

ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ТЮМЕНСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ТЮМЕНСКОЙ ОБЛАСТИ "ТЮМЕНСКИЙ КОЛЛЕДЖ ПРОИЗВОДСТВЕННЫХ
И СОЦИАЛЬНЫХ ТЕХНОЛОГИЙ"
(ГАПОУ ТО «ТКПСТ»)

А. А. Ашихмин, А. В. Козлов

ПРОГРАММИРОВАНИЕ ЛОГИЧЕСКИХ РЕЛЕ

Методические указания к выполнению практических работ

Для специальностей

08.02.09 Монтаж, наладка и эксплуатация электрооборудования промышленных и
гражданских зданий,

13.02.09 Техническая эксплуатация и обслуживание электрического и
электромеханического оборудования (по отраслям)

Тюмень, 2020

А. А. Ашихмин, А. В. Козлов Методические указания для выполнения практических работ по программированию логических реле. Для специальностей 08.02.09 Монтаж, наладка и эксплуатация электрооборудования промышленных и гражданских зданий, 13.02.09 Техническая эксплуатация и обслуживание электрического и электромеханического оборудования (по отраслям) / ГАПОУ ТО «ТКПСТ». – Тюмень, 2020 – 41 с.

Методические указания по выполнению практических работ включают в себя обзор ПО для программирования программируемых логических реле, а также задания для самостоятельного выполнения студентами.

Материалы составлены в соответствии с ФГОС СПО по специальностям 08.02.09 Монтаж, наладка и эксплуатация электрооборудования промышленных и гражданских зданий и 13.02.09 Техническая эксплуатация и обслуживание электрического и электромеханического оборудования (по отраслям), а также с учетом особенностей обучения в ГАПОУ ТО «ТКПСТ».

Адресовано студентам и преподавателям техникумов и колледжей технического профиля.

Печатается по решению заседания предметно-цикловой комиссии Монтаж и эксплуатация электрооборудования, протокол № 01 от 20 января 2020 года.

СОДЕРЖАНИЕ

1 Основные сведения о программируемом реле ОВЕН.....	4
2 Функции логических элементов программы	8
А.1 Битовые операции	8
А.2 Арифметические операции	10
А.3 Операции сравнения и выбора.....	14
А.4 Операции логического битового сдвига и преобразования ..	15
3 Функциональные блоки программы	19
4 Последовательность работы над проектом	26
5 Примеры выполнения задания.....	26
6 Создание коммутационной программы	27
7 Варианты заданий	37
8 Порядок выполнения работы.....	40
Литература	41

1 Основные сведения о программируемом реле ОВЕН

В настоящее время для решения простейших задач автоматизации в области бытовой техники, машино - и приборостроении еще очень часто используются традиционные реле и контакторы.

В 2009 году Российской фирмой ОВЕН был начат выпуск программируемых реле для дискретных локальных систем автоматизации ОВЕН ПР110 на базе среды программирования OWEN Logic.

ОВЕН ПР – это свободно программируемое устройство, которое не содержит в своей памяти заранее написанной программы. Алгоритм работы программируемого реле формируется непосредственно пользователем, что делает прибор универсальным и дает возможность широко использовать его в различных областях промышленности, сельском хозяйстве, ЖКХ и на транспорте. Специалисты ОВЕН рекомендуют использовать приборы данной линейки при замене устаревших релейных систем защиты и контроля. За счет внутренней логики прибора можно значительно сократить количество коммутируемых электромагнитных устройств, что снизит затраты на проектирование и эксплуатацию систем, а также повысит их надежность.

Программирование ПР не требует специальных навыков, поскольку осуществляется с помощью простой и интуитивно понятной среды программирования.

В простейший логический модуль были встроены 8 цифровых входов и 4 цифровых выхода, элементы индикации и обслуживания, блок питания на 24 В постоянного тока или 115 В или 230 В переменного тока, а также 14 самых распространенных на практике логических функций. Эти 14 функций подразделяются на 6 базовых функций (AND, OR, NOT, NAND, NOR, XOR) и 8 специальных функций (задержка включения и выключения, импульсное реле, выключатель с часовым механизмом, реле с самоблокировкой, тактовый генератор, задержка включения с памятью, а также прямой и реверсивный счетчики).

ОВЕН ПР объединяет эти разнообразные функции в одном устройстве. Это позволяет сократить до минимума затраты на электрический монтаж. Вместо того, чтобы соединять друг с другом отдельные устройства (например, такие как вспомогательные контакторы, реле или счетчики) множеством проводов, нужные Вам логические функции, а также взаимосвязи между ними просто программируются в одном единственном логическом модуле.

Создаваемые схемы управления можно в любой момент быстро и легко изменить. Для этого не нужны дополнительные затраты на электрический монтаж.

Если некоторая программа управления используется многократно, она может быть скопирована на специальный съемный модуль памяти и перенесена в другой логический модуль.



Рис.1.1 - Программируемое реле ПР 110

Программируемое реле подходит для задач локальной автоматизации с линейным или слаборазветвленным алгоритмом управления, например:

- Системы контроля доступа и релейной защиты,
- Реализация автоматов АВР,
- Управление наружным и внутренним освещением, освещением витрин,
- Управление технологическим оборудованием (насосами, вентиляторами, компрессорами, прессами),
- Реализация конвейерных систем,
- Управление подъемниками, парковочными автоматами и т.д.

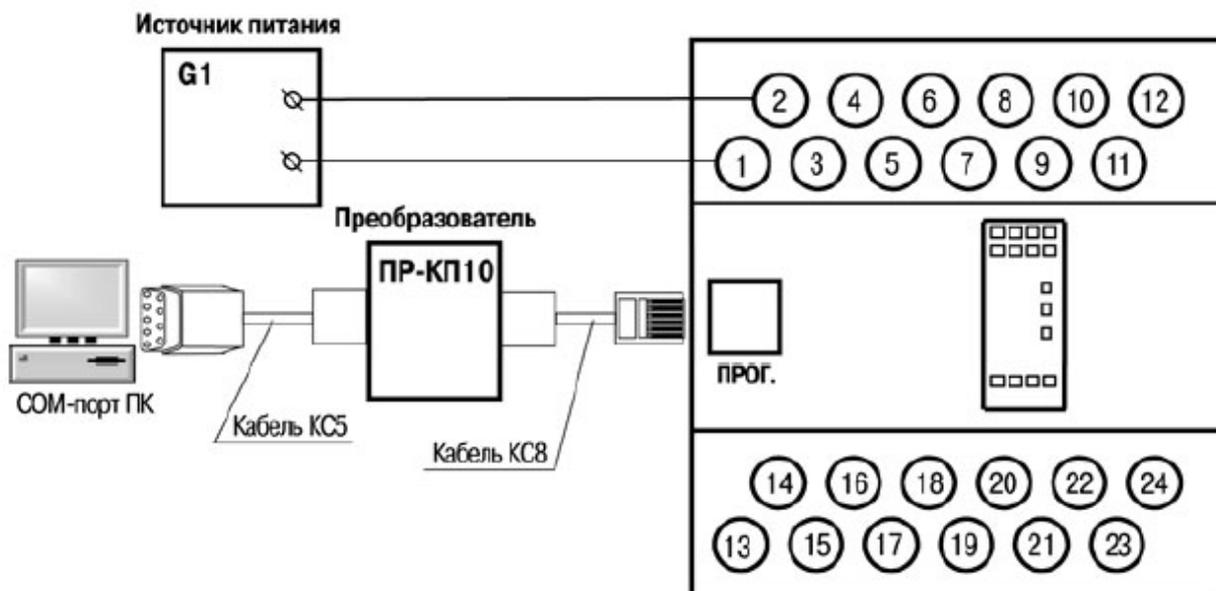


Рисунок 1.2 - Схема подключения напряжения

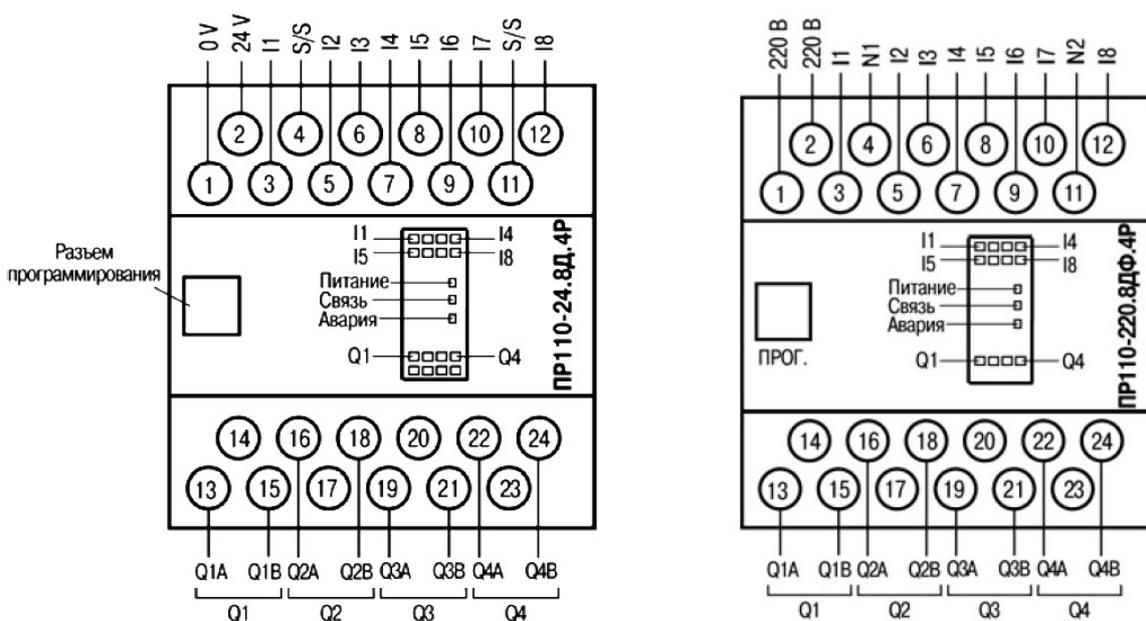


Рисунок 1.3 - Расположение контактов и элементов индикации в приборах ПР 110-х.8х.4х.

