

## **21. Назначение и область применения реле ONI**

Программируемые логические реле модульного исполнения ONI PLR-S (логические реле) предназначены для построения базовых систем автоматизированного управления малой и средней степеней сложности.

Логические реле могут быть применены для автоматизации различного технологического и инженерного оборудования, построение систем автоматизированного сбора и обработки информации, построение систем учета и распределения энергоресурсов, систем дистанционного управления и т. д.

# Общие технические характеристики

Таблица 1

Параметр		Значение
Напряжение питания, В	DC	от 10,8 до 28,8
	AC	от 85 до 265
Потребляемая мощность, Вт, не более	DC	4 на один модуль
	AC	10 на один модуль
Диапазон рабочих температур, °C		от минус 20 до плюс 55
Относительная влажность воздуха, %		5–95, без образования конденсации
Степень загрязнения микросреды по ГОСТ Р МЭК 60664		2, без содержания агрессивных и взрывоопасных паров и газов в концентрациях, вызывающих коррозию металлов и разрушение изоляции
Способ охлаждения		естественное охлаждение окружающим воздухом
Степень защиты по ГОСТ 14254 (IEC 60529)		IP20
Срок службы, лет		7
Ремонтопригодность		неремонтопригодны
Масса, кг		не более 0,45 на один модуль

Таблица 2 – Модули ЦПУ

Типоисполнение	Конфигурация											
	Входы			Выходы		Интерфейсы			Периферия			
	Цифровые	Аналоговые	Универсальные	Цифровые	Аналоговые	RS232	RS485	Ethernet**	Экран	Клавиатура	Расширение	Питание
PLR-S. CPU0804	4	-	4	4R	-	1	-	-	-	-	-	DC
PLR-S. CPU0804(T) 24В DC	4	-	4	4Т	-	1	-	-	+	+	-	DC
PLR-S. CPU0804(R) 220В AC	8	-	-	4R	-	1	-	-	-	-	-	AC
PLR-S. CPU1004(R) 24В DC	4	-	6	4R	-	1	1	-	+	+	+	DC
PLR-S. CPU1004(R) 220В AC	10	-	-	4R	-	1	1	-	+	+	+	AC
PLR-S. CPU1206	6	-	6	6R	-	1	-	-	+	+	+	DC
PLR-S. CPU1206(T) 24В DC	6	-	6	4R/2Т	-	1	-	-	+	+	+	DC
PLR-S. CPU1206(R) 220В AC	12	-	-	6R	-	1	-	-	+	+	+	AC
PLR-S. CPU1410	8	-	6	10R	-	1	1	-	+	+	+	DC
PLR-S. CPU1410(T) 24В DC	8	2	4	6R/2Т	1	1	1	-	+	+	+	DC
PLR-S. CPU1410(R) 220В AC	14	-	-	10R	-	1	1	-	+	+	+	AC

\*Примечание:

R – выход релейный;

T – выход транзисторный (открытый коллектор).

\*\*Примечание: тип разъема – 8P8C.

