

Проект «Инженерные кадры Зауралья»

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Курганский государственный университет»

Кафедра автоматизации производственных процессов

ИССЛЕДОВАНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПЕРЕКАЧИВАНИЕМ И ДОЗИРОВАНИЕМ ЖИДКИХ ПРОДУКТОВ

Методические указания
к выполнению лабораторной работы
по дисциплине «Автоматизация технологических процессов и производств»
для студентов очной и заочной форм обучения направления 220700.62
«Автоматизация технологических процессов и производств»

Курган 2014

Кафедра автоматизации производственных процессов

Дисциплина: «Автоматизация технологических процессов и производств»

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

Утверждены на заседании кафедры «14» ноября 2013 г.

Рекомендованы методическим советом университета в рамках проекта «Инженерные кадры Зауралья» «22» ноября 2013 г.

ВВЕДЕНИЕ

Ускорение научно-технического прогресса и интенсификация производства невозможны без применения средств автоматизации. Важной особенностью современного этапа автоматизации является тот факт, что она опирается на самое широкое использование микропроцессорных контроллеров.

Широкое применение систем автоматизации привело с одной стороны к существенному сокращению количества работников занятых в производстве, а с другой стороны повысила роль персонала занятого обслуживанием и сопровождением систем автоматизации, так как в виду высокой производительности автоматизированных систем, даже не большой по времени простой системы приводит к существенным экономическим потерям.

Названные обстоятельства способствовали возникновению целого комплекса проблем по сопровождению, диагностированию и техническому обслуживанию систем автоматизации технологических процессов. Решение указанных вопросов предопределяет огромную потребность в высококвалифицированных специалистов, владеющих знаниями и умениями в сфере технического и программного обеспечения автоматизированных систем.

Целью лабораторной работы является изучение принципов построения автоматизированной системы управления процессом дозирования и перекачивания жидких продуктов на базе программируемого контроллера компании OMRON и сенсорного терминала фирмы DELTA, исследование работы названной системы и приобретение навыков в разработке программ управления для контроллера [1-3].

1 ОПИСАНИЕ ЛАБОРАТОРНОГО СТЕНДА

1.1 Назначение и функциональные возможности лабораторного стенда

Лабораторный учебный стенд позволяет изучить процесс дозирования и перекачивания жидких продуктов, а также получить практические навыки по программированию контроллера компании OMRON и терминала фирмы DELTA.

В набор стандартных функций и возможностей лабораторного учебного стенда входят:

- ручной и автоматический режимы работы;
- поочередное перекачивание жидкости между емкостями;
- измерение уровня жидкости в емкостях;
- визуализация технологического процесса средствами терминала;
- возможность модифицировать демонстрационную базовую прошивку контроллера;
- возможность программно реализовать разнообразные задачи, связанные с технологическими процессами дозирования и перекачки.

На боковой панели стенда установлен оснащенный ключом

переключатель. С его помощью можно отключить ручной режим, остановить работу насосов стенда и предотвратить перегрев электродвигателей.

1.2 Состав и принцип работы стенда

Лабораторный учебный стенд состоит из двух основных частей: блок управления и объект управления (рисунок 1).



Рисунок 1 – Общий вид лабораторного стенда

В состав **блока управления** входят:

- 1 Система питания
- 2 Система управления, состоящая из:
 - программируемого контролера модели CP1L компании OMRON;
 - сенсорного терминала серии DOP-B фирмы DELTA;
 - блока реле;
 - переключателей и сигнальных светодиодов.

В состав **объекта управления** входят:

- 1 Емкости
- 2 Дискретные емкостные датчики уровня
- 3 Аналоговые емкостные датчики уровня
- 4 Насосы
- 5 Трубопроводы

Дальнейшее описание принципа работы стенда приведено с привязкой к базовому алгоритму управления, реализованному в виде демонстрационной базовой прошивки контроллера и терминала.

Три рабочие емкости заполнены жидкостью (синтетическим маслом). Лабораторный стенд имеет два режима работы: ручной и автоматический.

В ручном режиме возможно управление перекачиванием жидкости из одной емкости в другую может выполняться при помощи переключателей, расположенных на лицевой панели стенда. Перекачивание жидкости может производиться только в определенном порядке: из первой емкости во вторую, из второй в третью и из третьей в первую.

В автоматическом режиме управление процессом перекачивания жидкости осуществляется контроллером в соответствии с загруженной в него программой. Для начала работы в автоматическом режиме нужно выполнить несколько условий. Во-первых, нужно задать уровень во второй и третьей емкостях (в литрах). Во-вторых, нужно «вручную» при помощи переключателей, не выходя из автоматического режима, наполнить первую емкость до момента срабатывания дискретного датчика верхнего уровня. После выполнения этих двух условий можно запускать перекачивание в автоматическом режиме. Процесс перекачивания будет протекать по следующему базовому алгоритму: включается первый насос и наполняет вторую емкость жидкостью до заданного уровня, затем включается второй насос и перекачивает жидкость в третью емкость до заданного уровня, затем включается третий насос и перекачивает половину жидкости в первую емкость из третьей.

2 ПРОЕКТИРОВАНИЕ ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ

2.1 Базовая версия алгоритма работы системы

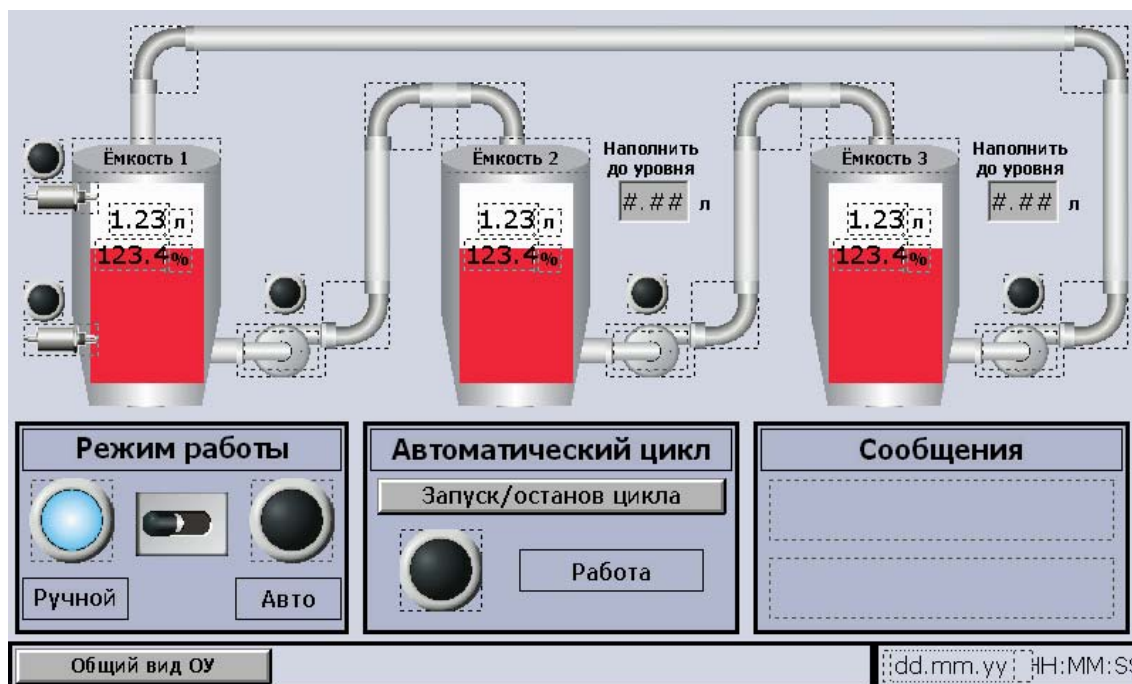
На экране терминала схематично отображается технологический процесс перекачки жидкости (рисунок 2).

В верхней части изображены три ёмкости с жидкостью, насосы, соединительные трубопроводы. При работе насоса лампа, индицирующая его состояние, светится зелёным, когда насос включен, при выключенном насосе она чёрная. Вместе с лампой изменяют цвет и соответствующие трубопроводы с нейтрального на красный цвет, а изображение насоса становится мерцающим.

На первой (левой) ёмкости показаны дискретные датчики уровня жидкости с индикаторами их состояния (индикатор светится зелёным, если сигнал приходит с датчика, иначе индикатор чёрный). На второй и третьей емкостях аналоговые датчики уровня не показаны.

На каждой ёмкости графически отображается уровень её заполнения. Кроме этого, отображается цифровое значение уровня, в литрах и процентах от полной ёмкости (2,2л). Т.к. аналоговые датчики уровня установлены только на второй и третьей емкостях, уровень в первой ёмкости вычисляется как разница

между полным объёмом жидкости в гидросистеме и суммой уровней во второй и третьей емкостях.



а



б

а) в состоянии элементов "0"; б) в состоянии элементов "1"
Рисунок 2 – Экран терминала DELTA

В нижней части экрана расположена панель управления, поделённая на три секции. В секции «Режим работы» можно выбирать между автоматическим и ручным режимом. В ручном режиме можно управлять насосами, нажимая на

их изображение. Насос включён, пока его изображение удерживается нажатым.

В автоматическом режиме реализован цикл последовательного перекачивания жидкости как демонстрация процесса управления насосами. Для запуска процесса нужно перевести переключатель «Ручной/Авто» в положение «Авто», после чего нажать кнопку «Запуск/останов цикла» в секции «Автоматический цикл». Необходимым условием начала работы демонстрационного алгоритма является наличие верхнего уровня в ёмкости 1. Если это условие не выполняется, в секции «Сообщения» появляется сообщение «Для начала наполните ёмкость 1 до верхнего уровня». Произвести перекачивание жидкости можно в ручном режиме. При наличии необходимого уровня в ёмкости 1 начинается работа демонстрационного алгоритма, что сигнализируется лампой «Работа» секции «Автоматический цикл». При этом ёмкость 2 наполняется до уровня, заданного в поле ввода «Наполнить до уровня, л», расположенного возле изображения ёмкости 2. Далее ёмкость 3 наполняется аналогично. Последним шагом является опустошение ёмкости 3 до половины значения уровня, заданного в поле ввода около ёмкости 3. На этом работа автоматического цикла прекращается. Лампа «Работа» гаснет, выводится сообщение «Конец автоматического цикла».

