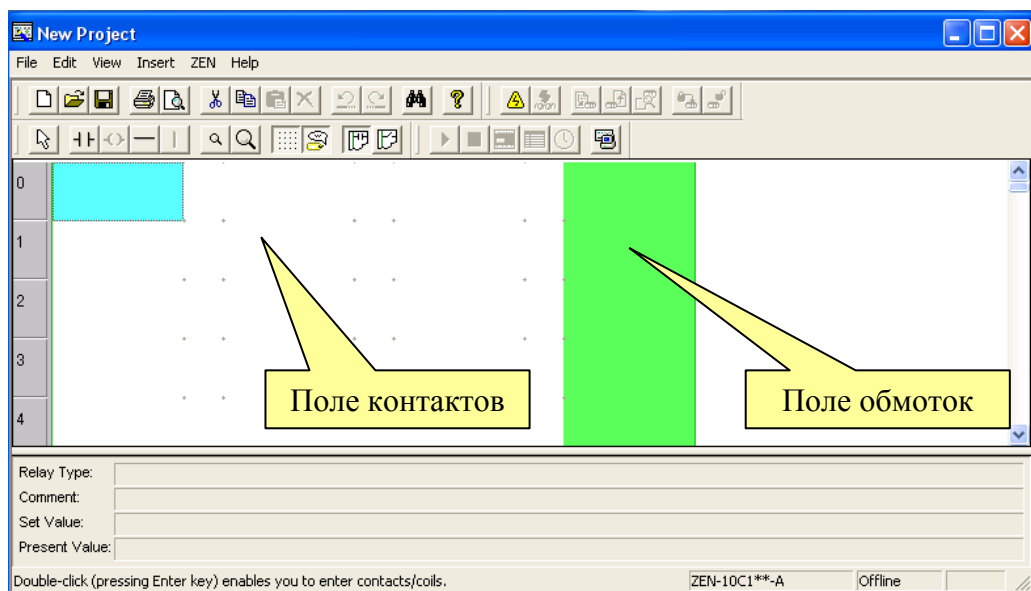


Рисунок 5 – Главное окно среды программирования ZEN Support Software



Рисунок 6 – Панель инструментов редактора

2 РАЗРАБОТКА ПРОГРАММ УПРАВЛЕНИЯ С ПОМОЩЬЮ ZEN SUPPORT SOFTWARE

2.1 Основные сведения о языке программирования

Для программирования контроллера ZEN применяется одна из разновидностей языка релейно-контактных схем [1, 2].

Язык релейно-контактных схем (РКС) или иначе язык лестничных диаграмм LD (Ladder Diagram) - графический язык, реализующий структуры электрических цепей. В 70-х годов XX века устройства управления, построенные на релейных элементах, начали постепенно вытесняться програм-

мируемыми контроллерами. Некоторое время те и другие работали одновременно и обслуживались одним и тем же персоналом.

Так появилась задача прозрачного переноса релейных электрических схем в ПЛК. Различные варианты программной реализации релейных схем создавались практически всеми ведущими производителями ПЛК. Благодаря простоте представления, язык LD обрел заслуженную популярность, что и стало основной причиной включения его в стандарт МЭК 61131.

Графически LD-диаграмма (программа на языке РКС) представлена в виде двух вертикальных шин питания. Между ними расположены цепи, образованные соединением контактов (рисунок 7). Нагрузкой каждой цепи обычно служит реле (обмотка). Каждое реле имеет контакты, которые можно использовать в других цепях. Количество контактов в цепи произвольно, обмотка одна. Если последовательно соединенные контакты замкнуты, ток идет по цепи и реле включается. При необходимости можно включить параллельно несколько обмоток, последовательное включение не допускается.

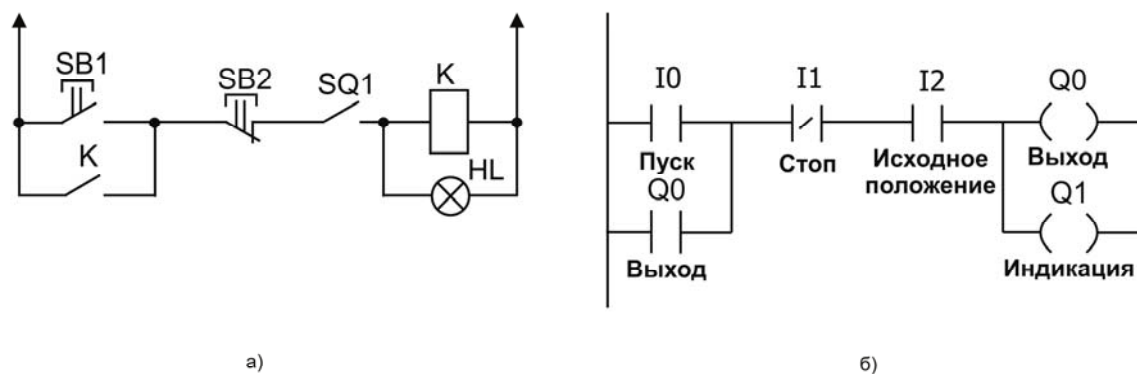


Рисунок 7 – Соответствие LD-программы электрической принципиальной схеме

Важно понимать, что контакты, обмотки реле, цепи, шины питания и пр. в LD-программе это не физические устройства, а элементы, которые реализованы **программно**.

В программе каждому контакту ставится в соответствие логическая переменная (бит), определяющая его состояние. Графическое обозначение нормально разомкнутого (замыкающего) контакта в программе $\text{—} | \text{—}$. Если контакт замкнут, то переменная имеет значение равно логической 1. Если разомкнут - логический 0. Имя переменной пишется над контактом и фактически является его названием.

Контакт может быть инверсным - нормально замкнутым (размыкающим). Такой контакт обозначается с помощью символа $\text{—} | / \text{—}$ и замыкается, если значение переменной равно логическому 0. Инверсный контакт равнозначен логической операции НЕ.

Последовательное соединение контактов или цепей равноценно логической операции И. Параллельное соединение соответствует логической операции ИЛИ.

Цепь, в зависимости от состояния контактов, может быть либо замкнутой, либо разомкнутой. Это как раз и отражается на обмотке реле и соответственно на значении логической переменной обмотки.

Каждая цепь программы на языке LD (РКС) соответствует отдельному фрагменту алгоритма управления технологическим объектом. Обычно левая часть каждой цепи программы соответствует проверке определенных условий, связанных с состоянием входных дискретных сигналов и задаваемых с помощью команд —| |— и —|/|— , которые включены последовательно и (или) параллельно. Правая часть цепи обычно соответствует некоторому действию по формированию выходного дискретного сигнала контроллера, выполняемому только в случае выполнения условия левой части цепи.

Включение реле К и лампы индикации НЛ в электрической принципиальной схеме (рисунок 7 а) произойдет в том случае, если будут замкнуты замыкающие контакты кнопки управления SB1 и конечного выключателя SQ1, а размыкающий контакт кнопки SB2 будет в исходном замкнутом состоянии. При размыкании контакта кнопки SB1 реле К и лампа индикации НЛ остаются включенными за счет замкнувшегося контакта К реле, подключенного параллельно кнопке SB1. Выключение реле и лампы обеспечивает размыкание контакта кнопки SB2.


В LD-программе (рисунок 7 б), эквивалентной электрической принципиальной схеме, значения логических переменных ПУСК, СТОП, ИСХОДНОЕ ПОЛОЖЕНИЕ (логический 0 или логическая 1) определяется состоянием соответствующих дискретных входов ПЛК с адресами I0, I1, I2. Значения указанных входных переменных определяют значения логических переменных ВЫХОД и ИНДИКАЦИЯ, связанных с дискретными выходами с адресами Q0, Q1.