Таблица 3 – Модули расширения

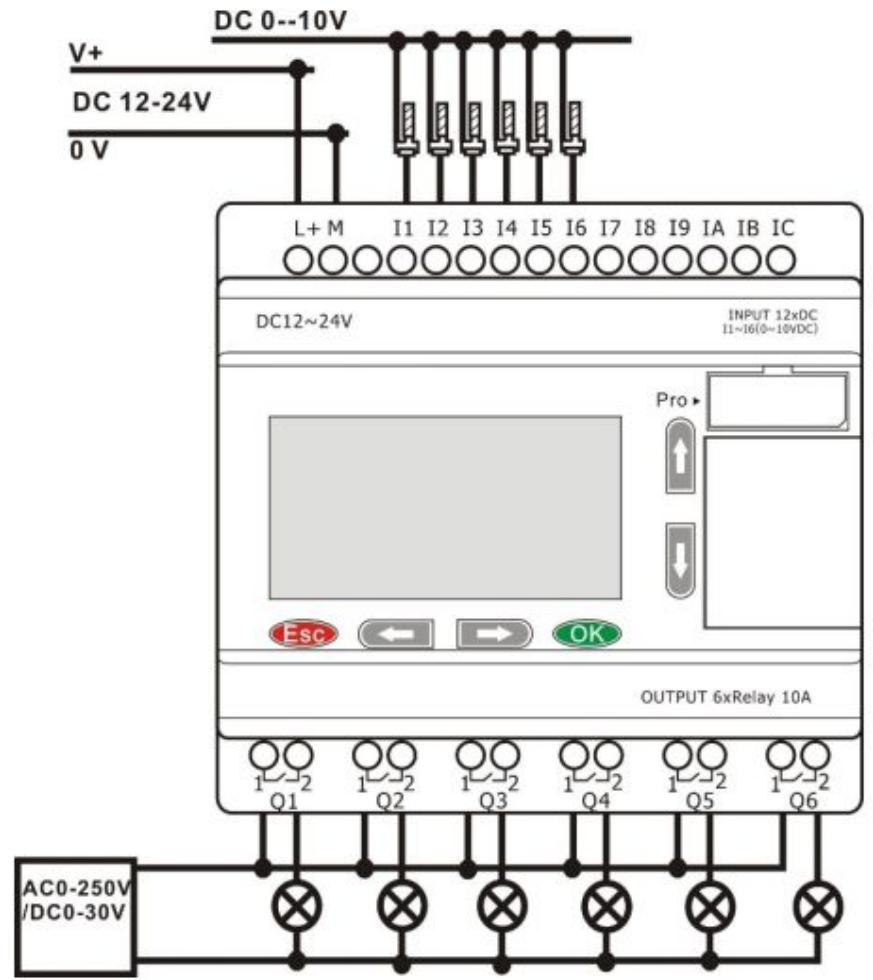
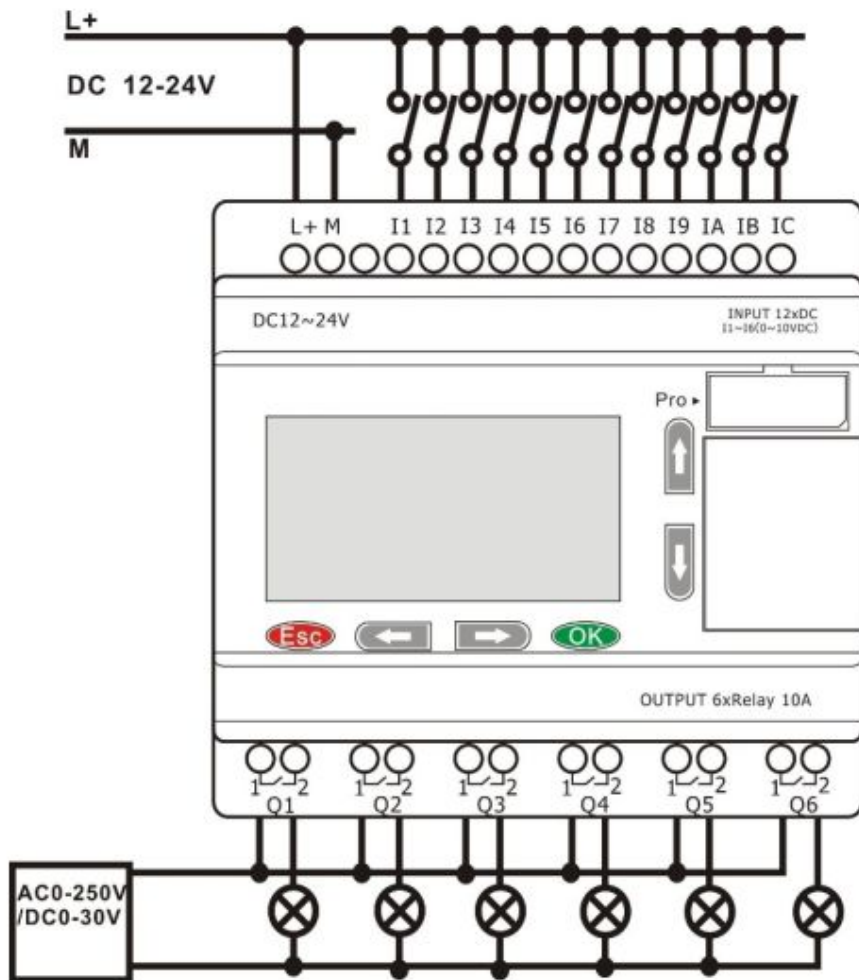
Типоисполнение	Конфигурация												
	Входы			Выходы			Интерфейсы			Периферия			
	Цифровые	Аналоговые	Универсальные	Цифровые	Аналоговые	Термо-сопротивления	RS232	RS485	Ethernet**	Экран	Клавиатура	Расширение	Тип питания
PLR-S. 8DI/8DO	4	-	4	8R	-	-	-	-	-	-	-	-	DC
PLR-S. 8DI/8DO (T) 24В DC	4	-	4	8T	-	-	-	-	-	-	-	-	DC
PLR-S. 8DI/8DO (R) 220В AC	8	-	-	8R	-	-	-	-	-	-	-	-	AC
PLR-S. 4AI	-	4	-	-	-	-	-	-	-	-	-	-	DC
PLR-S. 2AO	-	-	-	-	2	-	-	-	-	-	-	-	DC
PLR-S. 3 RTD	-	-	-	-	-	3	-	-	-	-	-	-	DC
PLR-S. RS485	-	-	-	-	-	-	-	1	-	-	-	-	DC