2.2 Основные сведения о программной среде разработки прикладных программ

Программный комплекс CX-Programmer – инструментальное средство, предназначенное для создания программ управления на языке лестничных диаграмм (LD), выполняемых в ПЛК компании OMRON [2, 3]. Помимо программирования контроллеров комплекс CX-Programmer обеспечивает и другие функции, необходимые при настройке и работе с ПЛК, среди которых: отладка программ, отображение адресов и значений, настройка и мониторинг ПЛК, а также дистанционное программирование и мониторинг по сети.

2.3 Пример №1 – Разработка прикладной программы управления процессом перекачки с заданием временных интервалов

Внимание! Перед началом выполнения примеров, убедитесь, что ёмкость 1 заполнена до верхнего уровня! Следует избегать длительного включения насосов, т.к. это приведет к их перегреву и дальнейшему выходу из строя!

В данном примере рассмотрены основы создания LD-программ для управления процессом перекачки.

Задача: В автоматическом режиме перекачать жидкость из ёмкости 1 в ёмкость 2 в течение заданного времени, используя инструкцию TIM (таймер).

Адреса, используемые в программе:

1.00 – виртуальный адрес нормально разомкнутого (замыкающего) контакта;

T0000 – таймер №0;

100.00 – насос 1 (физический выход контроллера).

Рассмотрим последовательность действий при создании LD-программы для данного примера.

Для того, чтобы насос начал работать только тогда, когда это требуется, необходимо создать кнопку пуска. Для этого нажимаем на клавиатуре кнопку [C].

Откроется диалоговое окно New Contact (Создание нормально разомкнутого контакта) (рисунок 3).

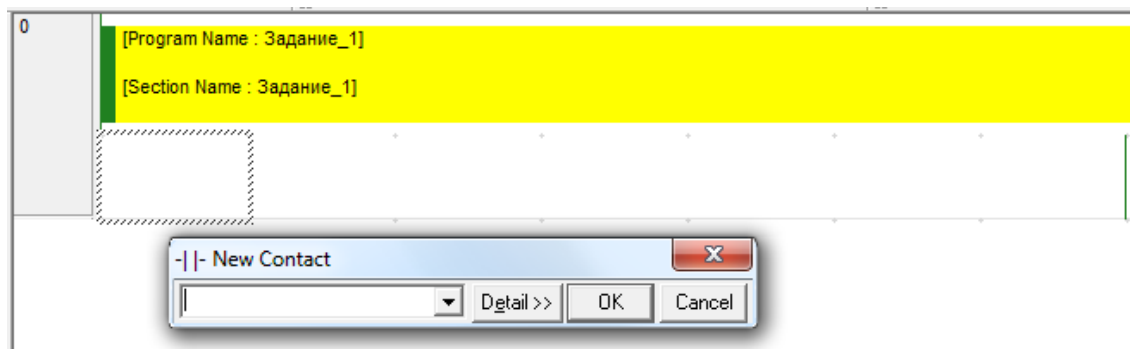


Рисунок 3 – Окно создания нормально разомкнутого контакта

Далее нам необходимо ввести адрес кнопки. Адрес может быть виртуальным, а может быть физическим (вход контроллера). В нашем случае нам необходим виртуальный адрес, например, 1.00 (можно вводить и 100, программа сама трансформирует значение).

Далее, после нажатия клавиши [Enter] или кнопки ОК, появится окно ввода комментария к данному элементу (рисунок 4).

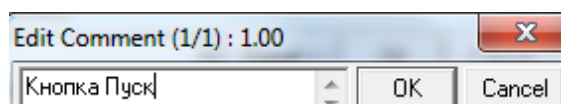


Рисунок 4 – Ввод комментария

Нажимаем клавишу [Enter] или кнопку ОК (рисунок 5).

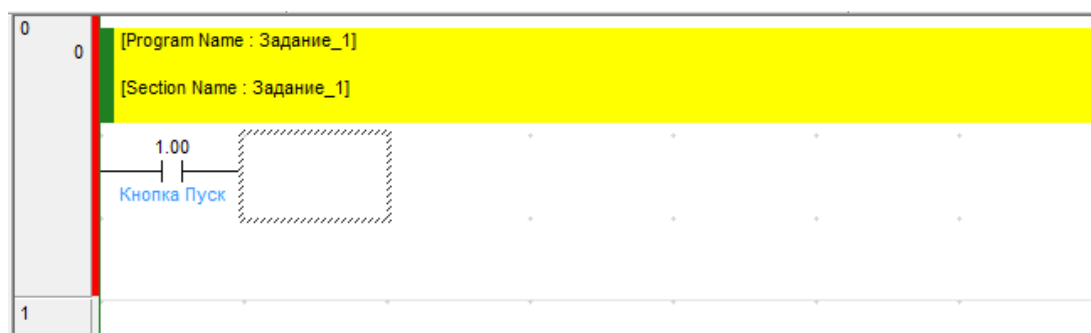


Рисунок 5 – Результат создания кнопки пуска

Инструкция ТИМ для программирования таймеров

Рассмотрим инструкцию ТИМ (Таймер) (рисунок 6).

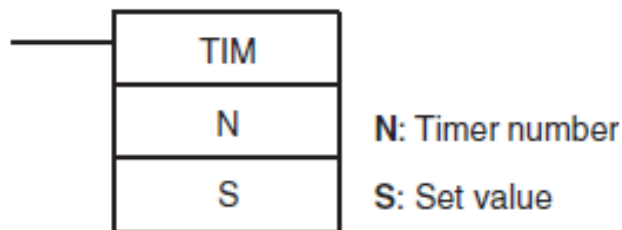


Рисунок 6 – Инструкция TIM

N: Timer Number (Номер таймера) – от 0000 до 4095

S: Set Value (Задание значения) - от #0000 до #9999

Таймер TIM работает как декрементный таймер с шагом 100 мс. Диапазон настройки значения (set value) от 0 до 999,9 с.

Таймер является конечной инструкцией в строке программы, поэтому добавляется в конце строки.

Нажмите клавишу [I]. Отобразится диалоговое окно «New Instruction» (Создание команды) (рисунок 7)..



Рисунок 7 – Окно создания команды

Введите команду таймера «TIM 0 #100». Нажмите клавишу [Enter] и введите комментарий. Сама инструкция TIM имеет шаг 100 мс. Команда «TIM 0 #100» соответствует таймеру задержки на 10 с, с флагом завершения таймера T0000 (рисунок 8).

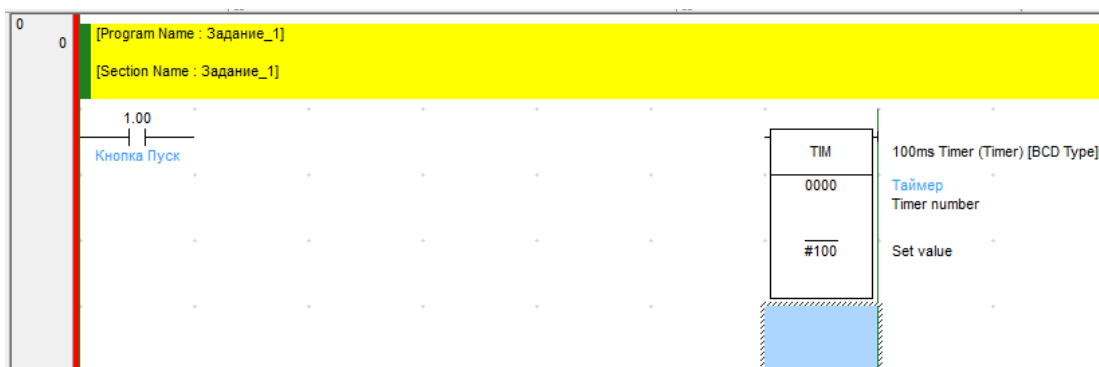


Рисунок 8 – Результат ввода таймера

Теперь необходимо соединить эти элементы. Для этого нажимаем ЛКМ на таймер и, зажав клавишу [Ctrl], нажимаем клавишу-стрелку [влево] (рисунок 9).

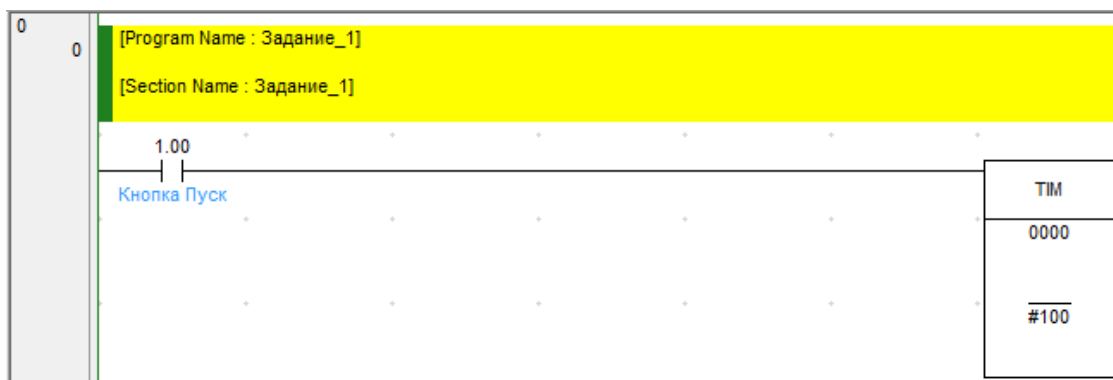


Рисунок 9 – Соединение элементов строки программы

Далее необходимо запустить насос 1 и «держат» его включенным, пока таймер не досчитает до нуля. Для этого переместите курсор под инструкцию таймера и нажмите клавишу [O]. Откроется диалоговое окно «New Coil» (Создание катушки) (рисунок 10).