Необходимо отметить, что в левой части цепи кроме логических переменных, связанных с состоянием входных дискретных сигналов, используется выходная логическая переменная ВЫХОД. Проверка состояния указанного дискретного выхода Q0 выполняется той же командой —| |— , что и проверка состояния дискретных входов. Таким образом, в левой части некоторых цепей LD-программы в качестве отдельных условий, определяющих выполнение действий, заданных в правой части цепей, могут присутствовать выходные переменные.

2.2 Программирование в ZEN SUPPORT SOFTWARE дискретных входов и выходов

Создание программы управления в ZEN Support Software выполняется в рабочей области главного окна (рисунок 5), в которой находится редактор программ.

Программа управления для контроллера ZEN может содержать до 96 цепей. Каждая цепь содержит до трех входов и одного выхода. Выходы вводятся с правой стороны цепи. Входы нельзя задать в цепи после выходов.

Для создания контакта используется кнопка панели инструментов . После создания контакта открывается окно редактирование контакта **Edit Contact** (рисунок 8). Это окно также может быть вызвано из меню **Edit** в верхней части главного окна или щелчком правой кнопки мыши по левой части рабочей области.

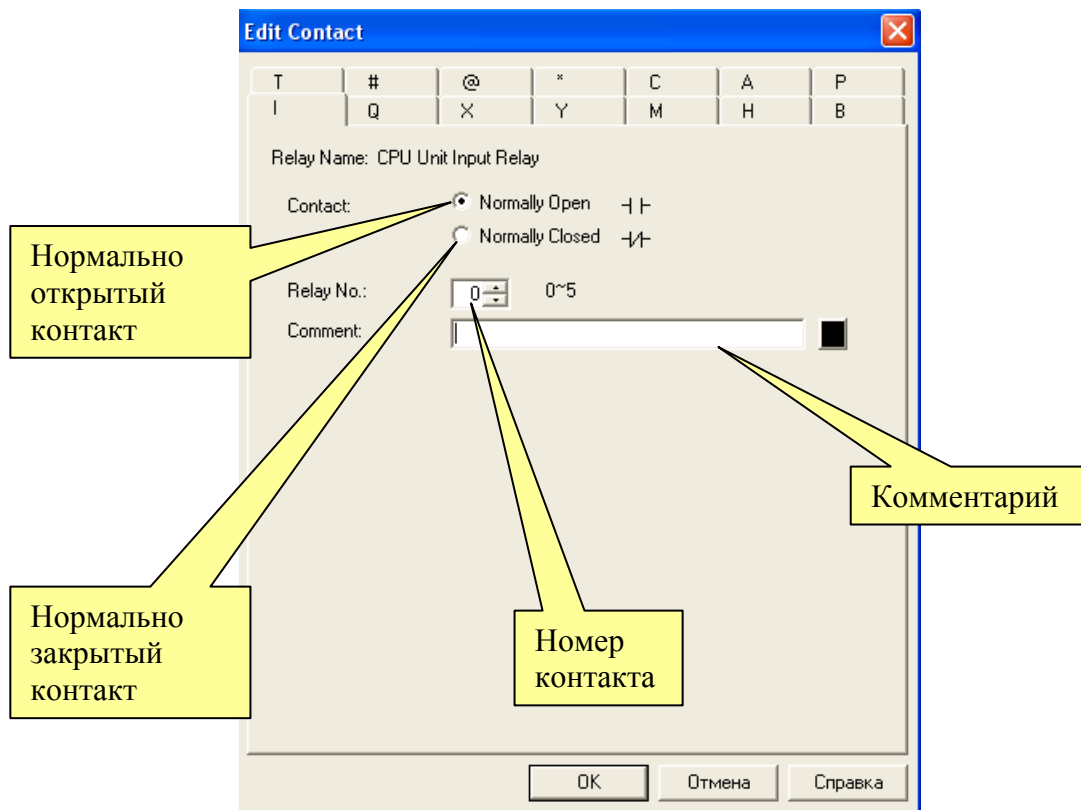




Рисунок 8 - Окно редактирования контакта

Задается тип контакта (нормально открытый или нормально закрытый) и его номер от **10** до **15**. Имя логической переменной, соответствующее заданному номеру (адресу) контакта (входа) контроллера ZEN, появляется в программе над обозначением контакта. При необходимости в строке **Comment** задаются необходимые пояснения, обычно представляющие собой наименование входного сигнала на русском языке.

Аналогичным образом программируются остальные контакты цепи.

Для создания линий связи между контактами используются кнопки панели инструментов  и .

Для создания обмотки в цепи необходимо открыть окно редактирования обмотки **Edit Coil** (рисунок 9). Это может быть сделано, например, щелчком правой кнопки мыши по правой части (зеленого цвета) рабочей области (рисунок 5), а затем - в открывшемся меню щелчком левой кнопки мыши по строке **Edit (E)**.

В окне редактирования обмотки **Edit Coil** выбирается тип обмотки:

- обычная обмотка, обозначаемая символом **[**;
- обмотка «Установка с фиксацией», обозначаемая символом **S**;
- обмотка «Сброс», обозначаемая символом **R**;

- переключающая обмотка, обозначаемая символом **A**.

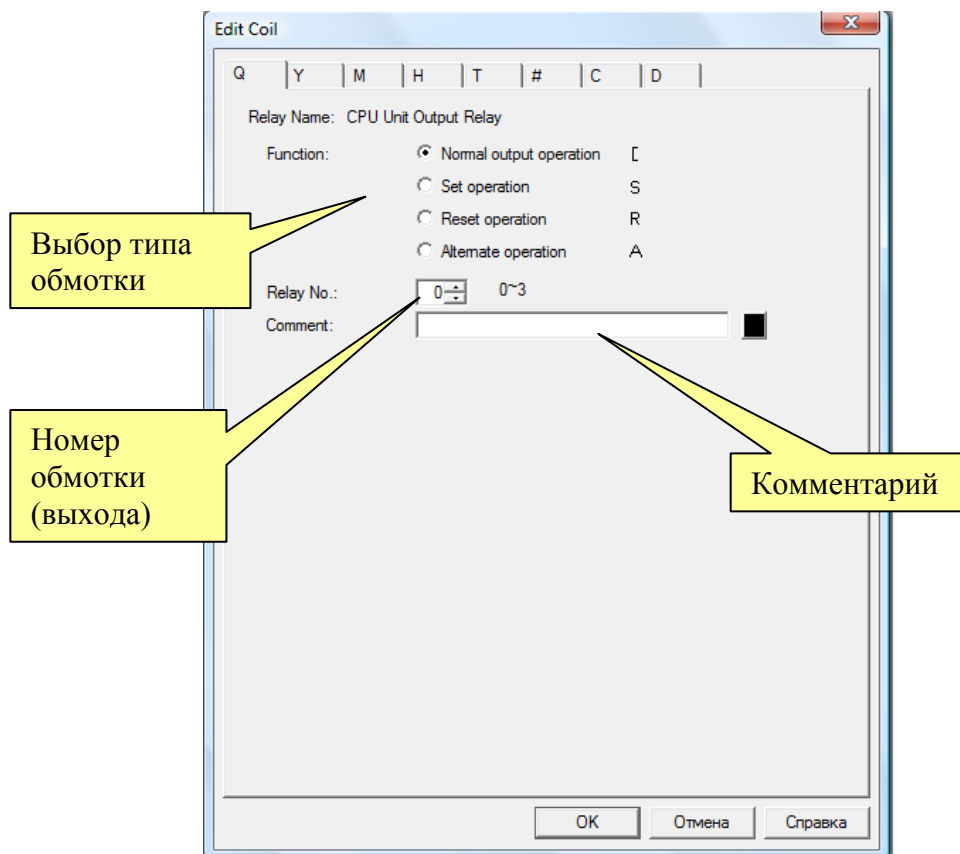


Рисунок 9 - Окно редактирования обмотки

Обычный выход будет включен только тогда, когда левая часть цепи, состоящая из контактов, будет замкнута.