\*Примечание:

R – выход релейный;

T – выход транзисторный (открытый коллектор).

\*\*Примечание: тип разъема – 8P8C.



## 22. Программное обеспечение

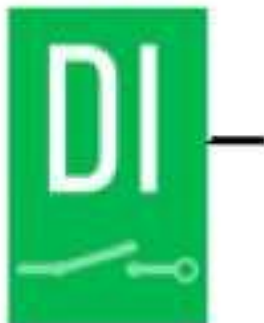
Программное обеспечение ONI PLR Studio предназначено для разработки и отладки прикладных программ для логических реле ONI серии PLR-S, с использованием графического языка диаграмм функциональных блоков FBD.



# Библиотека функциональных блоков

## Цифровой вход

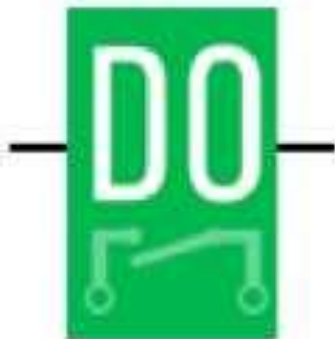
1001



Блок соответствует физическому цифровому входу модуля ЦПУ или модуля расширения.

## Цифровой выход

Q001



Блок соответствует физическому цифровому выходу модуля ЦПУ или модуля расширения.

#### 2.7.1.1.4 Постоянные логические уровни

При необходимости использования в программе постоянных логических уровней 0 или 1 их можно задать, подключив вход модуля к специальным блокам.

Low



Блок "всегда 0" постоянно формирует на выходе сигнал логического нуля.

High



Блок "всегда 1" постоянно формирует на выходе сигнал логической единицы.

## Аналоговый вход

AI001



Блок соответствует физическому аналоговому или универсальному входу модуля ЦПУ или модуля расширения.