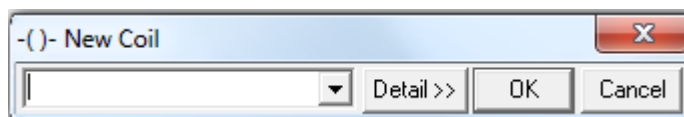


Рисунок 10 – Окно создания катушки

Введите адрес выхода контроллера, к которому подключен насос 1 «100.00» или «10000», нажмите клавишу [Enter] и введите комментарий Каждый дискретный выход контроллера нумеруется, начиная от 100.00 и заканчивая последним номером дискретного выхода контроллера (например, 100.04).

Катушка будет подавать сигнал на выход контроллера, который соединен с насосом 1 (рисунок 11).

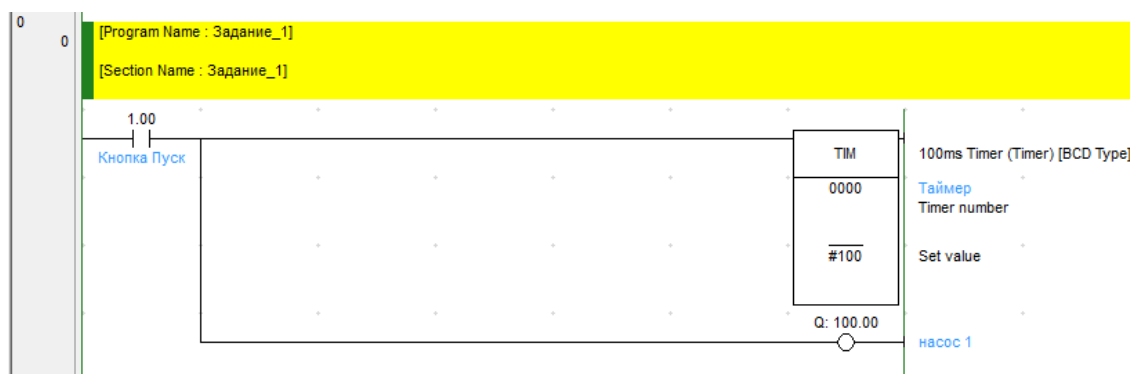


Рисунок 11 – Результат создания катушки

Теперь нужно, чтобы насос выключился после того, как таймер досчитает до нуля.

После нормально-разомкнутого контакта кнопки пуска, добавляем

нормально замкнутый контакт, который управляется таймером. Для этого нажмите клавишу [/]. Откроется окно «New Closed Contact» (Создание НЗ контакта) (рисунок 12).

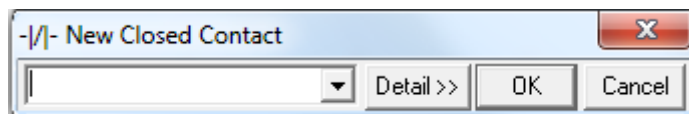


Рисунок 12 – Окно создания нормально замкнутого контакта

Введите адрес «Т0». Нажмите клавишу [Enter]. Далее введите в качестве комментария «Таймер» и нажмите «Enter». Затем соедините катушку 100.00 и НЗ контакт как показано на рисунке 13.

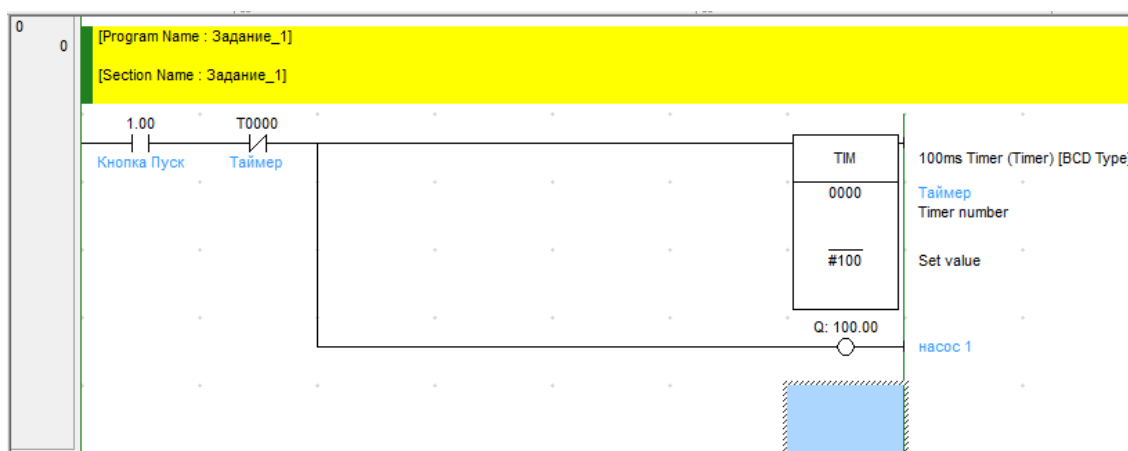


Рисунок 13 – Результат ввода нормально замкнутого контакта таймера

Казалось бы, программа готова, однако в ней не хватает ещё одного элемента. Если программа будет оставлена в текущем виде, то после нажатия кнопки пуска и её отпускания сигнал на таймер будет подан только на момент времени её «нажатия», а если «держат» кнопку зажатой дольше, чем заданное время таймера, то насос не отключится.

Для того, чтобы сигнал не прерывался, а насос отключился автоматически, необходимо ввести, параллельно кнопке пуска, НР контакт, который управляется выходом 100.00 (рисунок 14) и соединить его до НЗ контакта, управляемого таймером.

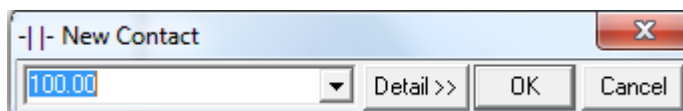


Рисунок 14 – Ввод нормально разомкнутого контакта 100.00

Программа готова (рисунок 15).

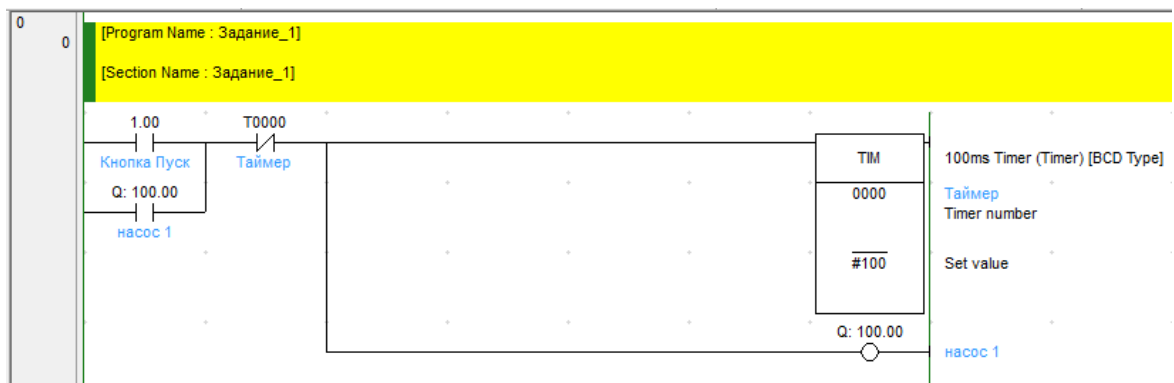


Рисунок 15 – Готовая LD программа

Теперь подробно рассмотрим, что происходит в данной программе.

После нажатия кнопки пуска и её отпускания, сигнал (логическая «1») проходит через НЗ контакт «Таймер» на инструкцию TIM и на выход 100.00 (на насос 1). Насос начинает перекачку жидкости из первой Емкости в Емкость 2, при этом таймер начинает отсчет времени от 10 с до 0. Когда таймер «досчитает» до 0, то на его выходе появится сигнал, который разомкнет НЗ контакт «Таймер» (флаг завершение T0000) и выключит насос. Программа будет выполнена.

Имитатор режима работы онлайн

Теперь, чтобы убедиться в работоспособности программы, воспользуемся имитатором работы контроллера. Для этого нажмите клавиши [Ctrl]+[Shift]+[W] (рисунок 16).

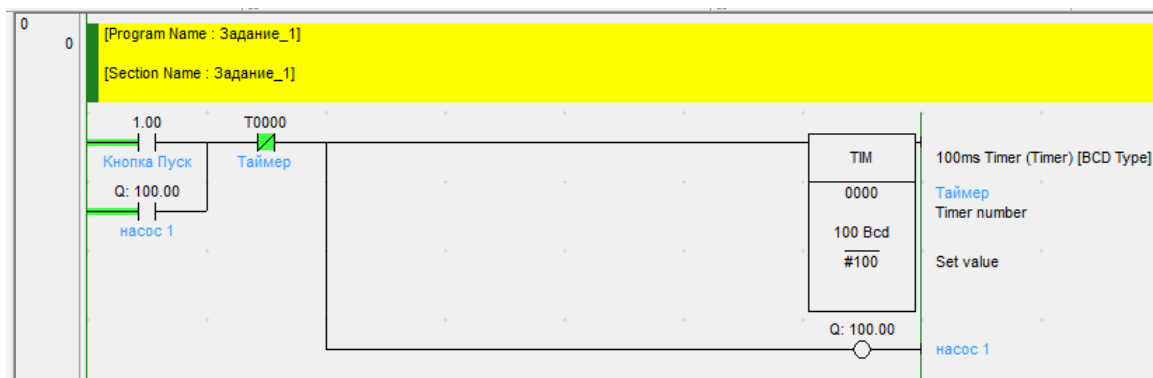


Рисунок 16 – LD программа в режиме имитатора

Как видно из рисунка 16, программа «ждет» нажатия кнопки пуска - таймер не функционирует и насос не работает. Для того чтобы нажать кнопку пуска, переместите курсор на НР контакт 1.00 и нажмите клавишу [Enter], после чего появится окно «Задание нового значения» (Set New Value) (рисунок 17), в котором нужно ввести значение «1» и нажать [Enter], для того, чтобы замкнуть контакт кнопки пуска.