Выход типа **S** «Установка с фиксацией» остается включенным после размыкания цепи. Для его выключения необходимо замкнуть цепь, в которую включена обмотка типа **R** «Сброс». Естественно эти две команды должны относиться к одному и тому же выходу. Данные команды эквивалентны обычному RS-триггеру в электрических схемах.

Для переключающего выхода типа **A** при каждом замыкание цепи, в которую он включен, состояние меняется на противоположное.

Далее в окне редактирования обмотки задается номер выхода от **Q0** до **Q3** и комментарий (наименование выходного сигнала).

Если для управления объектом с помощью контроллера ZEN требуется более 4 дискретных выходов и более 6 дискретных входов, то в этом случае необходимо использовать в ПЛК следующие модули расширения:

- модуль ZEN-8E** (4 дискретных входа и 4 дискретных выхода);
- модуль ZEN-4EA/4ED (4 дискретных входа);
- модуль ZEN-4ER (4 дискретных выхода).

Выбор необходимых модулей расширения осуществляется при начальном задании конфигурации контроллера (рисунок 3). Также это можно выполнить, выбирая строку **Properties (Свойства)** в меню **File (Файл)**.

Для модулей расширения входы (контакты) редактируются также как основные входы ПЛК в окне редактирования контактов **Edit Contact** (рисунок 10). Диапазон задания номеров контактов от **X0 - X3** до **X0 – Xb** определяется количеством и типом используемых модулей расширения.

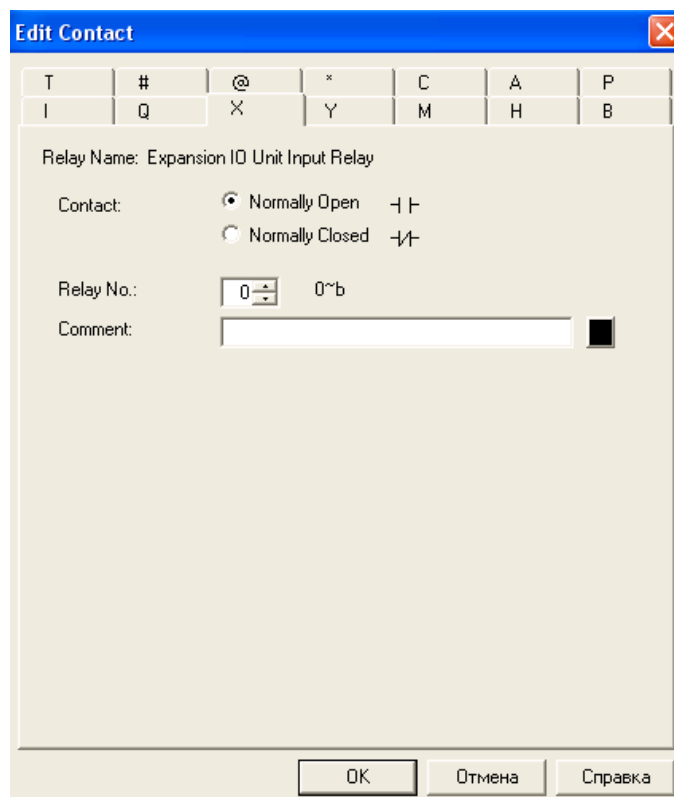


Рисунок 10 - Окно редактирования контакта модуля расширения

Выходы (обмотки) модулей расширения редактируются также как основные выходы ZEN в окне редактирования обмотки **Edit Coil**. Диапазон задания номеров обмоток от **Y0 - Y3** до **Y0 – Yb** также зависит от количества и типа используемых модулей расширения.

Кроме контактов и обмоток, соответствующих входам и выходам контроллера, в LD-программе могут использоваться внутренние переменные. Обмотки и контакты, соответствующие внутренним переменным, редактируются аналогично входам и выходам ПЛК. Обозначаются адресами от **M0** до **Mf**.

Фрагмент программы управления, созданной в ZEN Support Software, приведен на рисунке 11.

2.3 Программирование таймеров и счетчиков

Многие технологические процессы могут быть разбиты на временные интервалы. Для формирования данных интервалов и фиксации событий применяют встроенные (обычные) таймеры, таймеры удержания и календарные

таймеры. Таймер в программе управления для контроллера ZEN задается командой **—(T)—**.

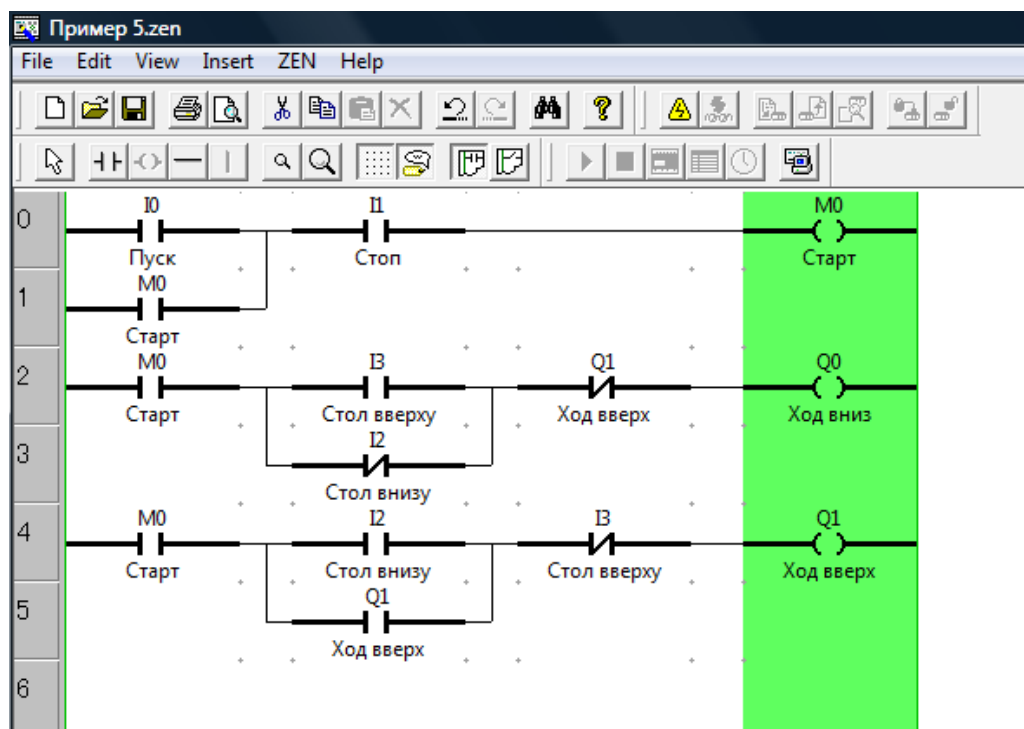


Рисунок 11 – Фрагмент LD-программы

В программе управления для ПЛК модели ZEN может быть до 8 встроенных таймеров и до 4 таймеров на удержание.

Таймеры на удержание в отличие от встроенных таймеров сохраняют свое текущее значение при переходе из режима RUN в режим STOP или при отключении питания.

Календарные таймеры обеспечивают включение и выключение исполнительных устройств в определенные периоды времени.

В данной лабораторной работе будут использоваться встроенные таймеры. В программе может быть до 8 встроенных таймеров с номерами (адресами) от **T0** до **T7**.

Создание в LD-программе таймеров схоже с заданием обмотки реле и его контактов.