## Аналоговый выход

AQ001



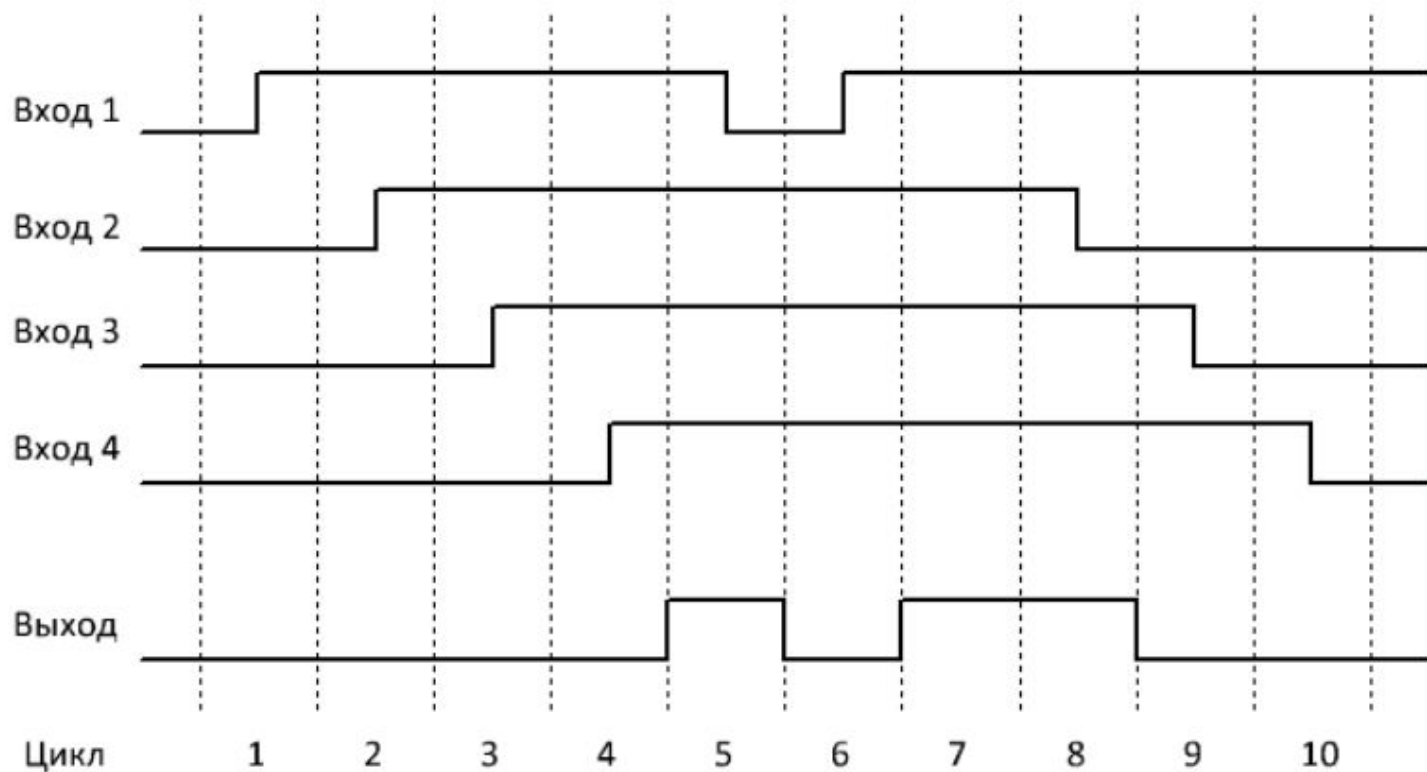
Блок соответствует физическому аналоговому выходу модуля ЦПУ или модуля расширения.

## 2.7.2 Логические функции

### 2.7.2.1 И

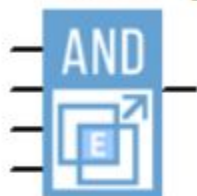
<p><b>В001 [M1]</b></p> 	<p>Выход блока переключается в состояние логической единицы, только если логическая единица действует на всех входах блока одновременно.</p>
---	--