Подключения нагрузки к ОВЕН ПР осуществляется как на рисунок 1.4.

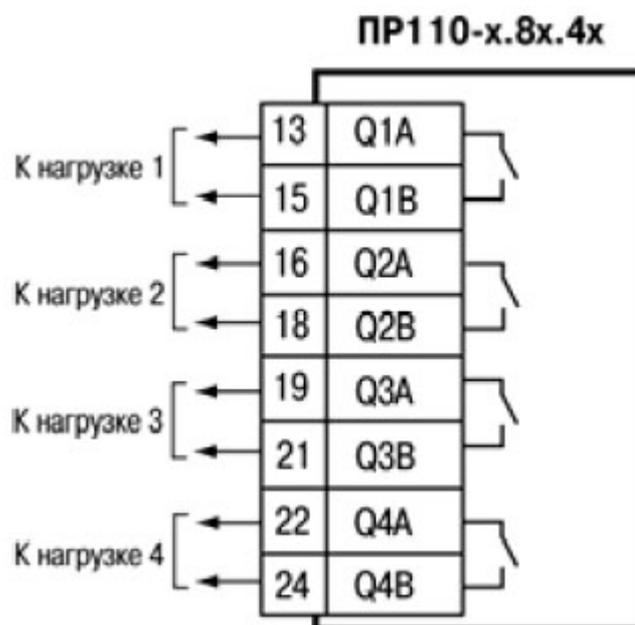


Рисунок 1.4 - Подключение нагрузок к выходным контактам реле

Под программированием мы подразумеваем ввод схемы. Программа OWEN Logic является на самом деле коммутационной схемой, представленной другим способом. Каждый вход обозначается буквой I и номером. Каждый выход обозначается буквой Q и номером.

Ссылка на программное обеспечение
OWEN Logic доступна по QR- коду:



2 Функции логических элементов программы

А.1 Битовые операции

При использовании функций И и ИЛИ следует учитывать, что не подключенные входы логических элементов к входам ПР или другим элементам в программе, будут иметь следующие состояния:

- для элемента И (AND) – логическая «1»;
- для элемента ИЛИ (OR) – логический «0».

В этом случае логические элементы выполняют функцию повторителя сигнала. Для увеличения числа входов у логических элементов используется их каскадное включение, например, как это показано на рисунке А.1.

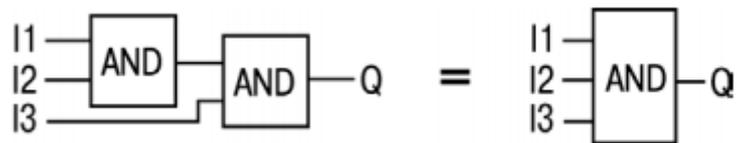


Рисунок А.1 – Пример каскадного включения логических элементов И

Функция «НЕ» (NOT)



I1	Q
0	1
1	0

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

Если на входы функции заведены целочисленные значения, то операция будет произведена над каждым битом в отдельности.

Функция «И» (AND)

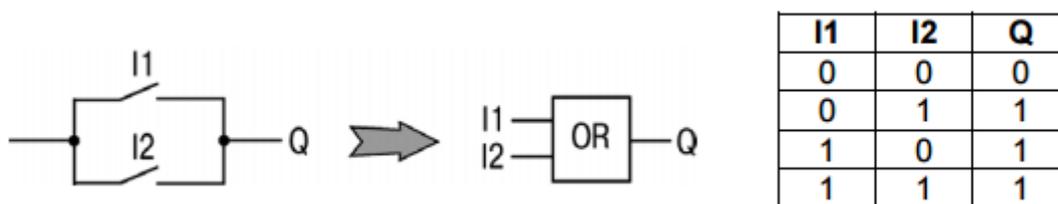


I1	I2	Q
0	0	0
0	1	0
1	0	0
1	1	1

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. Если на входы функции заведены целочисленные значения, то операция будет произведена над каждым битом в отдельности.

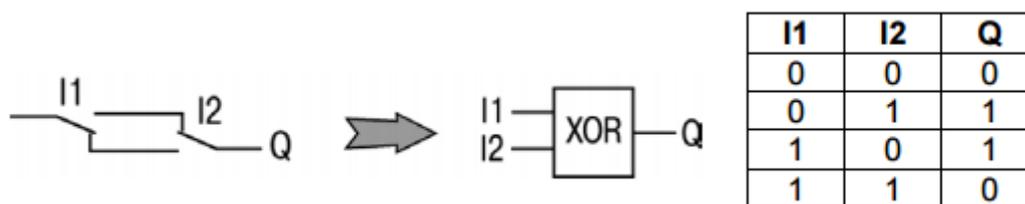
Функция «ИЛИ» (OR)



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. Если на входы функции заведены целочисленные значения, то операция будет произведена над каждым битом в отдельности.

Функция «исключающее ИЛИ» (XOR)

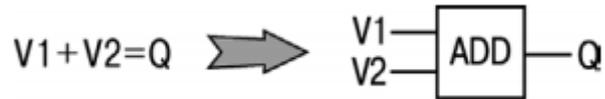


Элемент используется для выполнения логических операций. На выходе элемента логическая «1» (выход включен), только если на любом из входов логическая «1».

Работе соответствует приведенная таблица состояний. Если на входы функции заведены целочисленные значения, то операция будет произведена над каждым битом в отдельности.

А.2 Арифметические операции

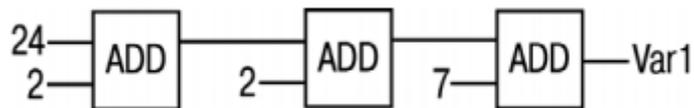
Сложение (ADD)



Результатом выполнения операции на выходе является сумма входных значений.

Если при выполнении операции значение числа получается больше 4294967295 (32 бита), выходящие за разрядность 32 бита биты отсекаются.

Пример применения элементов для сложения чисел $24 + 2 + 4 + 7 = \text{Var1}$:

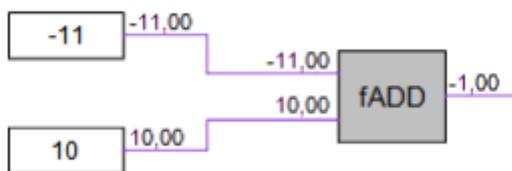


Сложение чисел с плавающей точкой (fADD)



Результатом выполнения операции на выходе является сумма входных значений.

Демонстрация работы элемента в режиме симулятора:



Вычитание (SUB)

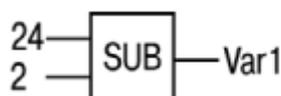


Результатом выполнения операции на выходе является разность входных значений.

Если при выполнении операции на первом входе (V1) значение числа меньше второго (V2), результатом будет число, полученное сложением младшего числа плюс 0x100000000 (4294967296) минус значение числа большего:

$$(V1 + 0x100000000) - V2 = Q.$$

Пример применения элемента для вычитания чисел $24 - 2 = \text{Var1} = 22$:



Пример применения элемента для вычитания чисел, где V1 меньше V2:

$$2 - 24 = \text{Var1} = 4294967274.$$

Вычитание чисел с плавающей точкой (fSUB)



Результатом выполнения операции на выходе является разность входных значений.

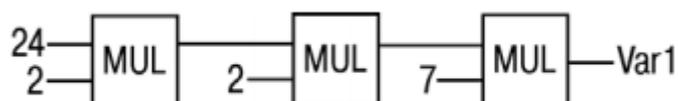
Умножение (MUL)



Результатом выполнения операции на выходе является перемножение входных значений.

Если при выполнении операции значение числа получается больше 4294967295 (32 бита), выходящие за разрядность 32 бита биты отсекаются.

Пример применения элементов для перемножения чисел $24 \cdot 2 \cdot 4 \cdot 7 = \text{Var1}$:

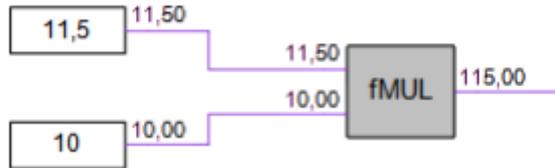


Умножение чисел с плавающей точкой (fMUL)



Результатом выполнения операции на выходе является перемножение входных значений.

Демонстрация работы элемента в режиме симулятора:



Деление (DIV)



Результатом выполнения операции на выходе является деление входных значений.

Если в результате деления получаются доли целого числа, то на выходе производится округление значения до целого числа в меньшую сторону.

При делении на 0 на выходе элемента будет 0xFFFFFFFF.

Пример применения элемента для деления чисел $24/2 = \text{Var1}$:

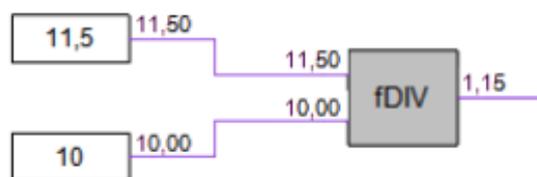


Деление чисел с плавающей точкой (fDIV)



Результатом выполнения операции на выходе является деление входных значений.