Для создания в цепи таймера необходимо открыть окно редактирования обмотки **Edit Coil**. Это может сделано, например, щелчком правой кнопки мыши по правой части (зеленого цвета) рабочей области (рисунок 5), а затем - в открывшемся меню щелчком левой кнопки мыши по строке **Edit (E)**. Выбирается вкладка **T**, в которой редактируется управляющий вход таймера (рисунок 12).

Левая часть цепи, в которую включается таймер, определяет условия формирования сигнала на управляющем входе таймера.

Для задания в программе команды проверки выхода таймера необходимо открыть окно редактирование контакта **Edit Contact**. Это окно может быть

вызвано щелчком правой кнопки мыши по левой части рабочей области или из меню **Edit** в верхней части главного окна. Выбирается вкладка **T** (рисунок 13).

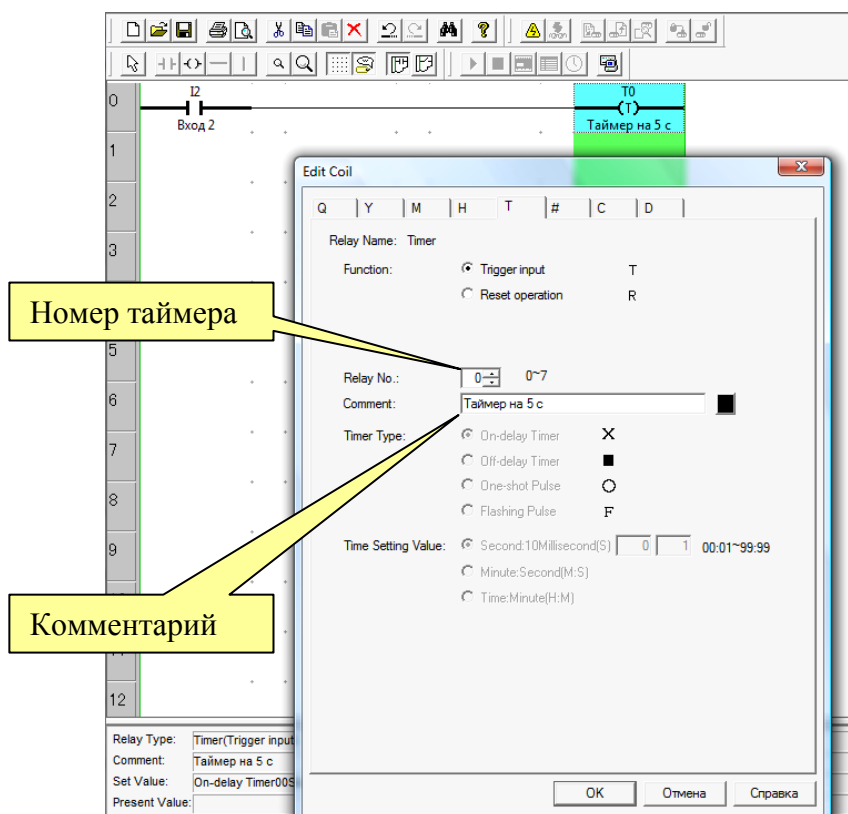


Рисунок 12 – Окно редактирования управляющего входа таймера

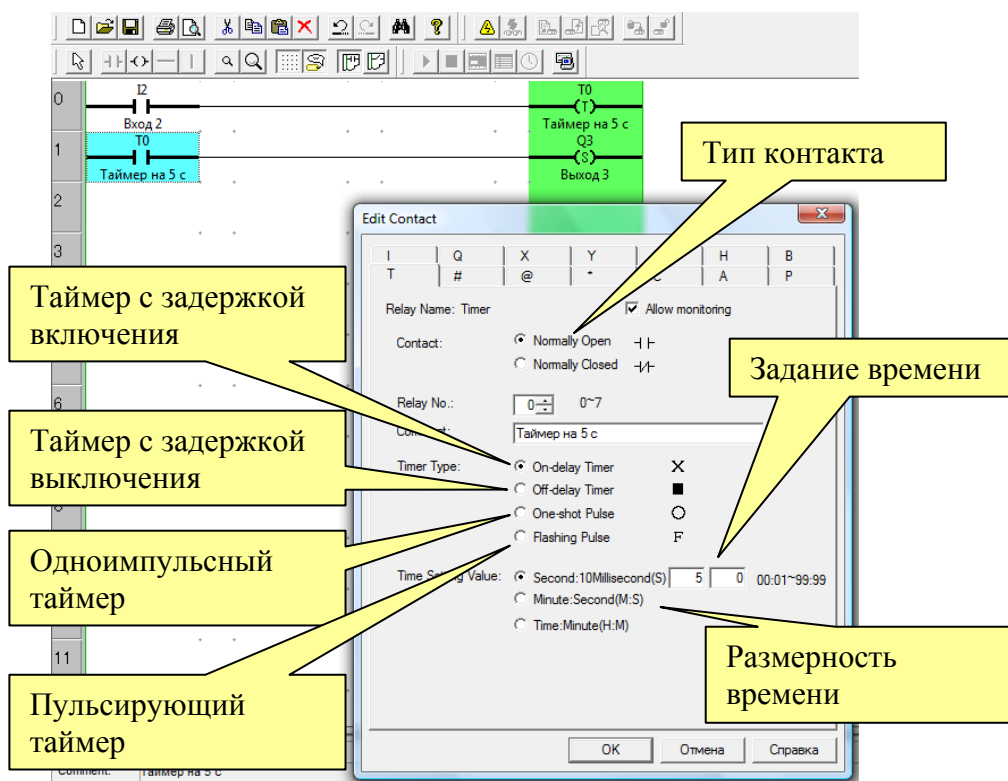


Рисунок 13 – Окно редактирования выхода таймера

Существует 4 типа (режима работы) таймеров, задаваемых выбором соответствующей функции.

Временная диаграмма работы таймера с задержкой включения (**тип X**) приведена на рисунке 14. При наличии управляющего сигнала (логическая 1) и отсутствии сигнала сброса таймер начинает отсчет выдержки времени. При достижении текущим значением времени заданного значения (уставки) выходной бит таймера устанавливается в единицу. Состояние выхода сбрасывается в ноль при переходе в ноль состояния управляющего входа. Состояние выходного бита остается равным нулю, если время активного состояния управляющего входа меньше заданного значения времени уставки.

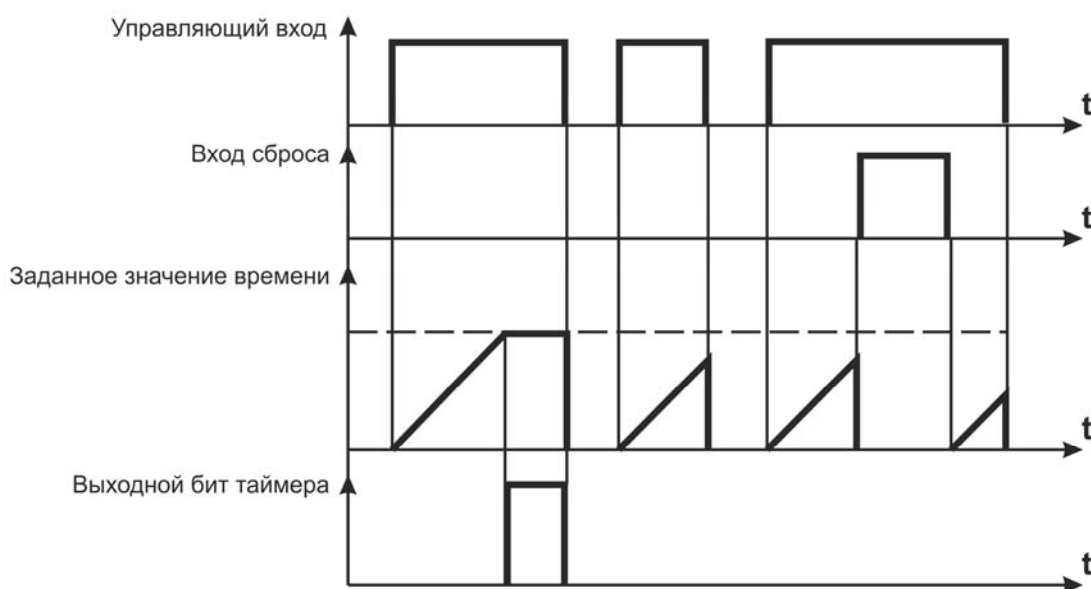


Рисунок 14 – Временная диаграмма работы таймера с задержкой включения (**тип X**)

Временная диаграмма работы таймера с задержкой выключения (**тип ■**) приведена на рисунке 15. Выходной бит остаётся включенным, пока управляющий вход включен, и отключается спустя некоторое время после отключения управляющего входа.