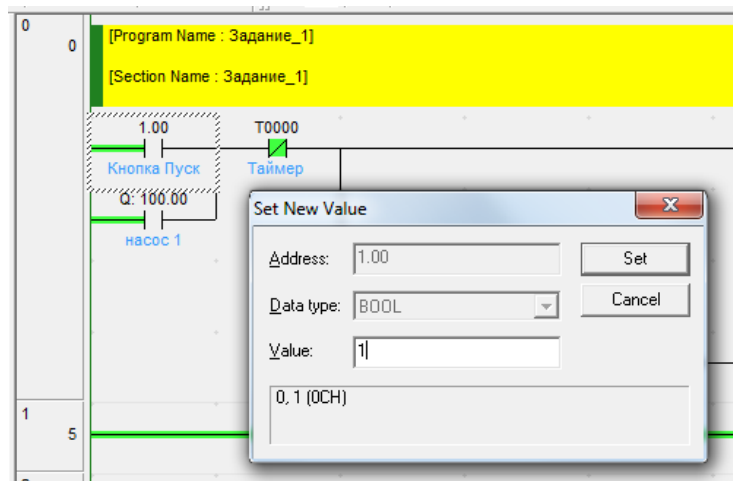


Рисунок 17 – Ввод значение сигнала «Пуск»

Как видно сигнал пуск активировал и таймер, и насос 1 (рисунок 18).



Рисунок 18 – Активация кнопкой пуска таймера и насоса 1

Однако, т.к. мы находимся в режиме симуляции, нам необходимо имитировать «отпускание» кнопки пуска, поэтому, сразу же после присвоения НР контакту 1.00 значения «1» нужно снова присвоить значение «0» (рисунок 19).



Рисунок 19 – Имитация размыкания контакта кнопки пуска

Теперь насос отключится в автоматическом режиме после срабатывания таймера (рисунок 20).

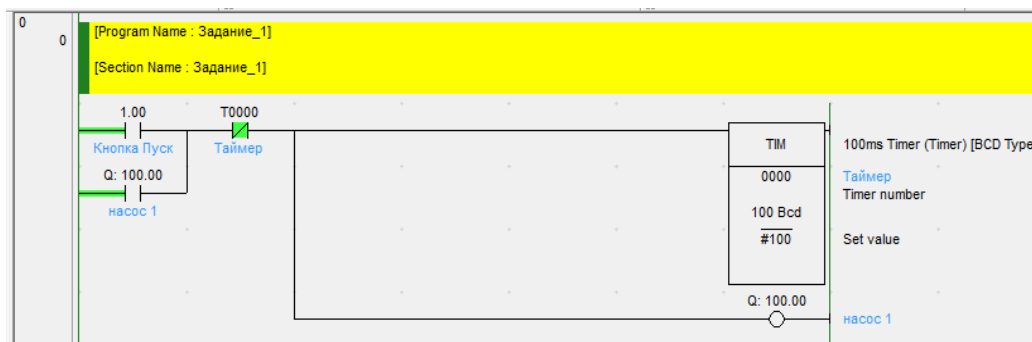


Рисунок 20 – Выключение насоса 1 после срабатывания таймера

2.4 Пример №2 – Разработка прикладной программы управления процессом перекачки с заданием уровня жидкости

В данном примере необходимо ознакомимся с аналоговым датчиком уровня жидкости в емкости 2, научиться «считывать» и использовать сигналы с этого датчика, а так же научиться использовать инструкцию «Входное сравнение» (Input Comparison Instructions).

Задача: Перекачать жидкость из Емкости 1 в Емкость 2 до заданного уровня в автоматическом режиме.

Адреса, используемые в программе:

100.00 – насос 1 (физический выход контроллера);

1.00 – виртуальный адрес нормально разомкнутого (замыкающего) контакта;

D25 – заданный уровень для емкости 2;

D0 – Текущий уровень в емкости 2;

A643 – адрес аналогового сигнала с датчика уровня.

Диапазон значений сигнала с аналогового датчика уровня от 0 до 255.

Создание LD-программы

Аналогично примеру 1 создадим кнопку пуска и катушку включения насоса 1 (рисунок 21).

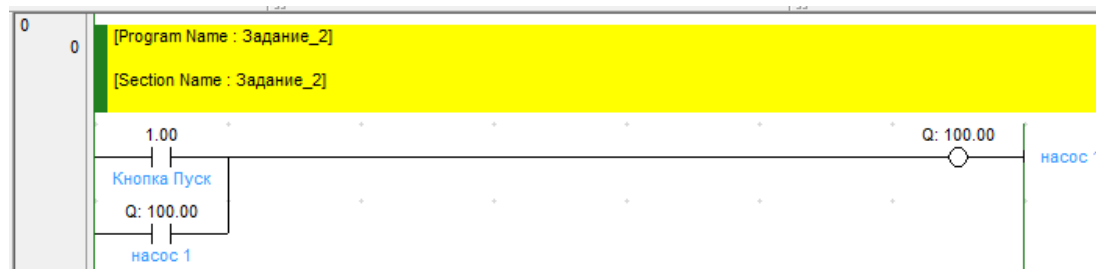


Рисунок 21 – Создание кнопки пуска и катушки включения насоса 1

Инструкция MOV передает слово данных в указанное слово (рисунок 22).

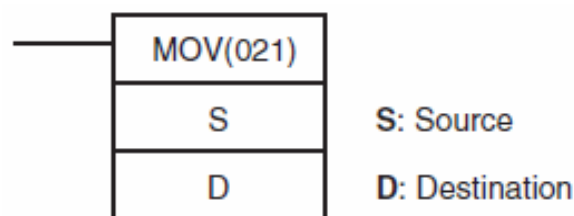


Рисунок 22 – Инструкция MOV

Используемые в инструкции MOV обозначения: S (Source) – слово данных, D (Destination) – указанное слово.

Чтение сигнала с аналогового датчика

В конце строки 1 добавляем инструкцию «MOV A643 D0» (рисунок 23).



Рисунок 23 – Создание инструкции MOV A643 D0

Инструкция MOV выполняет функцию переноса в ячейку памяти D0 текущего значения уровня жидкости. Это требуется для того, чтобы иметь возможность использовать один аналоговый вход для нескольких аналоговых датчиков уровня (более подробно этот вопрос рассмотрен ниже в примере 3).

A643 является адресом, соответствующим аналоговому входу контроллера. Т.к. нам требуется читать значение с датчика уровня в реальном времени, то просто установим флаг «P_On», который всегда (в соответствии с тактовой частотой контроллера) подает сигнал на инструкцию MOV. Для этого нажмите клавишу [C] и введите «P_On», поместив контакт в начало шага (рисунок 24).

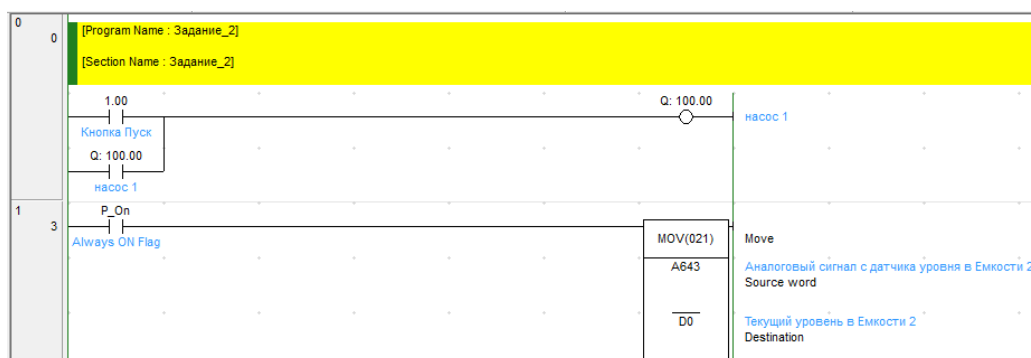


Рисунок 24 – Добавление инструкции MOV A643 D0 в программу

Создание условия наполнения Емкости 2

По условию задачи требуется выключить насос 1, в том случае, если Емкость 2 заполнена до нужного уровня. Это реализуется с помощью инструкции «Входное сравнение».

Инструкция входного сравнения показана на рисунке 25.

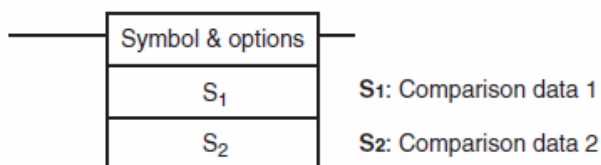


Рисунок 25 – Инструкция входного сравнения

Используемые в инструкции входного сравнения обозначения: S1: Comparison data 1 – слово сравнения 1, S2: Comparison data 2 - слово сравнения 2.

Инструкция входного сравнения сравнивает два значения (константы и/или содержание указанных слов) и имеют на выходе сигнал «1», когда условие сравнения – «ИСТИНА».