Демонстрация работы элемента в режиме симулятора:

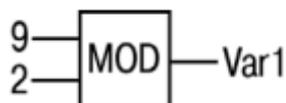


Деление с остатком (MOD)



Результатом выполнения операции на выходе является остаток от деления входных значений.

Пример применения элемента для выделения целого остатка от деления чисел 22 на 3 (Var1 = 1):

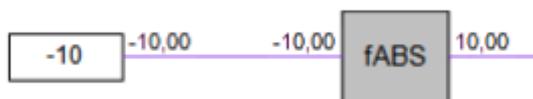


Абсолютное значение числа с плавающей точкой (fABS)



Результатом выполнения операции на выходе является деление входных значений.

Демонстрация работы элемента в режиме симулятора:



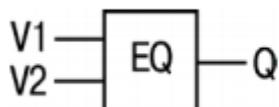
Возведение в степень числа с плавающей точкой (fPOW)



Результатом выполнения операции на выходе является деление входных значений.

А.3 Операции сравнения и выбора

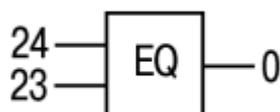
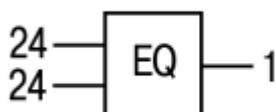
Равно (EQ)



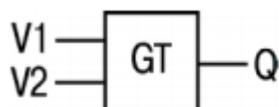
$V1 = V2; Q = 1$
 $V1 > V2; Q = 0$
 $V1 < V2; Q = 0$

Результатом выполнения операции на выходе является лог. «1», если входные числа равны.

Примеры применения элементов:



Больше (GT)



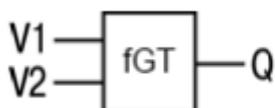
$V1 = V2; Q = 0$
 $V1 > V2; Q = 1$
 $V1 < V2; Q = 0$

Результатом выполнения операции на выходе является лог. «1», если входное число на верхнем входе (V1) больше числа на втором входе (V2).

Примеры применения элементов:

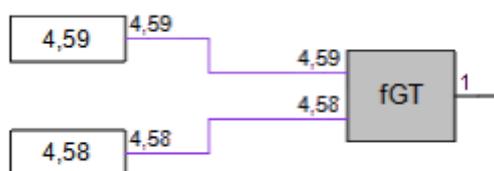


Больше, числа с плавающей точкой (fGT)

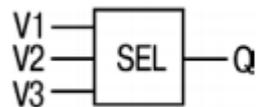


$V1 = V2; Q = 0$
 $V1 > V2; Q = 1$
 $V1 < V2; Q = 0$

Результатом выполнения операции на выходе является лог. «1», если входное число на верхнем входе (V1) больше числа на втором входе (V2).



Выбор (SEL)

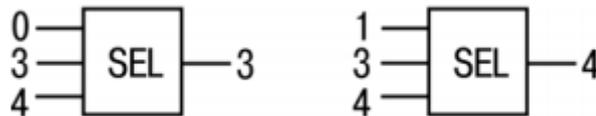


$V1 = 0; Q = V2$

$V1 = 1; Q = V3$

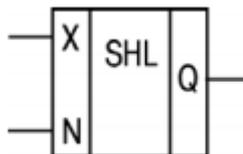
Результатом выполнения операции на выходе является входной сигнал V2, если V1 равен лог «0», и V3, если V1 равен лог «1».

Примеры применения элементов:



А.4 Операции логического битового сдвига и преобразования

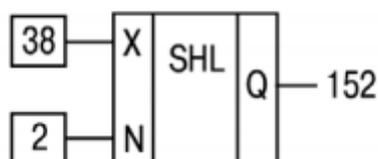
Побитный логический сдвиг влево (SHL)



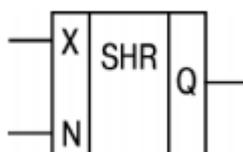
Элемент используется для выполнения операции побитного логического сдвига операнда X влево на N бит с дополнением нулями справа.

Пример применения для сдвига числа 38(десятичное) = 00100110(двоичное):

00100110 → 10011000

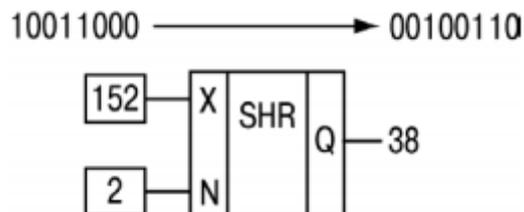


Побитный логический сдвиг вправо (SHR)

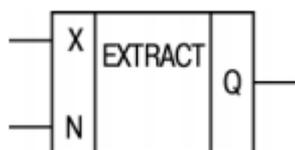


Элемент используется для выполнения операции побитного логического сдвига операнда X вправо на N бит с дополнением нулями слева.

Пример применения для сдвига числа 152(десятичное) = 10011000(двоичное):

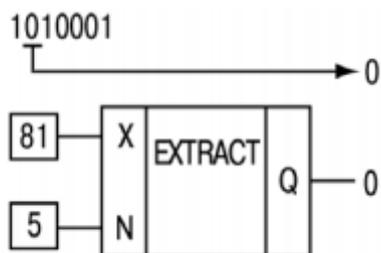


Чтение бита (EXTRACT)

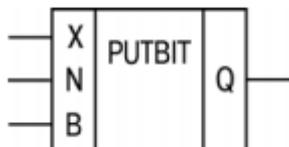


Элемент используется для выполнения операции чтения значения бита N в числе на входе X. Биты нумеруются с нуля. Число на входе X задается в десятичной системе. Выходное значение всегда двоичное (0 или 1).

Пример применения для чтения пятого бита из числа 81(десятичное) = 1010001(двоичное):



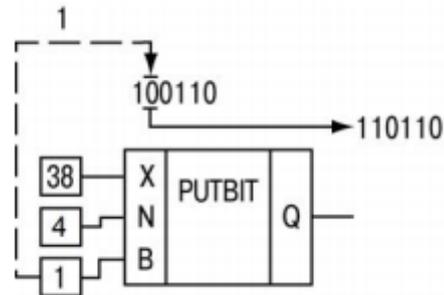
Запись бита (PUTBIT)



Элемент используется для выполнения операции записи в числе X значения бита N в состояние, указанное на входе B (лог. «0» или «1»). Число

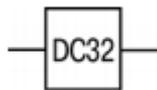
на входе X задается в десятичной системе. Выходное значение всегда двоичное (0 или 1).

Пример применения записи четвертого бита в лог. «1» для числа 38(десятичное) = 100110(двоичное):



Дешифратор (DC32)

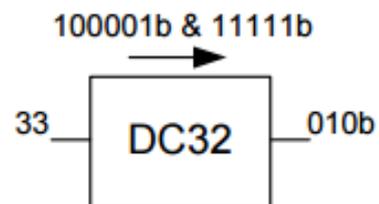
Двоичный код					Позиционный код							
5	4	3	2	1	32	31	6	5	4	3	2	1
0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	0	1	0	0
0	0	0	1	1	0	0	0	0	1	0	0	0
		1	0	0	0	0	0	1				
							
1	1	1	0	1	0	0	0	0	0	0	0	0
1	1	1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0



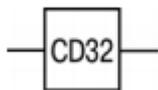
Элемент используется для выполнения операции преобразования двоичного кода на входе в позиционный код на выходе. Перед выполнением операции над значением на входе предварительно выполняется побитовая логическая операция «И» с операндом 0x1F (11111b).

Работе элемента соответствует приведенная таблица состояний.

Пример применения:



Шифратор (CD32)



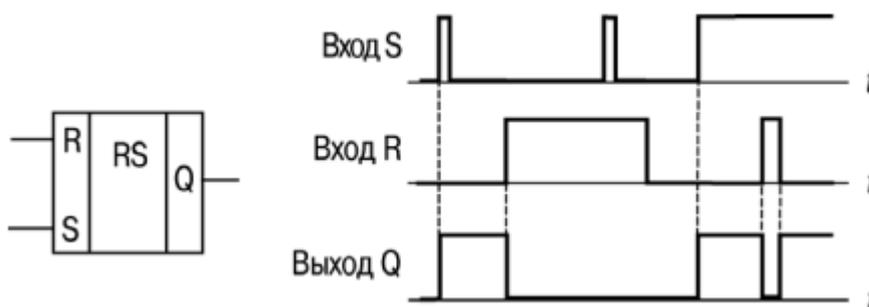
Элемент используется для выполнения операции преобразования позиционного кода на входе в двоичный код на выходе.

В случае, если входное двоичное значение имеет более одной логической «1» в разрядах, работа элемента ведется только со старшим единичным разрядом.

Работе элемента соответствует приведенная таблица состояний в пункте «Дешифратор».

3 Функциональные блоки программы

RS-триггер с приоритетом выключения (RS)

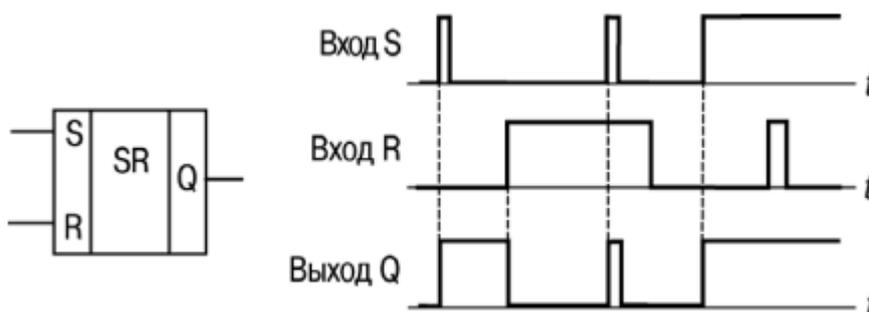


Блок используется для переключения с фиксацией состояния при поступлении коротких импульсов на соответствующий вход. Работу поясняет приведенная на рисунке диаграмма.