Временная диаграмма работы таймера в режиме короткого импульса – одноимпульсного таймера (**тип O**) приведена на рисунке 16. Выходной бит остаётся включенным, в течение определенного времени после включения управляющего входа.

Временная диаграмма работы таймера в режиме циклической последовательности – пульсирующего таймера (**тип F**) приведена на рисунке 17. Выходной бит циклически включается и выключается с заданным интервалом времени пока включен управляющий вход.

Во фрагменте программы, приведенной на рисунке 13, при поступлении на вход **I2** сигнала «Вход 2» запускается таймер с задержкой на включение (цепь 0). После достижения текущим значением времени заданной уставки, равной 5 с, выходной бит таймера устанавливается в единицу (цепь 1). Это

вызывает включение выхода Q3 «Выход 3» и фиксацию включенного состояния. При отработке таймером заданной уставки цепь (контакт I2), формирующая сигнал на управляющем входе таймера, должна быть постоянно замкнутой. При размыкании контакта I2 после отработки таймером заданной уставки выходной бит сбрасывается в ноль (рисунок 14).

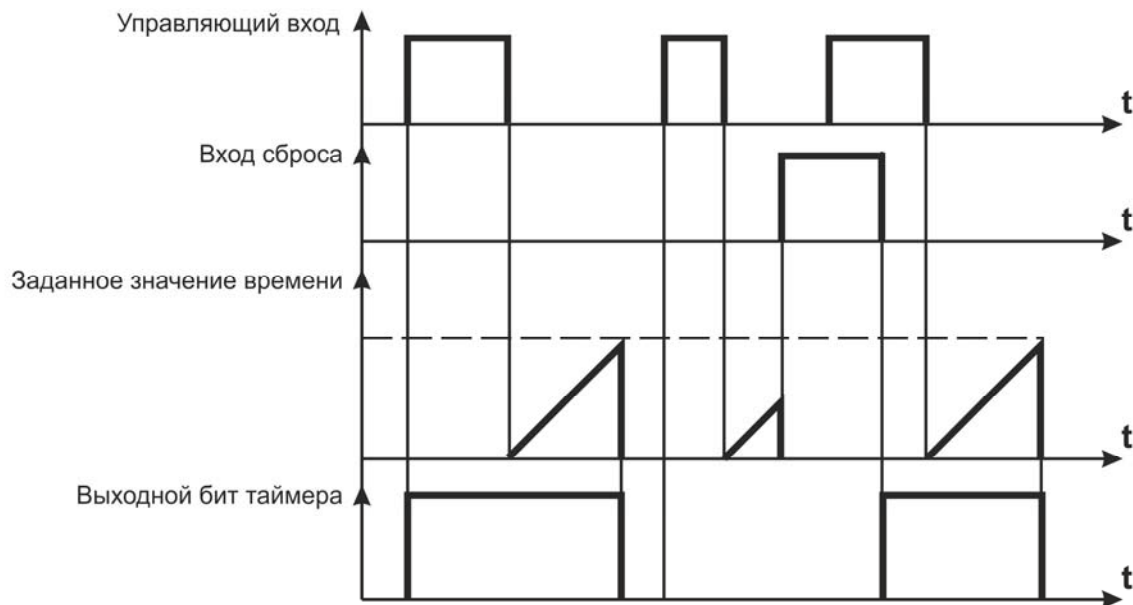


Рисунок 15 – Временная диаграмма работы таймера с задержкой выключения (тип ■)

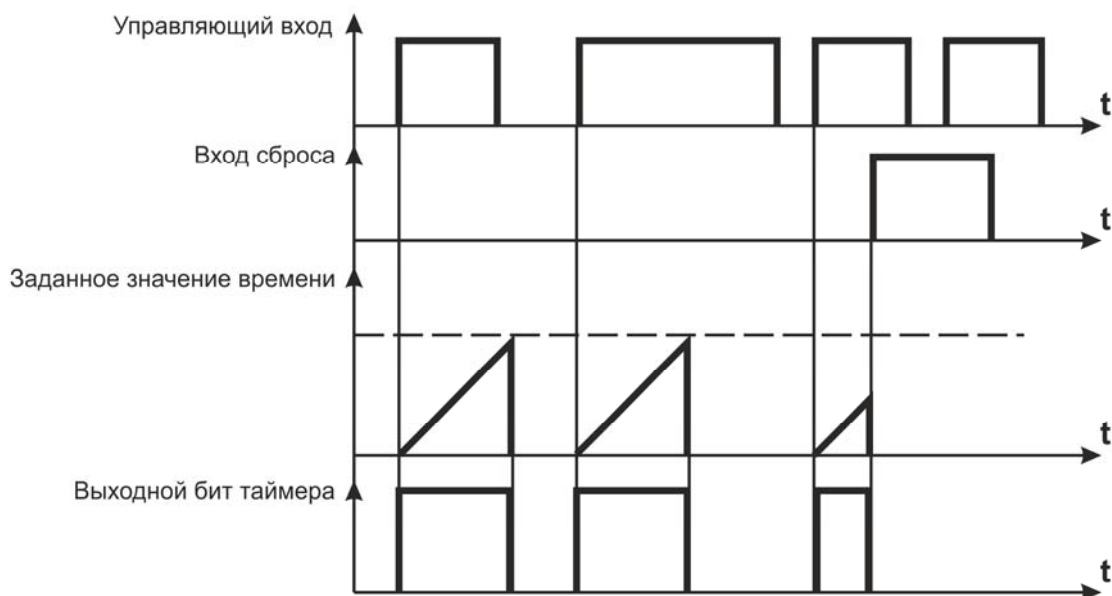


Рисунок 16 – Временная диаграмма работы таймера в режиме короткого импульса (тип ○)