В нашем случае, мы воспользуемся инструкцией меньше «<». В шаге 0, сразу после кнопки пуска, добавляем инструкцию «<», введя команду «< D0 D25», где D25 – заданное значение уровня (рисунок 26).

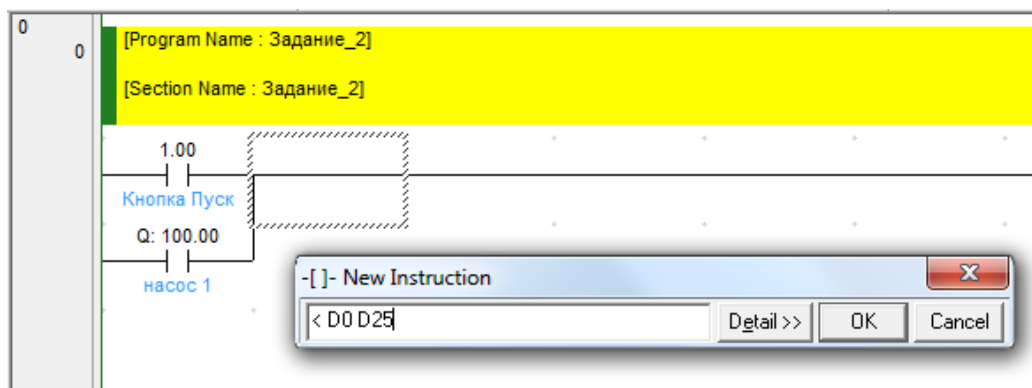


Рисунок 26 – Добавление инструкции «< D0 D25» в программу

Программа готова (рисунок 27).

Инструкция «<» сравнивает значение D0 и D25. Если D0 < D25, то на выходе инструкции будет «ИСТИНА», если D0 = или > D25, то на выходе инструкции будет «ЛОЖЬ».

Особенности симуляции в данном примере

Сразу после запуска симулятора нужно задать значения A643 и D25, и только потом, нажимать кнопку пуск аналогично примеру 1.

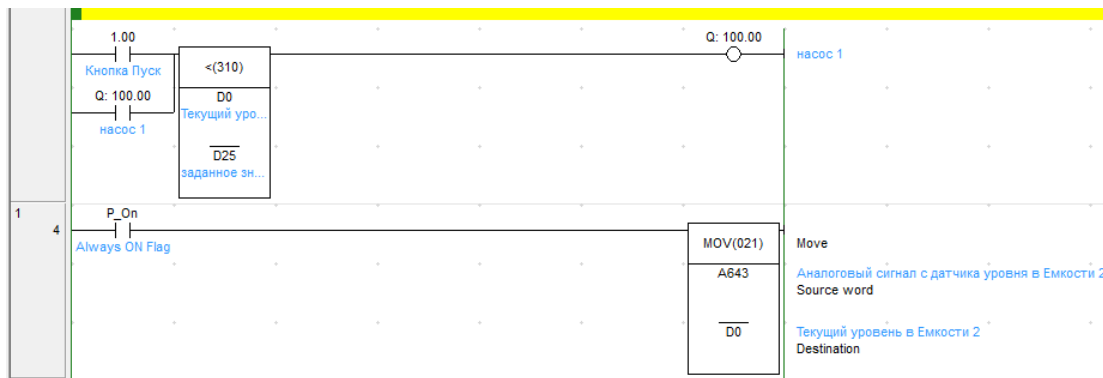


Рисунок 27 – Готовая программа

Нам нужно выполнить имитацию сигнала с датчика уровня. Для этого в инструкции MOV наводим курсор на значение A643, нажимаем [Enter] и вводим 0 – Емкость 2 пуста.

Инструкция MOV (рисунок 28) отображает данные в формате HEX (16-разрядная система счисления).

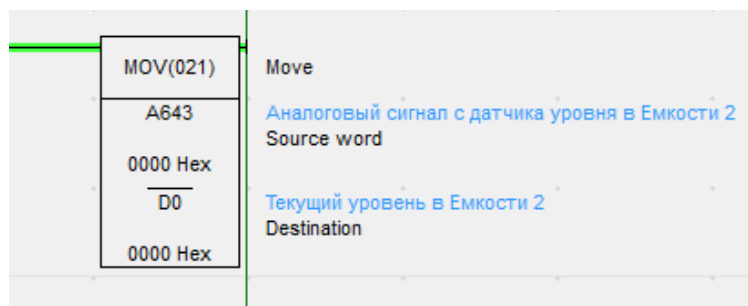


Рисунок 28 – Отображение данных инструкцией MOV

Теперь зададим требуемый уровень. Для этого в инструкции «<» наводим курсор на значение D25, нажимаем [Enter] и вводим, например, 50.

Нажимаем кнопку пуск. Насос 1 начал перекачку в емкость 2, т.к. уровень жидкости в ней меньше, чем заданный (рисунок 29).

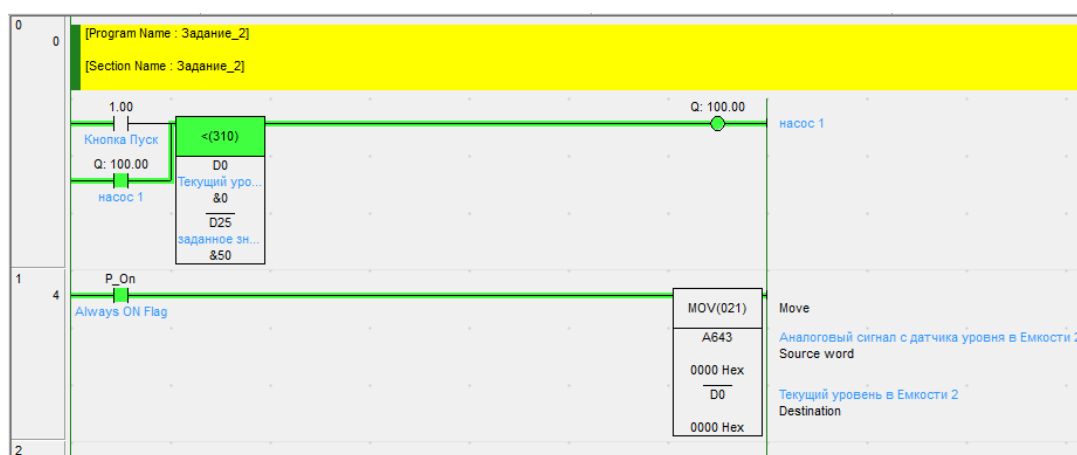


Рисунок 29 – Выполнение программы при значении уровня 0

Теперь зададим значение A643 равное 50 и убедимся, что насос 1 выключается, т.к. D0 = D25 (рисунок 30).

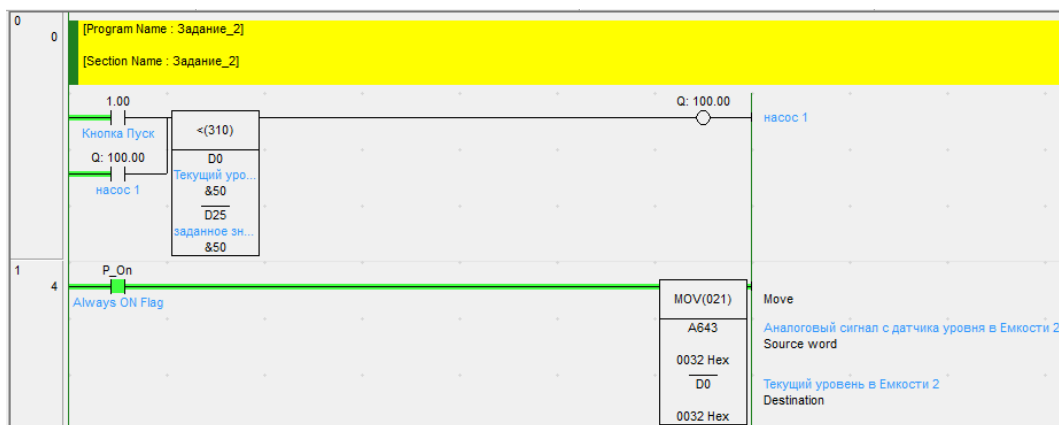


Рисунок 30 – Выполнение программы при значении уровня 50

2.5 Пример №3 – Разработка прикладной программы управления процессом дозирования с обеспечением коммутации аналоговых сигналов

В данном примере необходимо ознакомиться с аналоговым датчиком уровня в емкости 3 и дискретным датчиком нижнего уровня в емкости 1, научиться коммутировать сигналы с аналоговых датчиков, используя переключающее реле.

Задача: Требуется перекачать жидкость в емкость 3 до заданного уровня в автоматическом режиме.

Примечание: Необходимо исключить возможность работы насосов «вхолостую» при снижении уровня жидкости ниже минимального уровня.

Адреса, используемые в программе:

100.00 – насос 1;

100.01 – насос 2;

100.03 – реле коммутации аналоговых входных сигналов;

1.00 – виртуальный адрес НР контакта (кнопка Пуск);

2.00 – виртуальный адрес НЗ контакта (кнопка Стоп);

D30 – заданный уровень для емкости 3;

D0 – текущий уровень в емкости 2;

D2 – текущий уровень в емкости 3;

T0000 – таймер задержки чтения и записи значений текущих уровней;

A643 – аналоговый сигнал с датчиков уровня;

0.01 – дискретный вход контроллера (дискретный датчик нижнего уровня в емкости 1).

Создание LD-программы

Для того чтобы иметь возможность в любой момент времени остановить работу насосов необходимо создать не только кнопку «Пуск», но и кнопку «Стоп». Для этого, аналогично примерам 1 и 2 создаем кнопку пуска и катушки

управления насосами 1 и 2.