На выходе блока появится логическая «1» по фронту сигнала на входе S.

При одновременном поступлении сигналов на входы приоритетным является сигнал входа R.

SR-триггер с приоритетом включения (SR)

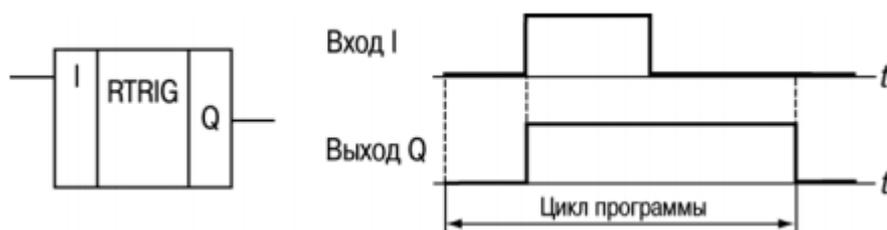


Блок используется для переключения с фиксацией состояния при поступлении коротких импульсов на соответствующий вход. Работу поясняет приведенная на рисунке диаграмма.

На выходе блока появится логическая «1» по фронту сигнала на входе S.

При одновременном поступлении сигналов на входы приоритетным является сигнал входа S.

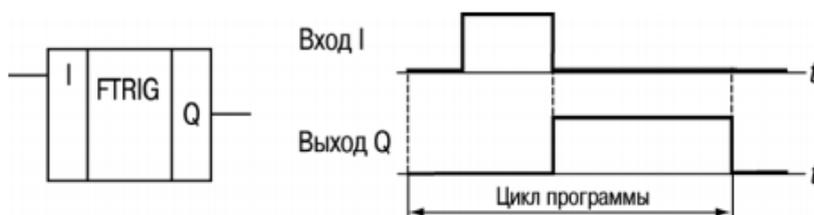
Детектор переднего фронта импульса (RTRIG)



Блок используется в случае необходимости иметь реакцию на изменение состояния дискретного входного сигнала. На выходе генерируется единичный импульс по переднему фронту входного сигнала.

Работу поясняет приведенная на рисунке диаграмма.

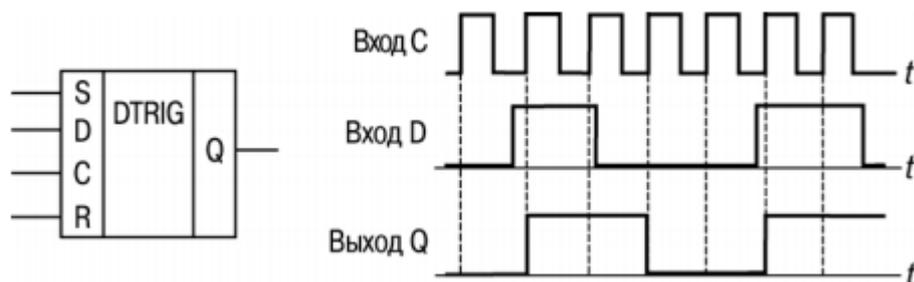
Детектор заднего фронта импульса (FTRIG)



Блок используется в случае необходимости иметь реакцию на изменение состояния дискретного входного сигнала. На выходе генерируется единичный импульс по заднему фронту входного сигнала.

Работу поясняет приведенная на рисунке диаграмма.

D-триггер (DTRIG)



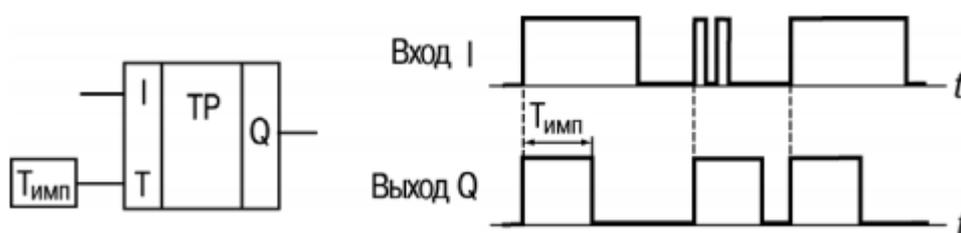
Блок используется для формирования импульса включения выхода на интервал времени импульса на входе D, при этом выходной интервал будет синхронизирован с тактовой частотой на входе C.

На выходе Q блока появится логическая «1» по фронту тактовых импульсов на входе С при наличии логической «1» на входе D. Возврат выхода в «0» произойдет по фронту тактовых импульсов на входе С при логическом «0» на входе D.

Работу поясняет приведенная на рисунке диаграмма.

Вход S обеспечивает принудительную установку выхода Q в состояние логической единицы; вход R является приоритетным и обеспечивает установку выхода Q в состояние логического «0».

Импульс включения заданной длительности (TR)

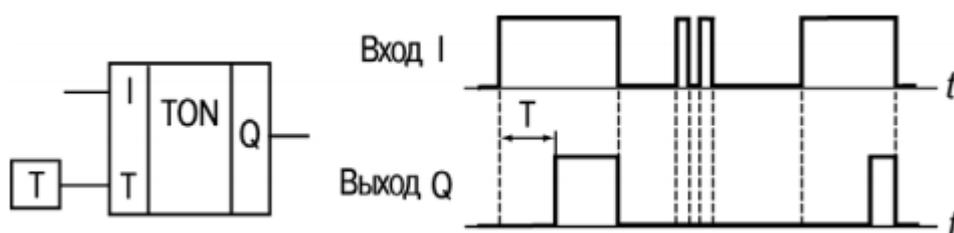


Блок используется для формирования импульса включения выхода на заданный интервал времени. Работу поясняет приведенная на рисунке диаграмма.

На выходе Q блока появится логическая «1» по фронту входного сигнала (I). После запуска выход Q не реагирует на изменение значения входного сигнала на интервале Т_{имп}, а по истечении этого интервала сбрасывается в «0».

Допустимый диапазон значений Т_{имп}=Т – от 0 до 4147200000 мс, или 48 дней.

Таймер с задержкой включения (TON)

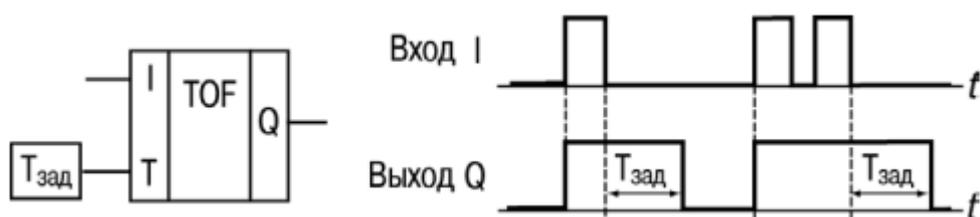


Блок используется для операции задержки передачи сигнала. Работу поясняет приведенная на рисунке диаграмма.

На выходе Q блока появится логическая «1» с задержкой относительно фронта входного сигнала (I). Выход включается логической «1» на входе продолжительностью не менее длительности T, а выключается по спаду входного сигнала.

Допустимый диапазон значений T – от 0 до 4147200000 мс, или 48 дней.

Таймер с задержкой отключения (TOF)

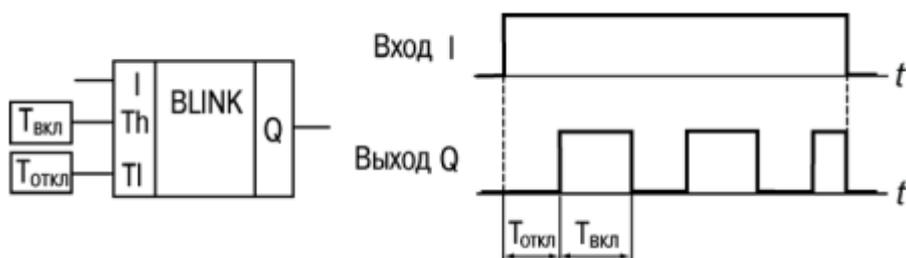


Блок используется для задержки отключения выхода. Работу поясняет приведенная на рисунке диаграмма.

На выходе блока появится логическая «1» по фронту сигнала на входе I, а начало отсчета времени задержки отключения (Tзад) происходит по каждому спаду входного сигнала. После отключения входного сигнала на выходе появится логический «0» с задержкой Tзад.

Допустимый диапазон значений Tзад – от 0 до 4147200000 мс, или 48 дней.

Генератор прямоугольных импульсов (BLINK)



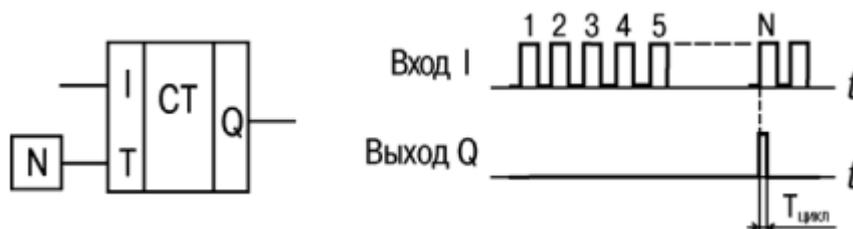
Блок используется для формирования прямоугольных импульсов пульсации.