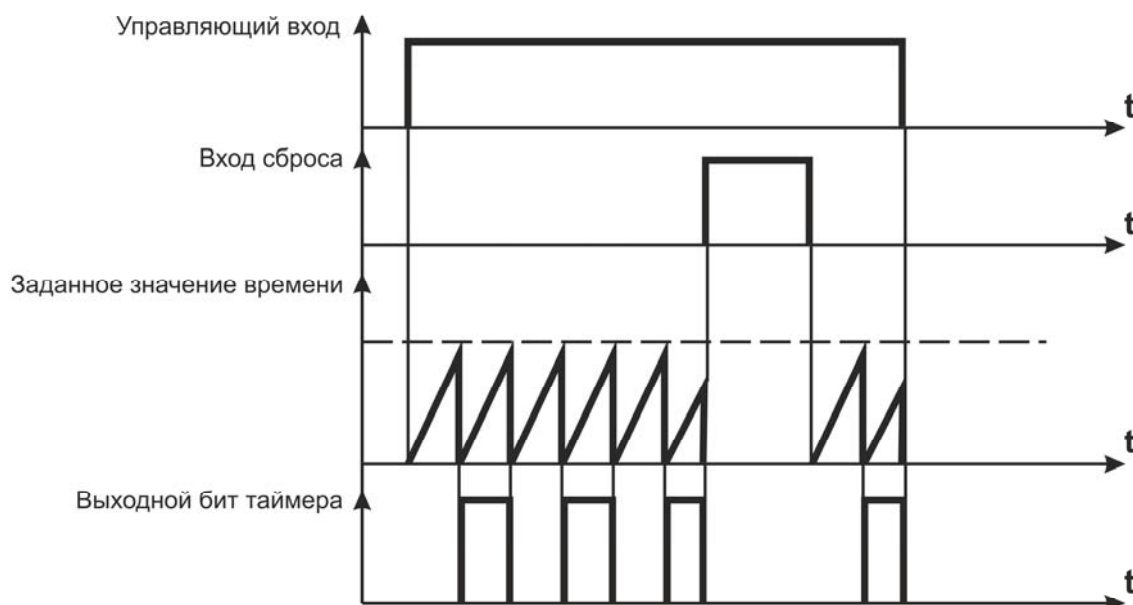


Рисунок 17 – Временная диаграмма работы пульсирующего таймера (тип F)

Создание в LD-программе счетчиков схоже с заданием обмотки реле и его контактов. Счетчик в ZEN Support Software задается командой **—(C)—**.

Для создания в цепи счетчика необходимо открыть окно редактирования обмотки **Edit Coil**. Выбирается вкладка **C**, в которой редактируется управляющий вход счетчика (рисунок 18). Левая часть цепи, в которую включается счетчик, определяет условия формирования сигнала на управляющем входе таймера.

Для задания в программе команды проверки выхода счетчика необходимо открыть окно редактирование контакта **Edit Contact**. Выбирается вкладка **C** (рисунок 19).

Во фрагменте программы, приведенной на рисунке 19, при поступлении на вход **I5** контроллера каждого счетного импульса с датчика «Наличие детали» состояние счетчика **C6** увеличивается на единицу (цепь 0). После достижения текущим состоянием счетчика **C6** «Наличие детали» заданной уставки, равной 15, выходной бит счетчика устанавливается в единицу (цепь 1). Это вызывает включение выхода **Q0** «Загрузка». Дальнейшее поступление счетных импульсов не изменяет состояние счетчика.

Для сброса счетчика в ноль необходимо организовать дополнительную цепь (цепь 2) и выбрать (рисунок 20) в окне редактирования счетчика функцию **R** «Сброс» («Reset operation»).

При поступлении на вход **I2** дискретного входного сигнала «Сброс» текущее состояние счетчика сбрасывается в ноль. Если выходной бит счетчика был установлен в единицу, то он также обнуляется. После окончания входного сигнала «Сброс» счетчик вновь готов выполнять свою функцию.