Чтобы остановить работу насосов по желанию, сразу же после кнопки пуска, создадим кнопку «Стоп», добавив НЗ контакт, например, с адресом 2.00 (рисунок 31).

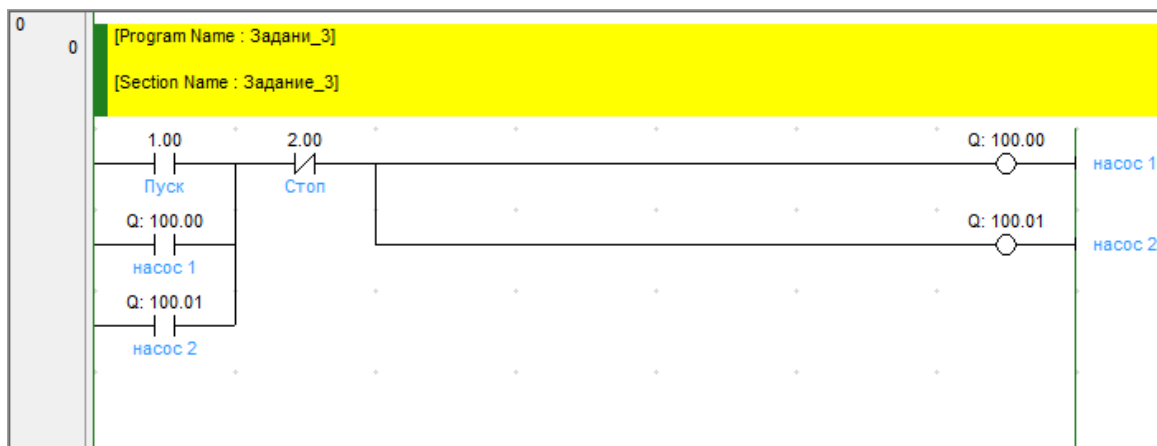


Рисунок 31 – Добавление в программу кнопки «Стоп»

Создание условия включения и отключения насосов 1 и 2

В примечании к данному примеру указано, что мы необходимо исключить возможность работы насосов «вхолостую».

Для насоса 1 создадим условие отключения по срабатыванию дискретного датчика нижнего уровня. Т.к. датчик, первоначально, уже находится в жидкости, с него будет идти сигнал «1». В этом случае, мы просто добавим НР контакт перед катушкой насоса 1, с адресом, соответствующим дискретному входу контроллера 0.01 (рисунок 32).

Для того чтобы создать условие отключения насоса 2, сначала необходимо прочитать данные с датчика уровня в емкости 2. Аналогично примеру 2 с помощью инструкции MOV считаем сигнал с датчика и запишем значение в область памяти D0 (рисунок 33).

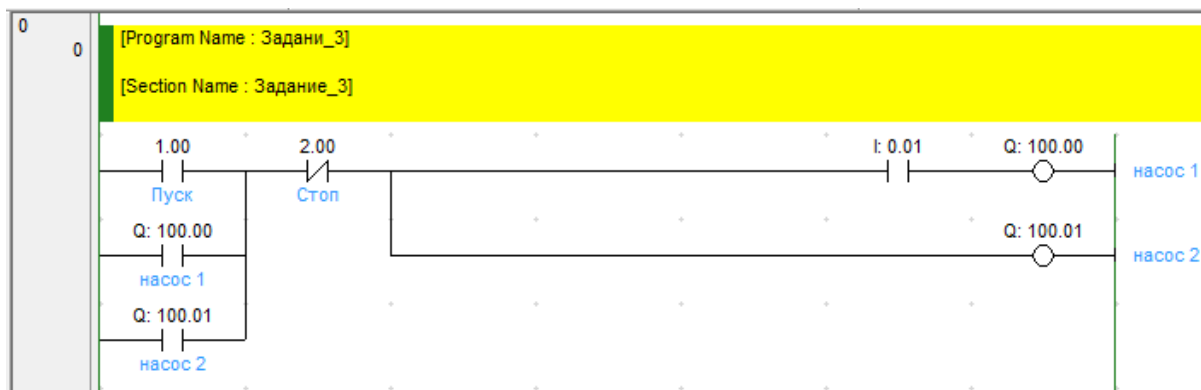


Рисунок 32 – Добавление в программу датчика нижнего уровня

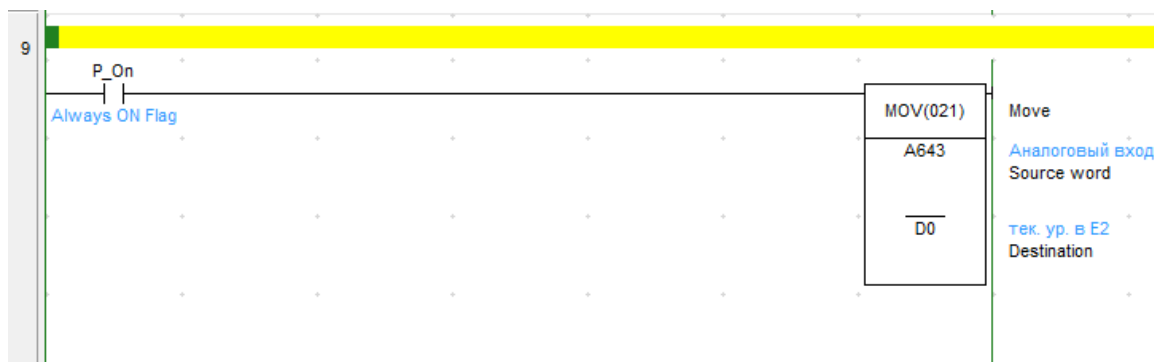


Рисунок 33 – Добавление инструкции MOV в программу

Теперь создадим условие отключения насоса 2. Для этого воспользуемся инструкцией входного сравнения «больше >». Инструкция «больше» работает аналогично инструкции «меньше»: если $D0 > 0$, то на выходе инструкции будет «Истина», если $D0 =$ или < 0 , то на выходе инструкции будет «Ложь».

Добавим эту инструкцию перед катушкой насоса 2 (рисунок 34).

Теперь нам нужно создать условие отключения обоих насосов после того, как емкость 3 будет заполнена до требуемого уровня, т.к. емкость 3 может быть заполнена раньше, чем сработает датчик нижнего уровня в емкости 1.

Ввод данных с двух аналоговых датчиков с помощью внешнего переключающего реле

Т.к. два аналоговых датчика подключены на один аналоговый вход контроллера через переключающее реле, то в каждый момент времени мы можем читать данные только с одного из датчиков.

Внешнее реле подключено и управляется через дискретный выход 100.03

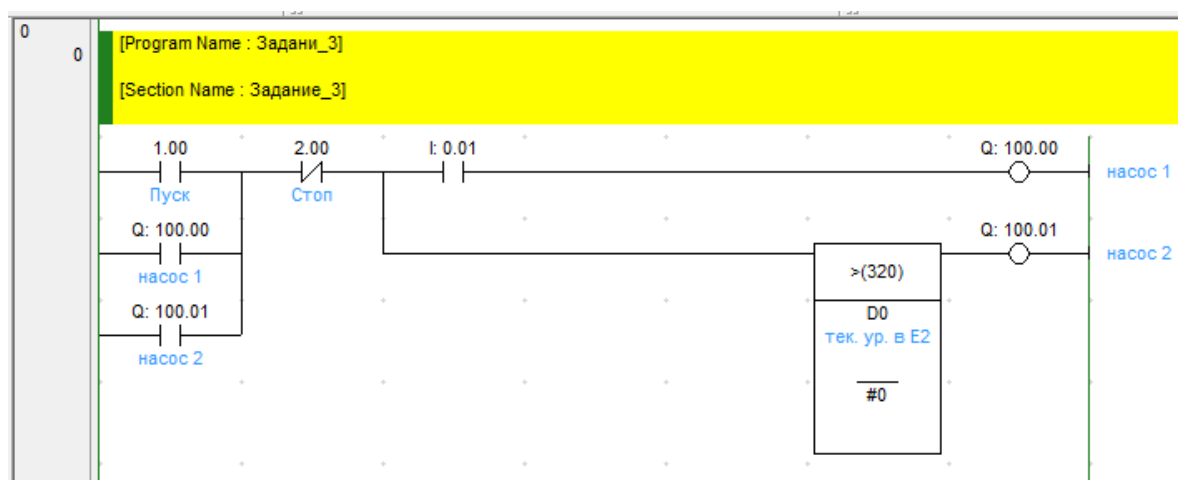


Рисунок 34 – Добавление в программу инструкции входного сравнения

Инструкция КЕЕР

Для того чтобы переключать внешнее реле поочередно с одного датчика

на другой, необходимо использовать инструкцию КЕЕР.

Инструкция КЕЕР имеет два входа S (Set) и R (Reset). Если на вход S подана логическая «1», то на выходе инструкции будет значение логическая «1», пока не будет активирован вход R (рисунок 35).

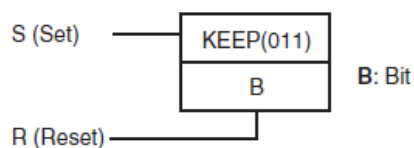


Рисунок 35 – Инструкция КЕЕР

Реле имеет ограниченный ресурс срабатывания, поэтому мы должны включать его только тогда, когда нам нужно использовать данные с датчиков. Это требуется только при работе насоса 2 (рисунок 36).