На выходе Q генератора формируются импульсы с заданными параметрами длительности включенного (Tвкл – логическая «1») и

отключенного (Тоткл – логический «0») состояния на время действия управляющего сигнала на входе I (логической «1»). Работу поясняет приведенная на рисунке диаграмма.

Допустимый диапазон значений Твкл и Тоткл – от 0 до 4233600000 мс, или 49 дней.

Инкрементный счетчик с автосбросом (СТ)

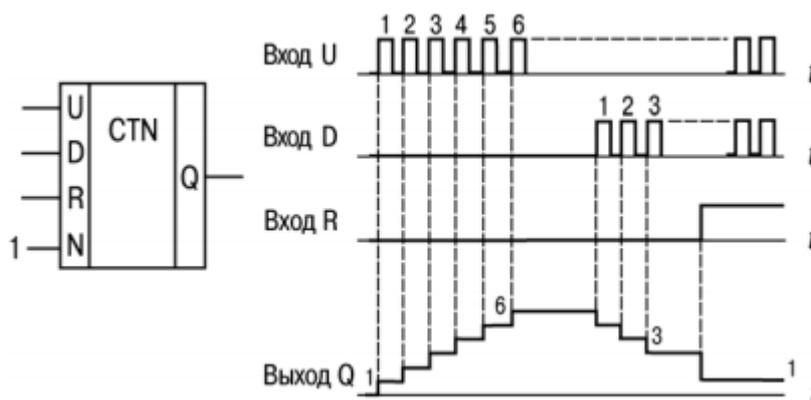


Блок используется для подсчета заданного числа импульсов N (вход T – уставка числа импульсов). Работу поясняет приведенная на рисунке диаграмма.

На выходе Q счетчика появится импульс логической «1» с длительностью рабочего цикла прибора ($T_{\text{цикл}}$), если число приходящих на вход I импульсов достигнет установленного значения N (на входе T).

Допустимый диапазон значений числа импульсов N – от 0 до 65535.

Универсальный счетчик (СТN)



Блок является счетчиком, который используется для прямого и обратного счета. Операция «прямой счет» выполняется по переднему фронту импульса на входе прямого счета (U), что увеличивает значение выходного

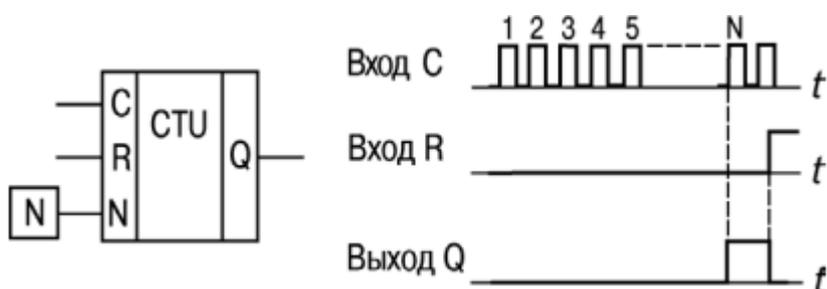
сигнала «Q». Импульсы, приходящие на вход D («обратный счет»), уменьшают значение выхода «Q».

При поступлении на вход R логической «1», выход счетчика «Q» устанавливается в значение входа N.

Работу блока поясняет приведенная на рисунке диаграмма. При одновременном поступлении сигналов на входы U и D приоритетным является сигнал входа U.

Допустимый диапазон значений числа импульсов N – от 0 до 65535.

Инкрементный счетчик (CTU)

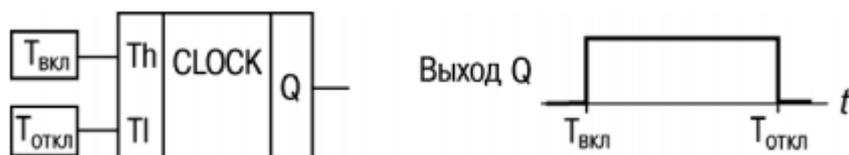


Блок используется для подсчета числа импульсов, приходящих на вход C. Работу, поясняет приведенная на рисунке диаграмма – на выходе Q счетчика появится импульс лог. «1», если число приходящих на вход импульсов достигнет установленного значения на входе N (N – уставка).

Допустимый диапазон значений числа импульсов N – от 0 до 65535.

Счетчик сбрасывается в 0 по переднему фронту импульса на входе R. При одновременном поступлении сигналов на входы приоритетным является сигнал входа R.

Интервальный таймер (CLOCK)



Блок используется для формирования импульса включения выхода по часам реального времени. Работу поясняет приведенная на рисунке диаграмма.

Время включения ($T_{вкл}$) и отключения ($T_{откл}$) выходов устанавливаются в качестве параметров блока.

Допустимый диапазон значений $T_{вкл}$ и $T_{откл}$ – от 0,00 с до 24 ч.

В случае если значение время отключения задано раньше времени включения (рисунок Б.1), диаграмма переключений будет иметь вид, приведенный на рисунке Б.2 (выход будет включен до момента времени отключения).

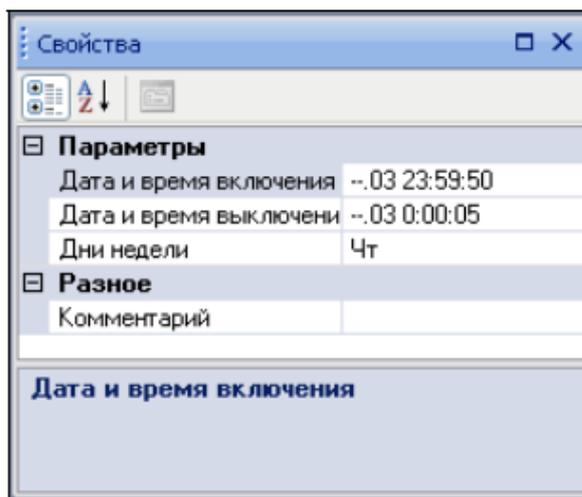


Рисунок Б.1 – Вкладка «Свойства» для настройки временных параметров работы блока

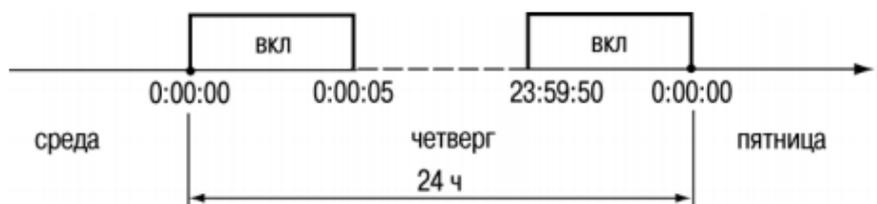
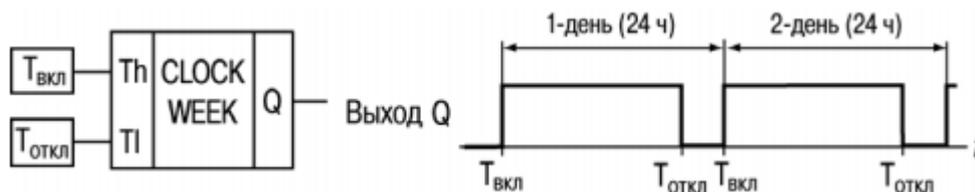
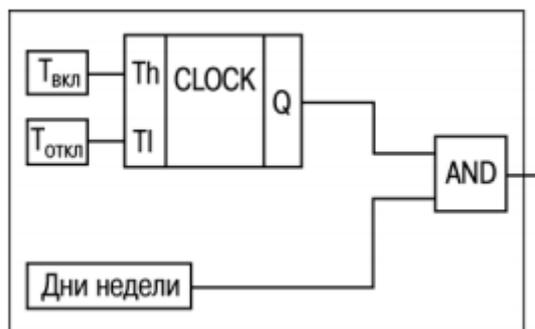


Рисунок Б.2 – Диаграмма включения выхода блока

Интервальный таймер с недельным циклом (CLOCK WEEK)



Блок используется для формирования импульса включения выхода по часам реального времени, с учетом дней недели. Работу поясняет приведенная на рисунке диаграмма. Внутренняя структура блока имеет вид:



Время включения ($T_{вкл}$), отключения ($T_{откл}$) выхода и дни недели работы устанавливаются в качестве параметров блока. Допустимый диапазон значений $T_{вкл}$ и $T_{откл}$ – от 0,00 с до 24 ч.

4 Последовательность работы над проектом

Составление коммутационной 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 программу на основе функций (логических элементов) и функциональных блоков.

5 Примеры выполнения задания

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

5) предусмотреть возможность включения светильника только в вечернее и ночное время. Выбор программируемого прибора.

Для реализации данной задачи управляющий автомат должен иметь встроенные часы реального времени, три входа и один выход, что обеспечивают приборы из серии ПР110, имеющие последнюю букву «Ч» в обозначении. При выполнении автоматики на основе ПР110-24.8Д4Р-Ч можно воспользоваться схемой, приведенной на рисунке 2.2 (номера входов и выходов для ПР выбраны произвольно).

6 Создание коммутационной программы

На этапе планирования составляется схема реализации поставленной задачи на основе логических элементов и функциональных блоков, доступных в программе (см. Приложения А и Б). Схема может иметь вид, приведенный на рисунке 6.1. Выход Q2 схемы используется как контрольный для проверки функционирования логической части (элементов D1–D6). Выход Q1 является основным и может включиться только в заданные таймером D7 интервалы времени, при соблюдении логических условий, обеспечиваемых элементами D1– D6.