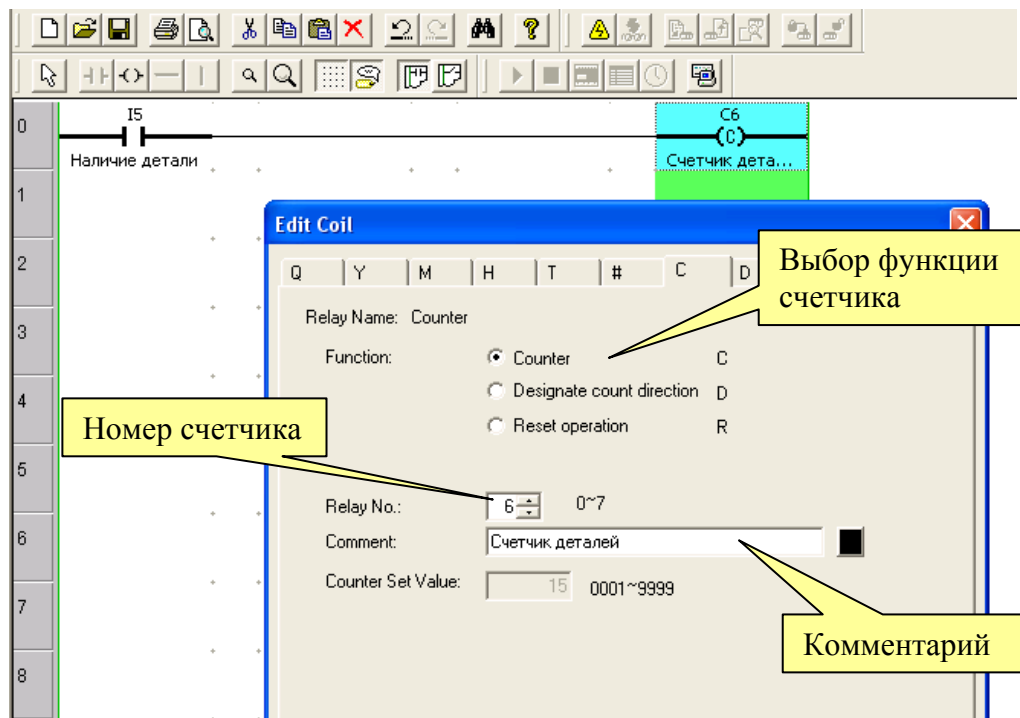


Рисунок 18 – Окно редактирования управляющего входа счетчика

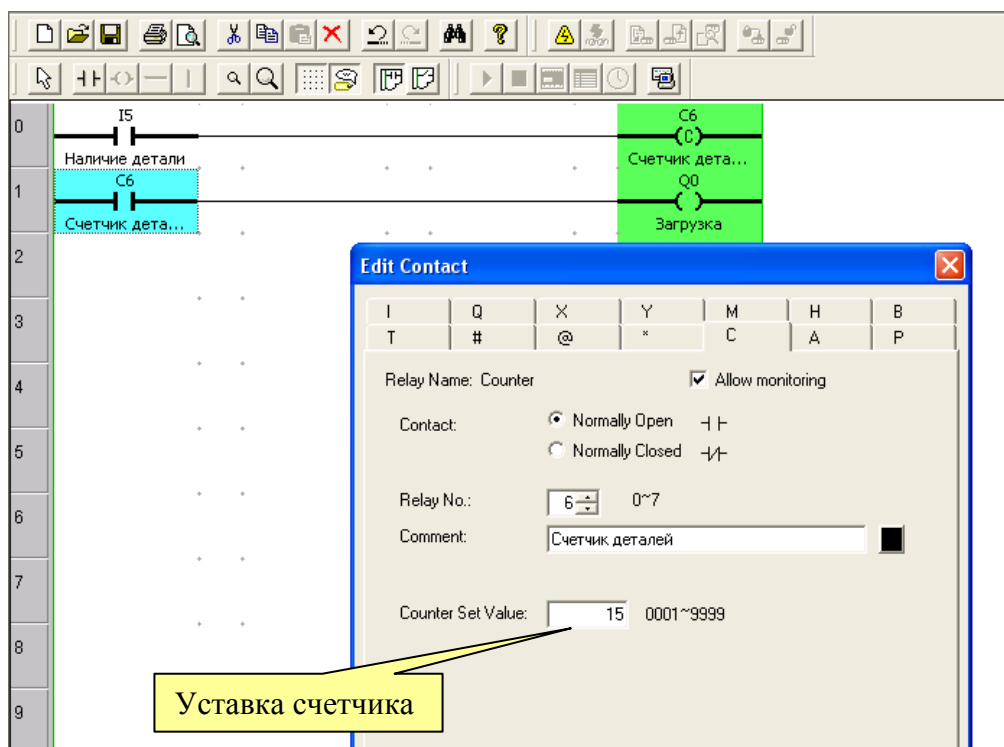


Рисунок 19 – Окно редактирования выхода счетчика

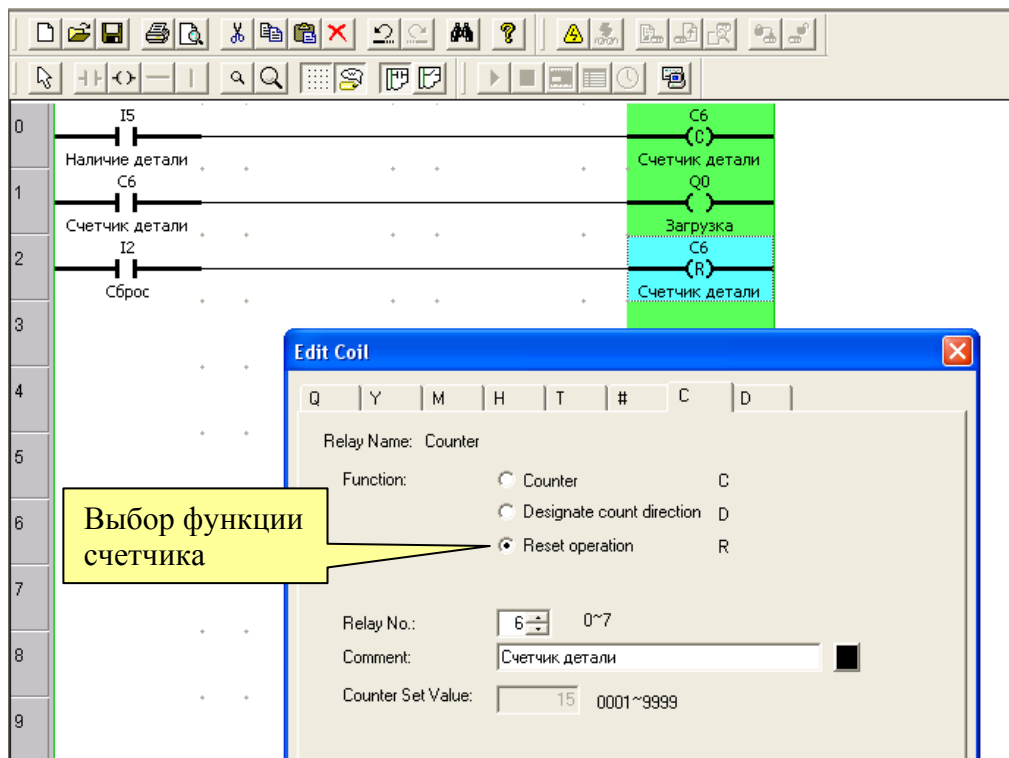


Рисунок 20 – Окно редактирования функции счетчика

3 ИССЛЕДОВАНИЕ LD-ПРОГРАММЫ С ПОМОЩЬЮ ВСТРОЕННОГО СИМУЛЯТОРА

Разработанную LD-программу можно проверить без загрузки в контроллер. Для этих целей в ZEN Support Software имеется встроенный симулятор.

Перед запуском симулятора рекомендуется проверить программу с целью обнаружения возможных ошибок. Для этого открываем меню **ZEN** и выбираем строку **Program Check** (рисунок 21).

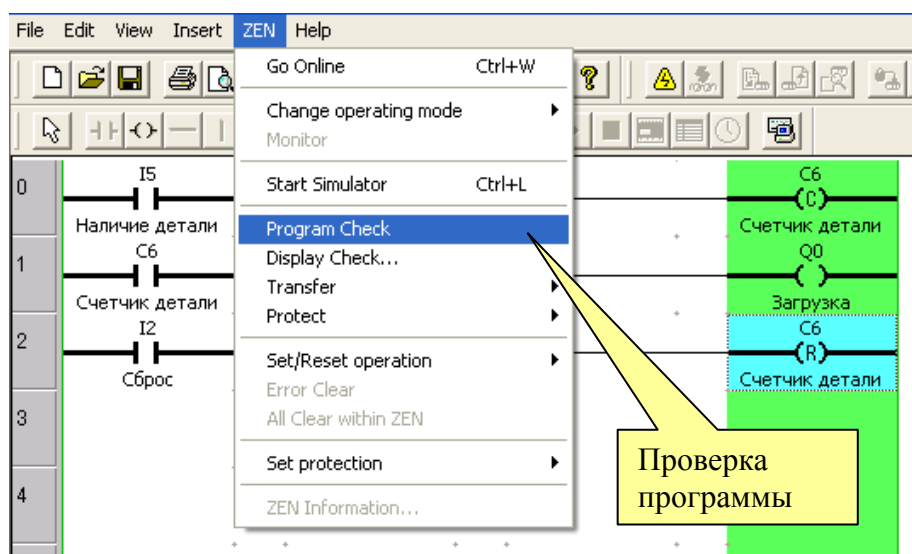



Рисунок 21 – Задание проверки программы

При отсутствии в программе ошибок выдается сообщение «**Normal End of Program Check**».

Для запуска симулятора необходимо на панели инструментов нажать кнопку , а затем подтвердить запуск кнопкой **ОК** (рисунок 22).

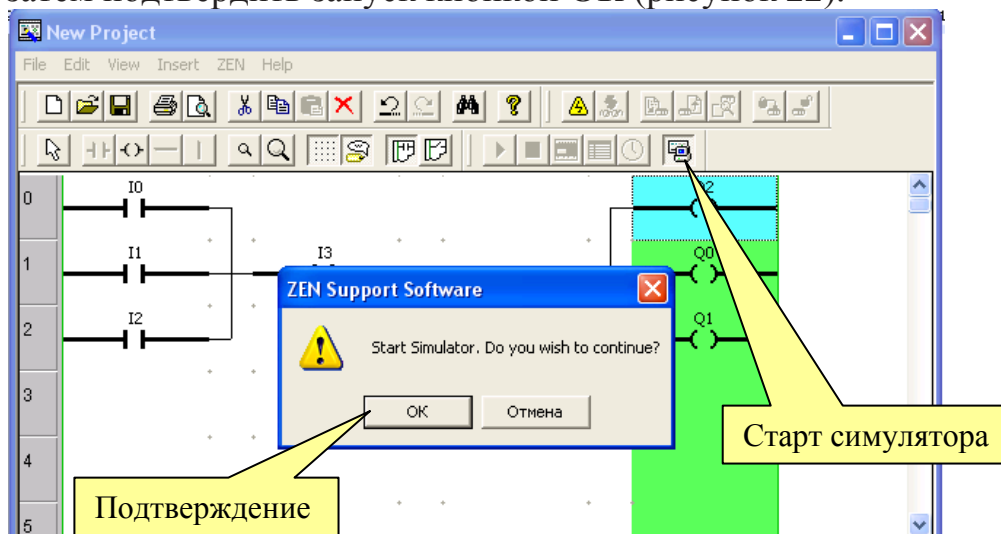


Рисунок 22 – Запуск симулятора

После запуска симулятора цвет фона рабочей области станет желтым, будут доступны кнопки **RUN** и **STOP** и откроется окно **ZEN Image** (рисунок 23).