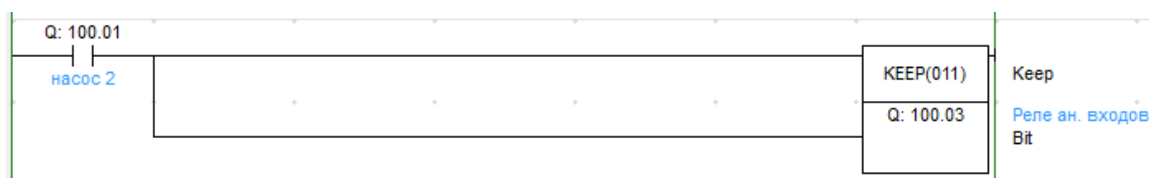


Рисунок 36 – Добавление в программу инструкции КЕЕР

Чтобы инструкция КЕЕР начала управление реле, необходимо создать условия присвоения логической «1» и сброса этого значения в «0».

Нельзя переключать реле слишком часто, т.к., во-первых, контроллер не будет успевать выполнять аналогово-цифровое преобразование при чтении сигнала с датчиков, во-вторых, это приведет к его преждевременному выходу из строя (т.к. ресурс количества переключений реле ограничен).

Эти условия реализуются с помощью НР контактов с использованием переднего и заднего фронта с флагом P_1s. P_1s – пульсирующий контакт, замыкающийся/размыкающийся каждые 0,5 с.

Передний и задний фронт

Положительный сигнал – это сигнал, активный уровень которого «Логическая 1». То есть нуль – это отсутствие сигнала, единица – наличие сигнал (рисунок 37).

Активный уровень сигнала – это уровень, соответствующий приходу сигнала, то есть выполнению этим сигналом соответствующей ему функции.

Пассивный уровень сигнала – это уровень, в котором сигнал не выполняет никакой функции.

Передний фронт сигнала – это переход сигнала из пассивного уровня в активный. Задний фронт сигнала – это переход сигнала из активного уровня в пассивный.

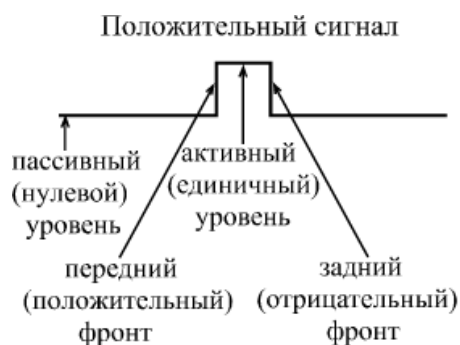


Рисунок 37 – Положительный сигнал

Для того чтобы включить опцию переднего или заднего фронта, в окне редактирования контакта нажмите кнопку «Detail >>» и выберите передний фронт (Up) для входа Set инструкции КЕЕР и аналогично задний фронт (Down) для входа Reset (рисунок 38 и 39).

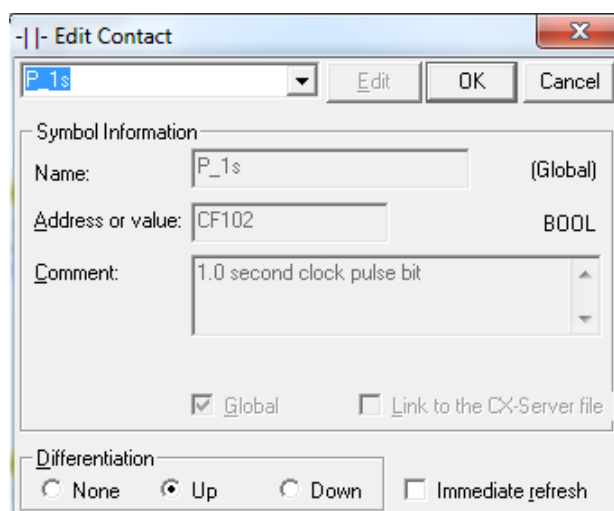


Рисунок 38 – Выбор пульсирующего контакта

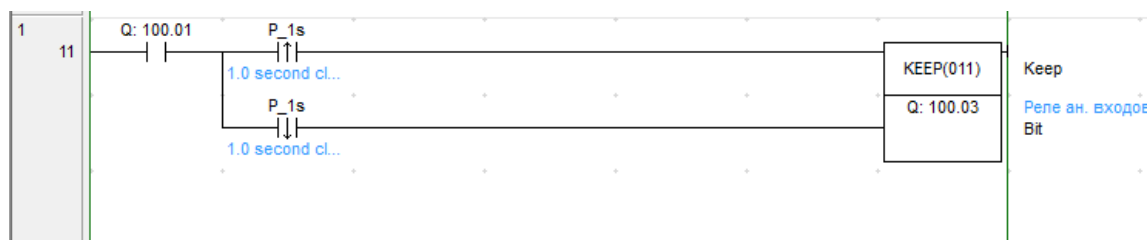


Рисунок 39 – Добавление в программу пульсирующего контакта

График работы НР контактов P_1s приведен на рисунке 36. Когда включается насос 2, на входы НР контактов P_1s приходит «1», т.к. верхний контакт активируется по переднему фронту, то первые 0,5 с значение «1» идет на вход Set инструкции КЕЕР. После размыкания контактов значение «1» идет уже на вход Reset, (нижний контакт активируется по заднему фронту) в течение следующих 0,5 с. Т.о, реле переключается каждые 0,5 с.

На рисунке 40 А, С – график флага P_1s, В – график импульса переднего фронта, D – график импульса заднего фронта.

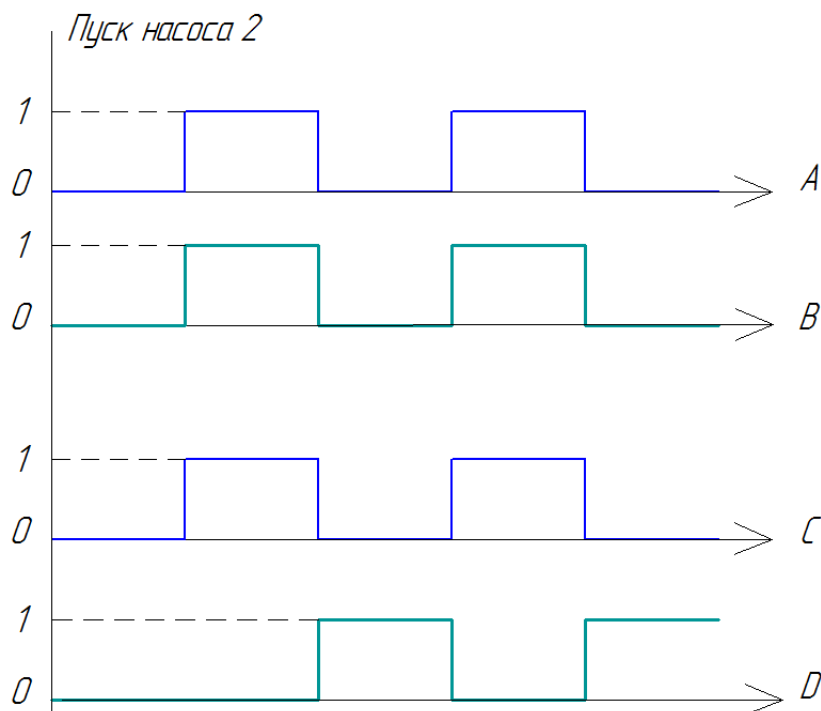


Рисунок 40 – Временные диаграммы работы пульсирующего таймера

Таймер переключения аналоговых входов

Для того, чтобы получать достоверные данные с датчиков с минимальной задержкой, нужно установить таймер на 200 мс (это время подбирается экспериментально). Таймер необходимо сбрасывать в промежутке между переключениями реле.

Для этого, установим два НЗ контакта с адресом выхода 100.03 инструкции КЕЕР, используя передний и задний фронт (рисунок 41).



Рисунок 41 – Добавление в программу таймера на

Чтение данных с датчиков

Аналогично примеру 2 воспользуемся инструкцией MOV для записи сигналов с датчиков в ячейки D0 и D2. Т.к. реле в данный момент времени находится в одном из положений (0 или 1), необходимо читать и записывать значения только одного датчика. Для этого воспользуемся контактами типа НЗ и НР (рисунок 42).



Рисунок 42 – Чтение данных с датчиков

T0000 создает задержку чтения данных. Программа готова (рисунок 43).