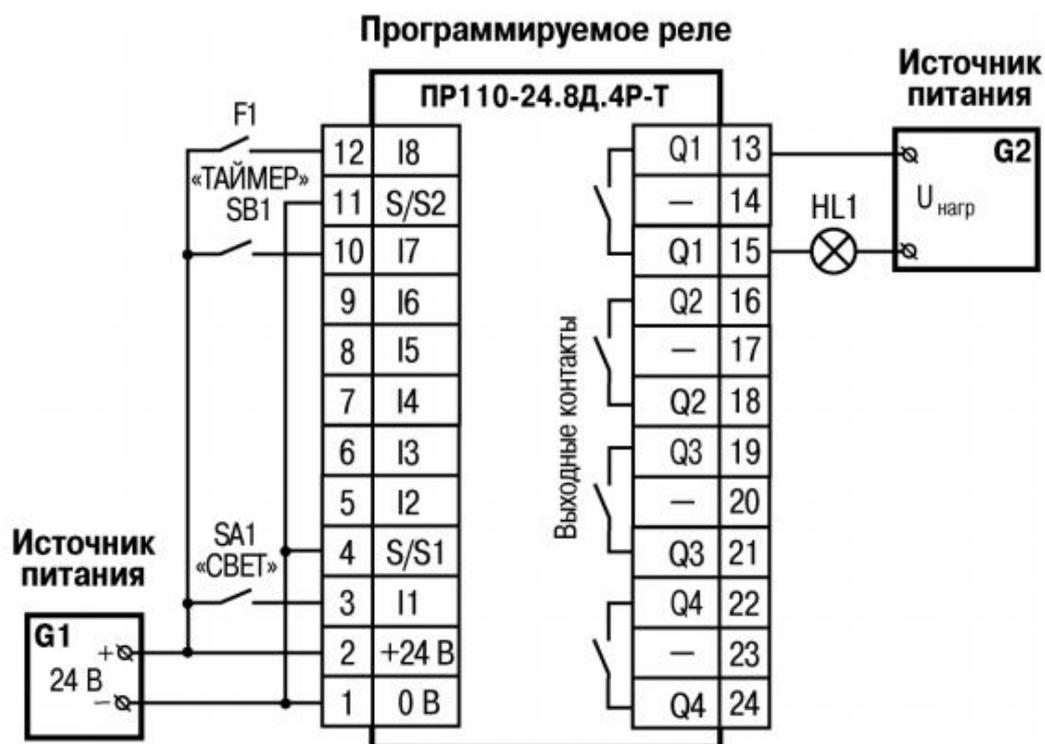


Рисунок 6.1 – Электрическая схема подключения ПР,
 SA1 – переключатель OFF – ON, SB1 – кнопка без фиксации OFF – ON;
 HL1 – световая лампа

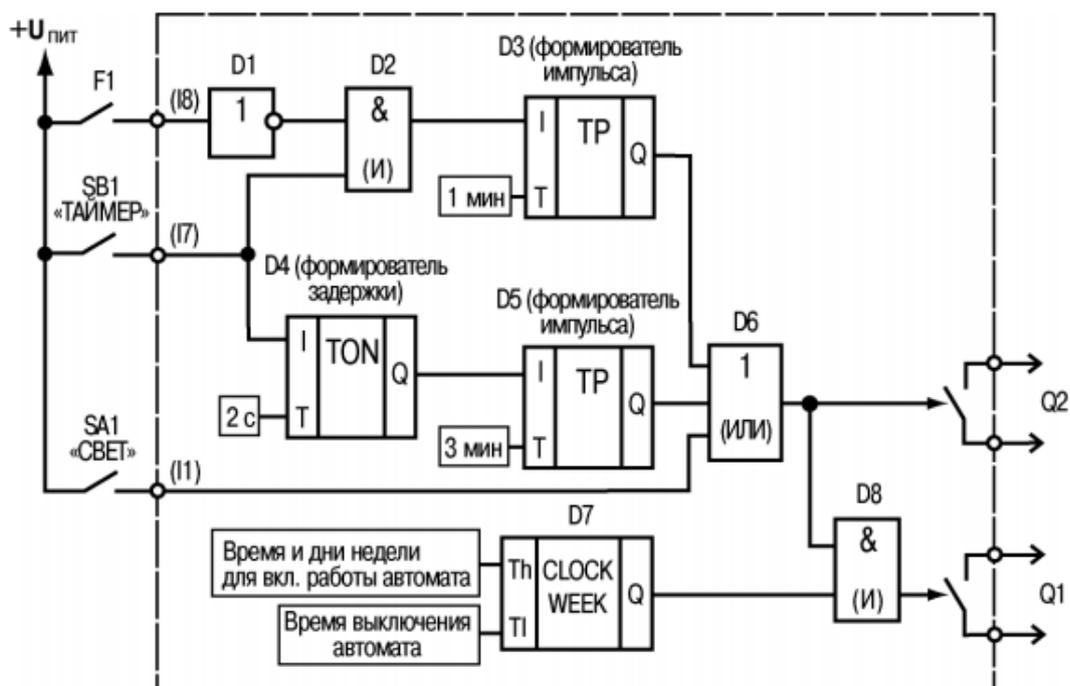


Рисунок 6.2 – Схема электрическая принципиальная для управляющей программы «Свет_Logic»

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 28 Среда программирования OWEN Logic формирователь импульса (D3) длительностью 1 мин. Этот импульс через элемент ИЛИ (D6) поступит на выход Q2;

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 Q2;

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 Q2 (интервал действия импульса не зависит от состояния датчика F1);

4) при включении тумблера «СВЕТ» (SA1) лог. «1» через элемент ИЛИ (D6) поступит на выход Q2 (на время включения SA1);

5) сигнал лог. «1» с выхода Q2 поступает на вход логического элемента И (D8), а если на второй вход этого элемента также поступает лог. «1» (от таймера реального времени D7) – на выходе Q1 появится лог. «1» (выходное реле Q1 включится).

Программа, обеспечивающая работу ПР, будет иметь вид, приведенный на рисунке 6.3.

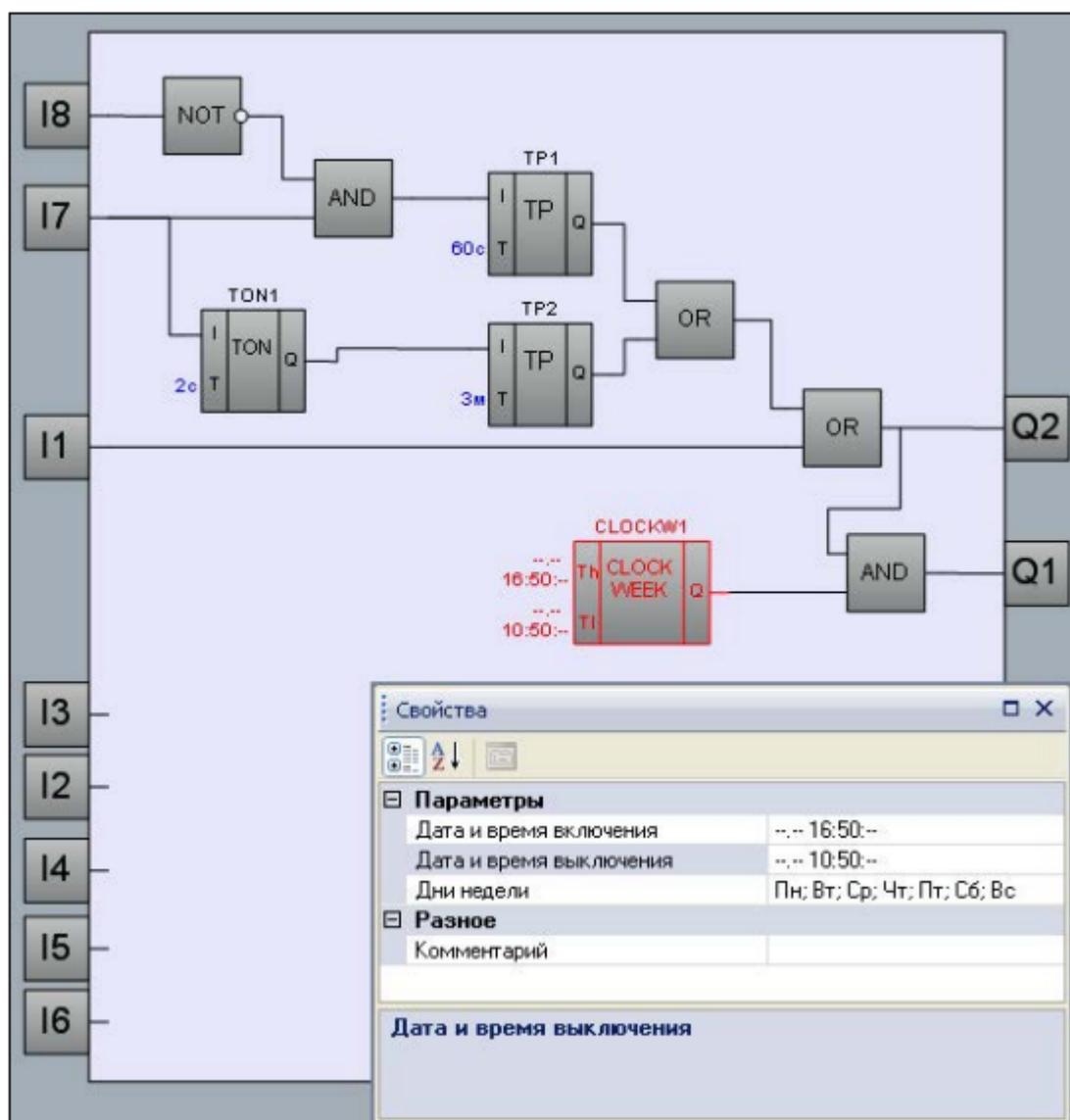


Рисунок 6.3 – Управляющая программа «Свет_Logic» на основе функциональных блоков (ФБ) программы OWEN Logic (с открытой вкладкой свойств для блока CLOCKW1)

Автоматическое управление электродвигателем мешалки

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

Задача 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 электромотора 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 электромотора от кнопок ПУСК и СТОП (без временных интервалов отключения);

4) при перегрузке электромотора (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. Выбор программируемого прибора.