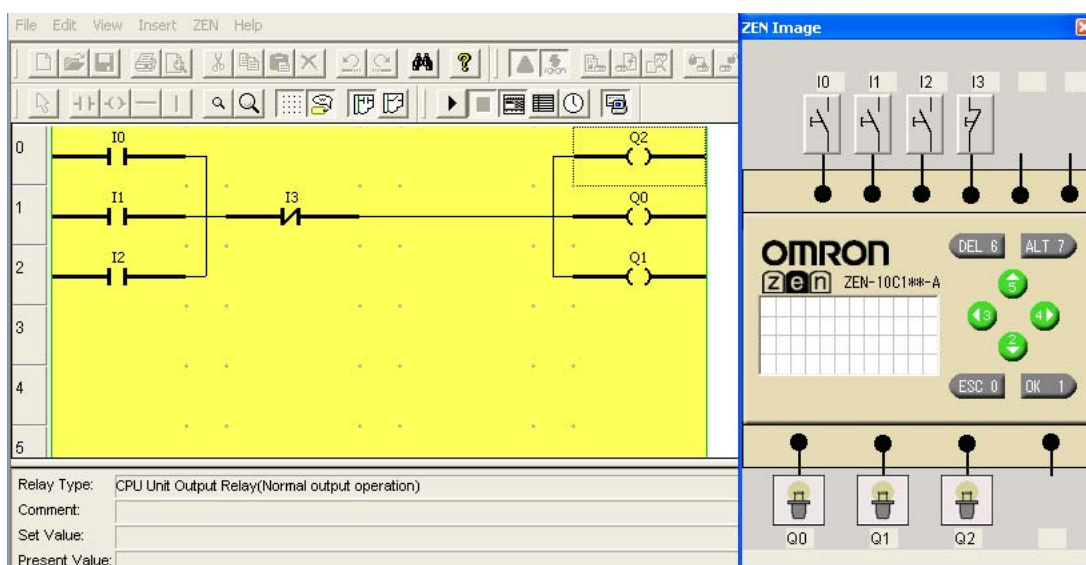



Рисунок 23 – Рабочая область в режиме симуляции

В окне **ZEN Image** отображаются образы (изображения) контроллера и модулей расширения в соответствии с конфигурацией, указанной в настройках проекта. Напротив дискретных входов/выходов, использованных в LD-программе, отображаются переключатели состояния входов и сигнальные лампы, отображающие состояния выходов. Для запуска выполнения программы необходимо нажать на панели инструментов кнопку .

Задавая с помощью переключателей в окне ZEN Image входные сигналы в контроллер, можно исследовать работу LD-программы с целью проверки правильности реализации алгоритма управления.

Работающие цепи программы подсвечиваются в рабочей области симулятора зеленым цветом. Включение выходов имитируется загоранием соответствующих сигнальных ламп (рисунок 24). Тип входного переключателя можно изменить, щелкнув по нему правой кнопкой мыши.

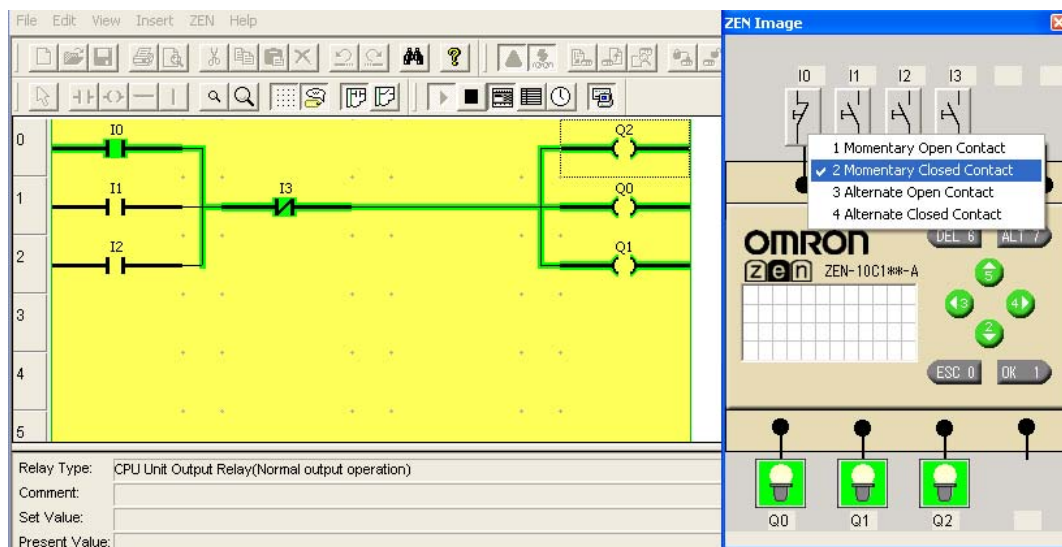




Рисунок 24 – Выполнение LD-программы в режиме симуляции

Для остановки выполнения LD-программы в режиме симуляции необходимо на панели инструментов нажать кнопку . Для выхода из режима симуляции необходимо на панели инструментов нажать кнопку , а затем подтвердить выход кнопкой **ОК**.

4 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Лабораторную работу необходимо выполнять в следующей последовательности:

- 1 Изучить основные сведения о составе и возможностях программного комплекса ZEN Support Software.
- 2 Запустить на компьютере программу ZEN Support Software.
- 3 Пользуясь данными методическими указаниями, изучить методику программирования контроллера ZEN на языке LD и отработать в среде ZEN Support Software все примеры, приведенные в разделах 2 и 3.
- 4 Получить у преподавателя задание на разработку программы управления дискретным технологическим объектом.
- 5 На основе анализа алгоритма управления определить состав входных и выходных дискретных сигналов.
- 6 Написать LD-программу управления в среде ZEN Support Software.
- 7 Исследовать разработанную программу в режиме симуляции.

8 Оформить отчет по лабораторной работе.

5 СОДЕРЖАНИЕ ОТЧЕТА

В отчете указывается цель лабораторной работы и приводятся следующие результаты:

- 1 Задание на разработку программы управления дискретным технологическим объектом.
- 2 Таблицы входных и выходных сигналов с указанием наименования сигналов, их условных обозначений, источников (датчиков) и приемников (исполнительных устройств) сигналов.
- 3 Блок-схема алгоритма управления.
- 4 Текст программы на языке LD, созданный в среде ZEN Support Software.
- 5 Выводы по результатам исследования программы в режиме симуляции.

6 КОНТРОЛЬНЫЕ ВОПРОСЫ

- 1 Каково назначение базовых команд языка?
2. Какова структура LD-программы?
- 3 Как в LD-программе задают проверку состояния входных дискретных сигналов?
- 4 Какими командами в LD-программе формируют выходные дискретные сигналы?
- 5 Каким образом можно фиксировать выходные дискретные сигналы?
- 6 Как в LD-программе для контроллера ZEN задают таймеры?
- 7 Как в программной среде ZEN Support Software задают счетчики?
- 8 Как проверить правильность LD-программы?

7 СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Минаев И.Г., Самойленко В.В. Программируемые логические контроллеры: практическое руководство для начинающего инженера.– Ставрополь: АРГУС, 2009. – 100 с.
- 2 Петров И.В. Программируемые контроллеры. Стандартные языки и инструменты. – М.: СОЛОН-Пресс, 2003. – 256 с.

Сбродов Николай Борисович

**ПРОГРАММИРОВАНИЕ КОНТРОЛЛЕРА ZEN
КОМПАНИИ OMRON В ПРОГРАММНОЙ СРЕДЕ
ZEN SUPPORT SOFTWARE**

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторной работы
по дисциплине «Технические средства автоматизации»
для студентов очной и заочной форм обучения специальности 220301
«Автоматизация технологических процессов и производств
(в машиностроении)», по дисциплине «Технические средства автоматизации и
управления» для студентов очной и заочной форм обучения направления
220400.62 «Управление в технических системах»,
по дисциплине «Средства автоматизации и управления»
для студентов очной и заочной форм обучения направления
220700.62 «Автоматизация технологических процессов и производств»

Авторская редакция

Подписано в печать
Печать трафаретная
Заказ

Формат 60x84 1/16
Усл. печ. л. 1,25
Тираж 20

Бумага тип. №1
Уч.-изд. л. 1,25
Цена свободная

РИЦ Курганского государственного университета.
640669, г. Курган, ул. Гоголя, 25.
Курганский государственный университет.