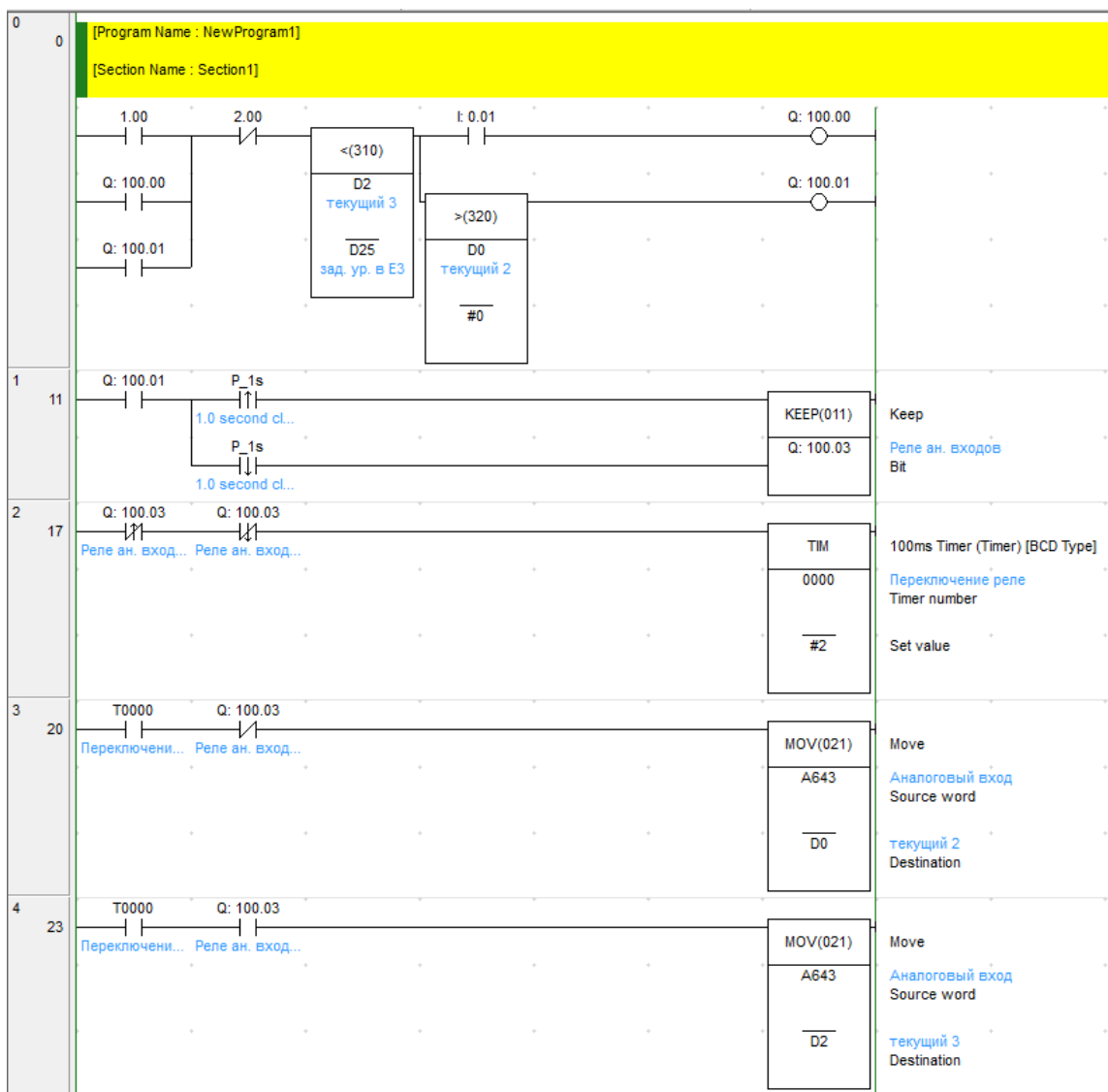


Рисунок 43 – Готовая программа

2.6 Дополнительные приемы проектирования программ управления

При разработке прикладных программ управления часто возникает необходимость использования рабочей области контроллера. Примером такой

задачи может служить создание LD-программы, в которой необходимо использовать входной дискретный сигнал от кнопки без фиксации.

В этом случае необходимо создать две кнопки: кнопку «Пуск» и кнопку «Стоп» (рисунок 44).

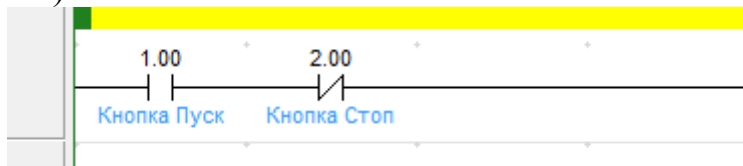


Рисунок 44 – Цепь программы с кнопками «Пуск» и «Стоп»

Далее, в конце строки, добавьте катушку с адресом W0 (рисунок 45), где W0 – рабочая область контроллера.



Рисунок 45 – Добавление в цепь катушки W0

Для того чтобы сигнал «1» продолжал идти на рабочую область после кратковременного нажатия кнопки «Пуск», добавим схему «ИЛИ» под НР контактом (рисунок 46).

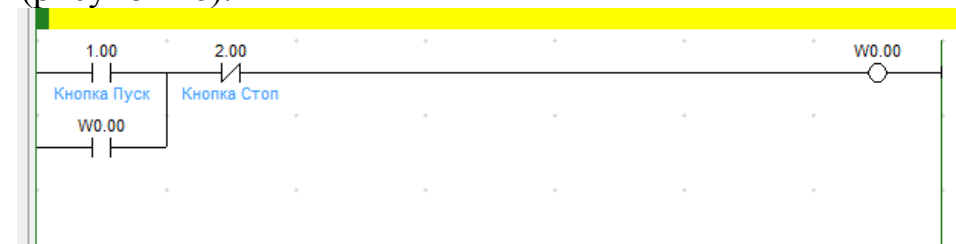


Рисунок 46 – Добавление в программу схемы «ИЛИ»

Используя режим имитации, проверить в выполнение программы (рисунок 47).



Рисунок 47 – Проверка программы в режиме имитации

3 ВАРИАНТЫ ЗАДАНИЙ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

В каждом задании первоначально емкость 1 заполнена до верхнего уровня.

Перечень адресов контроллера, используемых в стенде:
100.00 – насос 1; 100.01 – насос 2; 100.02 - насос 3; 100.03 – реле коммутации аналоговых сигналов; A643 – аналоговый сигнал с датчика уровня; 0.01 –

дискретный датчик нижнего уровня; 0.03 – дискретный датчик верхнего уровня.

Задание №1

Алгоритм:

1. Перекачивать жидкость из емкости 1 в емкость 2 в течение 15 с.
2. Ждать 5 с.
3. Перекачивать жидкость из емкости 2 в емкость 3 в течение 10 с.

Дополнительное условие:

1. Не допускать работы насосов «вхолостую».

Примечание: значения таймеров могут быть произвольными.

Задание №2

Алгоритм:

1. Перекачать жидкость из емкости 1 в емкость 2 до заданного уровня 1 и выключить насос 1.
2. Ждать 20 с.
3. Перекачать жидкость из емкости 2 до заданного уровня 2 в емкость 3.

Дополнительные условия:

Уровни 1 и 2 задаются произвольно.

Уровень 1 > Уровень 2.

Примечания:

1. Максимальное значение уровня в емкости 2 равно 255 (максимальный уровень сигнала с аналогового датчика).
2. В данной задаче используется сигнал только с аналогового датчика в емкости 2.

Задание №3

Алгоритм:

1. Перекачать жидкость в емкость 2 до заданного уровня в автоматическом режиме.

Дополнительные условия:

1. Общий объем жидкости неизвестен.
2. Неизвестен объем жидкости в каждой емкости.
3. Не допускать работы насосов «вхолостую».
4. Избежать возможного переполнения емкостей.

Примечание: в данной задаче используются все датчики.

4 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Лабораторную работу необходимо выполнять в следующей последовательности:

- 1 Пользуясь данными методическими указаниями, изучить функциональные возможности, состав и конструкцию лабораторного стенда (раздел 1).
- 2 Включить лабораторный стенд и проверить работу автоматизированной системы при выполнении базового алгоритма.
- 3 Запустить на компьютере программный комплекс CX-Programmer.
- 4 Изучить методику проектирования прикладных программ для

автоматизированной системы управления перекачиванием и дозированием, и отработать в среде CX-Programmer все примеры, приведенные в разделе 2.

5 Получить у преподавателя вариант задания на выполнение лабораторной работы.

6 Используя программный комплекс CX-Programmer, разработать LD-программу управления.

7 Исследовать разработанную программу, используя имитатор режима Online.

8 Выполнить подключение контроллера к компьютеру и подать напряжение питания на лабораторный стенд.

9 Загрузив LD программу в ПЛК, промоделировать на лабораторном стенде работу автоматизированной системы в режим Online симулятора.

10 Остановить выполнение программы в ПЛК и отключить питание.

11 Оформить отчет по лабораторной работе.

5 СОДЕРЖАНИЕ ОТЧЕТА

В отчете указывается цель лабораторной работы, и приводятся следующие результаты:

1 Задание на разработку программы управления для контроллера CP1L в программном комплексе CX-Programmer.

2 Копии экранов PrtScr (Print Screen) с рабочей областью CX-Programmer, в которой созданы лестничные диаграммы (LD-программы), реализующие заданные алгоритмы управления.

3 Выводы по результатам исследования программ в имитаторе режима Online и моделирования работы системы управления на лабораторном стенде.

6 КОНТРОЛЬНЫЕ ВОПРОСЫ

1 Каковы назначение и функциональные возможности лабораторного стенда?

2 Какова структура лабораторного стенда?

3 Как в LD-программе для контроллера CP1L задают таймеры?

4 Как в LD-программе для контроллера CP1L задают чтение сигнала с аналогового датчика уровня?

5 Каким образом в комплексе CX-Programmer можно проверить правильность созданных программ без их загрузки в ПЛК?

6 Каково назначение инструкции MOV в программе управления?

7 Каково назначение инструкции KEEP в программе управления?

8 Каким образом в лабораторном стенде организовано управление внешним реле, коммутирующим аналоговые сигналы датчиков уровня?

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Петров И.В. Программируемые контроллеры. Стандартные языки и инструменты. – М.: СОЛОН-Пресс, 2003. – 256 с.

2 <http://industrial.omron.ru>

3 Модуль ЦПУ CP1L: Вводное руководство. – Omron, 2007. – 165 с.

Сбродов Николай Борисович

**ИССЛЕДОВАНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ
УПРАВЛЕНИЯ ПЕРЕКАЧИВАНИЕМ И ДОЗИРОВАНИЕМ ЖИДКИХ
ПРОДУКТОВ**

Методические указания
к выполнению лабораторной работы
по дисциплине «Автоматизация технологических процессов и производств»
для студентов очной и заочной форм обучения направления 220700.62
«Автоматизация технологических процессов и производств»

Авторская редакция

Подписано к печати 03.02.14	Формат 60x84 1/16	Бумага тип. №1
Печать цифровая	Усл.печ.л. 1,75	Уч.-изд.л. 1,75
Заказ 38	Тираж 45	Не для продажи

Редакционно-издательский центр КГУ.
640669, г. Курган, ул. Гоголя, 25.
Курганский государственный университет.