Для реализации данной задачи управляющий прибор должен иметь шесть входов (по числу управляющих сигналов) и три выхода, что обеспечивает любая модель прибора из серии ПР110. При выполнении автоматики на основе ПР110-24.8Д4Р можно воспользоваться схемой, приведенной на рисунке 6.4.

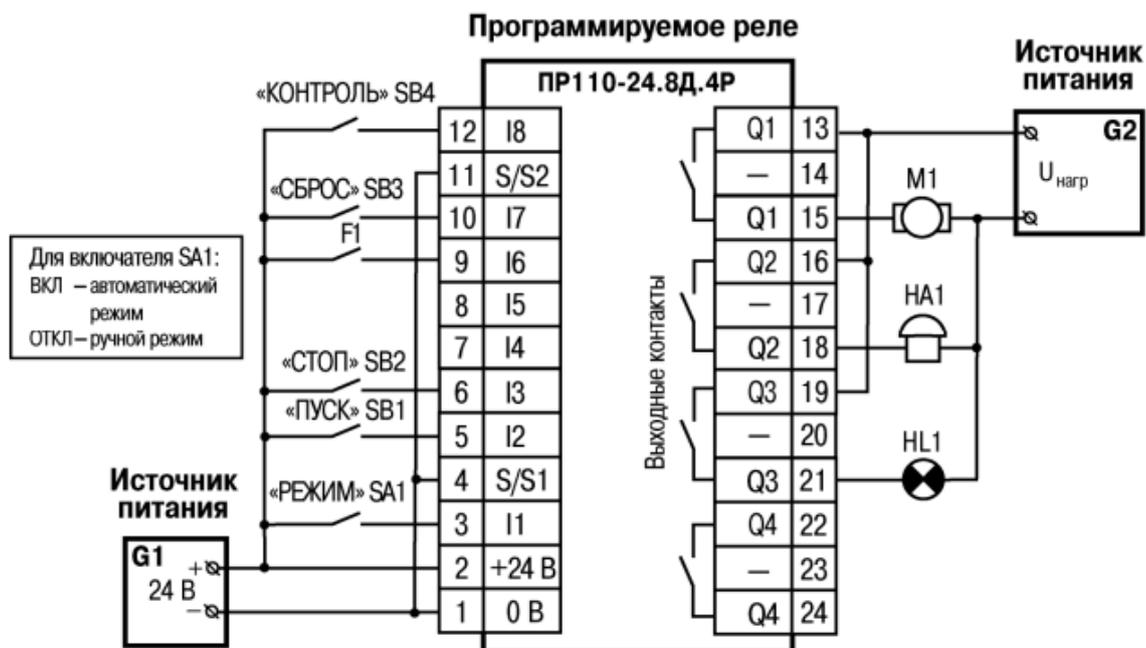


Рисунок 6.4 – Электрическая схема подключения ПР,

SA1 – переключатель OFF - ON;

SB1-SB4 – кнопки без фиксации OFF – ON;

M1 – электромотор; HA1 – звонок;

HL1 – индикаторная лампа

Работу программы прибора можно представить в виде цепей схемы, приведенной на рисунке 2.6 (выходы Q1–Q3 соответствуют двум контактам клеммника ПР).

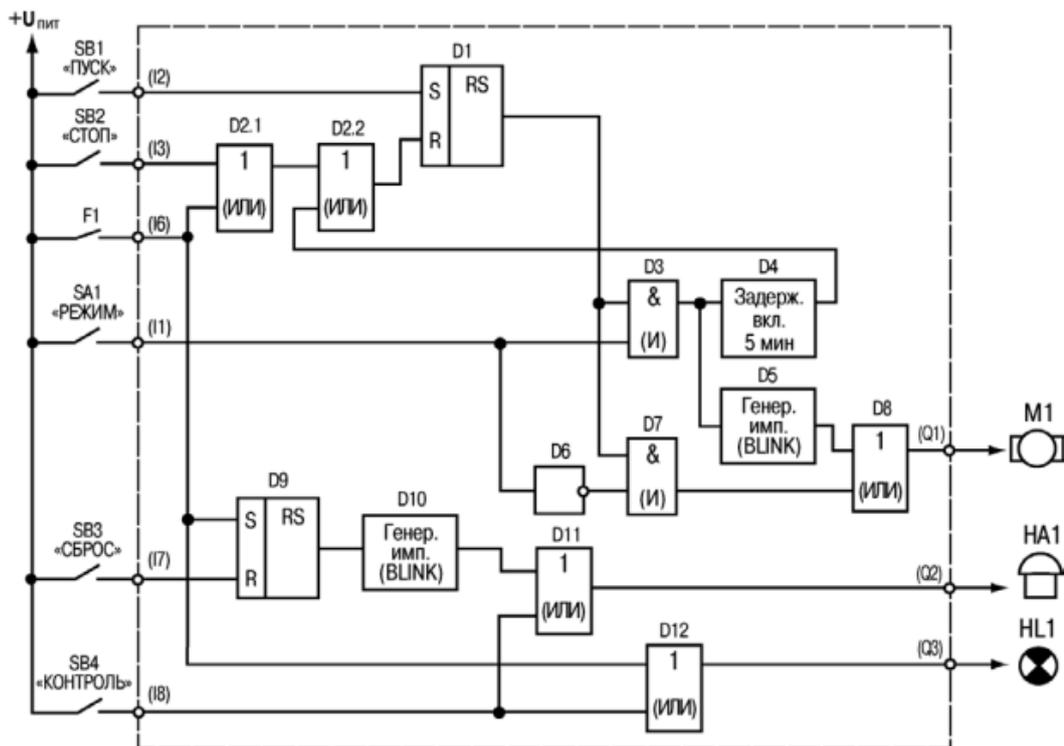


Рисунок 6.5 – Схема электрическая принципиальная для управляющей программы «Мешалка_Logic»

Описание работы схемы:

Цепь входа I2 (включение установки): при нажатии кнопки ПУСК (SB1) на входе S (D1) появляется лог. «1» – на выходе RS-триггера D1 установится лог. «1». Этот сигнал поступает дальше, в зависимости от состояния выключателя SA1:

- при разомкнутых контактах SA1 (РЕЖИМ – Ручной), лог. «1» проходит через элементы D8, D9 и поступает на выход Q1 (контактами выходного реле электромотор M1 включится);

- при замкнутых контактах SA1 (РЕЖИМ – Автоматический), лог. «1» проходит только через элемент D3 для запуска работы блоков D4, D5.

Цепь входа I3 (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» (включение выхода Q1 блокируется).

Цепь входа I1 (формирование интервалов работы электромотора): при включенном триггере D1 и замкнутых контактах включателя SA1 (РЕЖИМ – Автоматический):

– сигнал лог. «1» от SA1 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.

Сигнал лог. «1» с выхода D3 поступает на D5 – генератор импульсов с параметрами: лог. «1» – 15 с, лог. «0» – 10 с. Эти импульсы с выхода генератора проходят через элемент ИЛИ (D8) и поступают на выход Q1 для управления работой электромотора M1.

При разомкнутых контактах включателя SA1 (РЕЖИМ – Ручной) логический элемент D3 заблокирован и сигнал с выхода триггера D1 через элемент D6 сразу поступает на выход Q1, т. е. в этом режиме состояние выхода триггера непосредственно управляет работой электромотора M1.

Цепь входа I6 (включение звуковой сигнализации): при срабатывании датчика F1 на входе S (D9) появляется лог. «1» – на выходе RS-триггера D9 установится лог. «1», которая включает генератор 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 входа I7 (отключение звуковой сигнализации): при нажатии кнопки СБРОС (SB3) на входе R (D9) появляется лог. «1» – на выходе RS-триггера установится лог. «0» – генератор D10 отключится.

Цепь входа I8 (контроль работы лампы и звуковой 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. 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 6.6.

Примечания

1. 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.

2. Технологический цикл работы установки можно сделать полностью автоматическим, если схему дополнить инкрементным счетчиком (СТ), выходным сигналом которого можно выключать триггер D1.

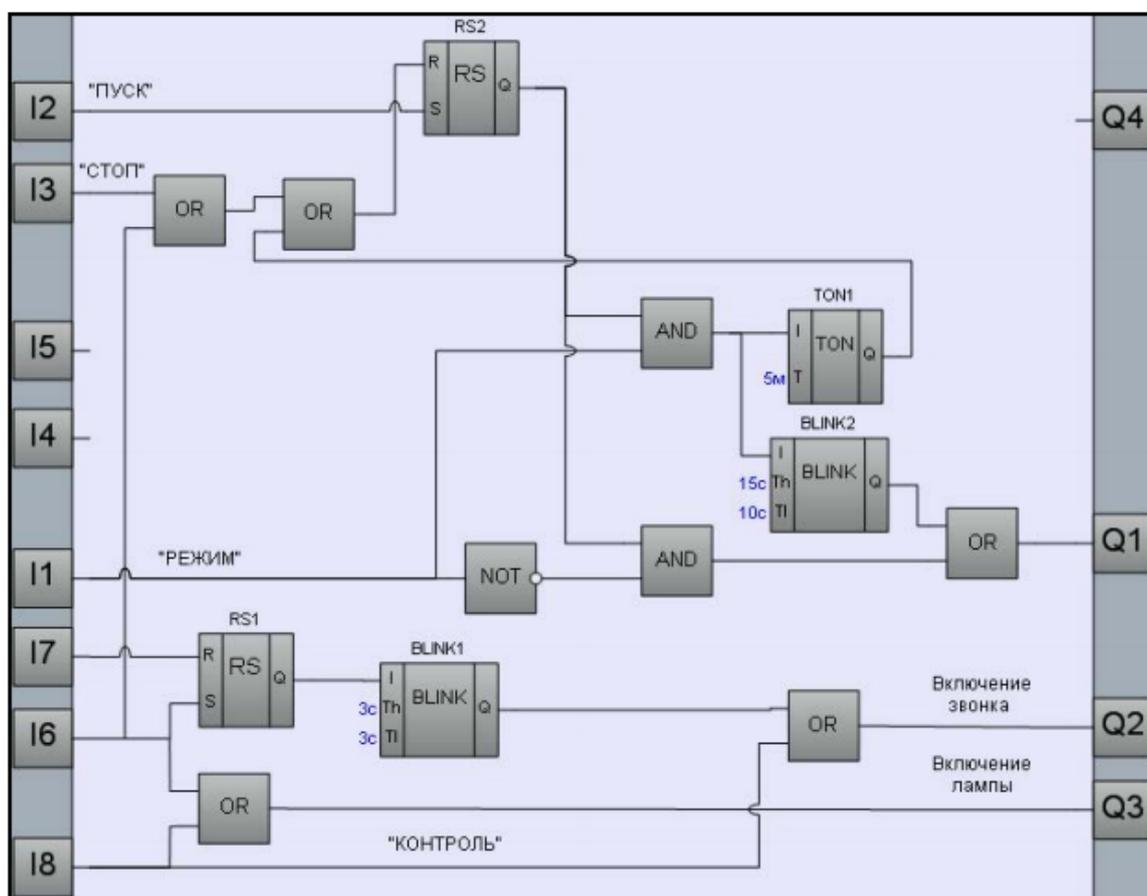


Рисунок 6.6 – Управляющая программа «Мешалка_Logic» на основе ФБ среды программирования OWEN Logic

7 Варианты заданий

1. Управление освещением

Алгоритм 1.

- При нажатии кнопки SB1 с трехсекундной задержкой загорается HL1.
- При нажатии кнопки SB2 или SB3 HL1 отключается.
- При нажатии кнопки SB4 через 5 секунд включается HL2, ещё через 5 секунд включается HL3, а HL4 начинает моргать с частотой 1 Гц.

Алгоритм 2.

- При нажатии кнопки SB1 включается HL1.
- При нажатии кнопки SB2 включается HL2.
- При нажатии кнопки SB3 включается HL3.
- При нажатии кнопки SB4 включается HL4.
- При чем светильники должны включаться последовательно (HL 2 не будет работать, если не работает HL1, HL3 не будет работать если не работает HL2, HL4 не будет работать если не работает HL3).

2. Создать алгоритм управления насосной станции, состоящей из 3-х насосов для откачки воды из резервуара с помощью программируемого реле.

Система управления должна обеспечивать:

- Поочередную работу двигателей (сначала №1, затем №2, а №1 отключился и так далее)
- При срабатывании датчика "Верхний уровень" насосы должны работать по парно (№1 + №2, №2 + №3, №3 + №1)
- При нажатии аварийный отсос необходимо обеспечить включение всех насосов.
- При нажатии кнопки «Стоп» прекращается работа всех насосов и системы управления (контроллер не должен реагировать на сработку датчиков).
- Датчик нижний останавливал работу всех насосов до наполнения резервуара вновь.

Алгоритм работы:

- Кнопка «Стоп» должна останавливать всю работу.
- При срабатывании датчика «Нижний уровень» (у нас концевой выключатель), должен останавливать работу работающего в настоящий момент насоса.
- Кнопка «Пуск» должен запускать работу в штатном режиме
- При срабатывании датчика «Верхнего уровня» (у нас концевой выключатель), должен переключить работу в режим «Турбо»

Описание режимов работы:

Режим «Штатный»:

- Начало работы начинается с нажатия на кнопку «Пуск»
- Насосы должны поочередно меняться. А именно: сначала насос №1, затем насос №2, после насос №3, после насос №1 и так далее.
- Насосы должны работать по 30 секунд каждый (имитация 12 часов работы)
- Остановка работы насоса, работающего в текущий момент должна производиться:

- а) Когда срабатывает датчик «Нижнего уровня».
- б) Когда срабатывает кнопка «Стоп».

Режим «Турбо»:

- Режим активируется при работающей в штатном режиме системе управления срабатыванием датчика «Верхнего уровня» в течении 10 секунд, затем происходит смена пары насосов.
- Насосы должны работать попарно и меняться: насосы №1 и №2, потом насосы №2 и №3, потом насосы №3 и №1, потом насосы №1 и №2 и так далее.
- Насосы должны работать по 10 секунд каждая пара. То есть 10 секунд работает пара насосы №1 и №2, затем 10 секунд работают насосы №2 и №3,

– Сигнальная лампа должна гореть постоянно, сигнализируя о режиме работы «Турбо».

– Остановка работы должна остановиться после нажатия на кнопку «Стоп» или срабатыванием «Нижнего» датчика уровня.

– Переключение системы из режима «Турбо» в режим «Штатный» должно осуществляться при обратном срабатывании датчика «Верхний уровень».

Режим «Аварийная откачка воды»

– Начало работы должно начинаться с нажатия на кнопку «Аварийная»

– Сразу все насосы начинают свою работу независимо от своего состояния

– Сигнальная лампа должна мерцать на частоте 1 Герц

– Остановка работы должна произойти после нажатия на кнопку «Стоп», или при срабатывании датчика «Нижнего уровня».

3. Двигатель включается с помощью 2-х кнопок, причем одна из них является главной (нажимается первой), а вторая должна быть нажата не позднее, чем через 3 сек. после нажатия первой. При неправильных действиях схема блокируется, пока не будет нажата третья кнопка – Сброс.

4. Реализовать устройство голосования мажоритарного типа для 4-х участников. Один из участников – главный, его голос считается за 2 голоса обычного участника.

5. Реализовать схему пожарной сигнализации объекта, контролируемого 4-я датчиками. При срабатывании любых 2-х датчиков через 5 сек. срабатывает сигнализация. Сигнал в виде прерывистого зуммера поступает диспетчеру, при этом автоматически включается подача воды на объект. Диспетчер имеет возможность выключить зуммер.

6. Реализовать схему охранной сигнализации объекта, контролируемого 4-я датчиками. При срабатывании любого датчика срабатывает сигнализация. Сигнал в виде прерывистого зуммера поступает диспетчеру, при этом

автоматически блокируются двери. Диспетчер имеет возможность выключить зуммер нажатием кнопки.

7. Реализовать схему управления промышленными воротами. Открытие и закрытие ворот осуществляется нажатием соответствующих кнопок. При движении ворот мигает лампочка Опасность. Частота мигания лампочки различная при открывании и закрывании ворот. Если в течение 2-х минут ворота не открылись (закрылись), то включается лампочка Авария.

8. Реализовать схему управления промышленными воротами. Открытие и закрытие ворот осуществляется нажатием соответствующих кнопок. При движении ворот 5 сек. мигает лампочка Опасность. Частота мигания лампочки различная при открывании и закрывании ворот. Если в течение 2-х минут ворота не открылись (закрылись), то включается лампочка Авария.

9. Реализовать схему по экономии электроэнергии. Инфракрасный датчик срабатывает при наличии людей в помещении, при этом размыкается нормально замкнутый контакт датчика и свет в помещении включается. Существует возможность отключения схемы кнопкой, при этом, если свет не горел, то он не включится, если свет горел, то он не выключится. Свет при этом не горит в дневное время.

10. Реализовать схему аварийного отключения электропечи сауны после 8 часов общего времени работы. Включение и выключение печи осуществляется двумя кнопками. В конце суток печь выключается автоматически.

11. Реализовать схему подачи звонков в учебном заведении согласно расписания занятий для 6 уроков. Звонок с урока и на урок – 3 сек.

8 Порядок выполнения работы.

1. Ознакомиться с работой логического модуля.
2. По содержательному описанию объекта управления полученного варианта задания составить схему логики функционирования системы управления.
3. Реализовать схему управления используя программный пакет OWEN Logic.
4. Проверить правильность функционирования схемы управления.
5. Показать функционирование схемы преподавателю.

Содержание отчета.

1. Цель работы и содержательное описание функционирования объекта управления.
2. Логическая схема функционирования объекта управления.
3. Монтажная схема системы управления.

Литература

1. Техническая документация фирмы OWEN. Логический модуль OWEN Logic.

Информационно-методическое обеспечение

2. Программный пакет OWEN Logic.