Временная диаграмма



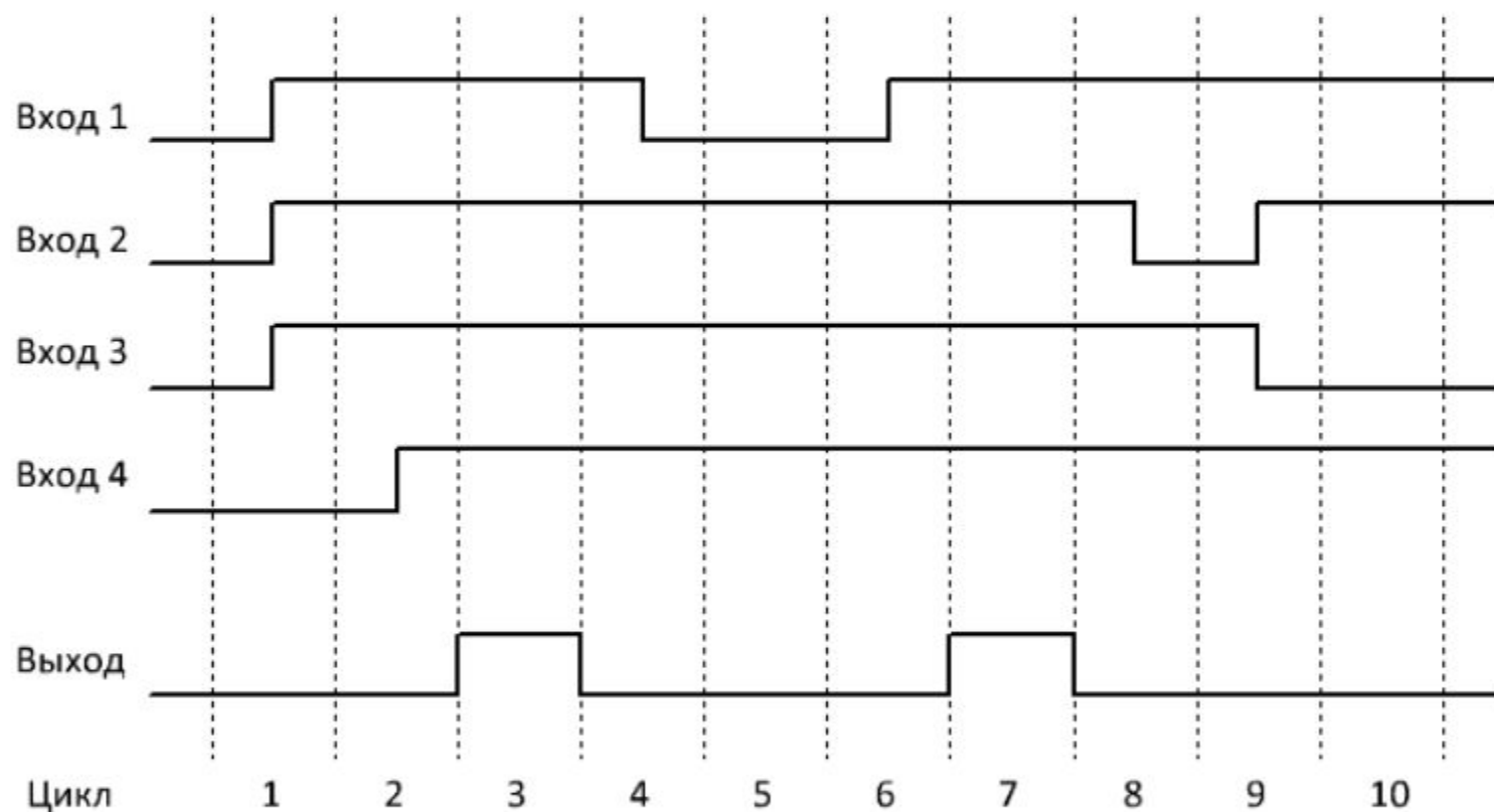
## 2.7.2.2 И (по фронту)

**B002[M2]**



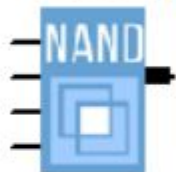
Выход блока переключается в состояние логической единицы на один цикл программы, только если логическая единица действует на всех входах блока одновременно, но при условии, что по крайней мере один вход был в состоянии логического нуля в предыдущем цикле программы.

Временная диаграмма



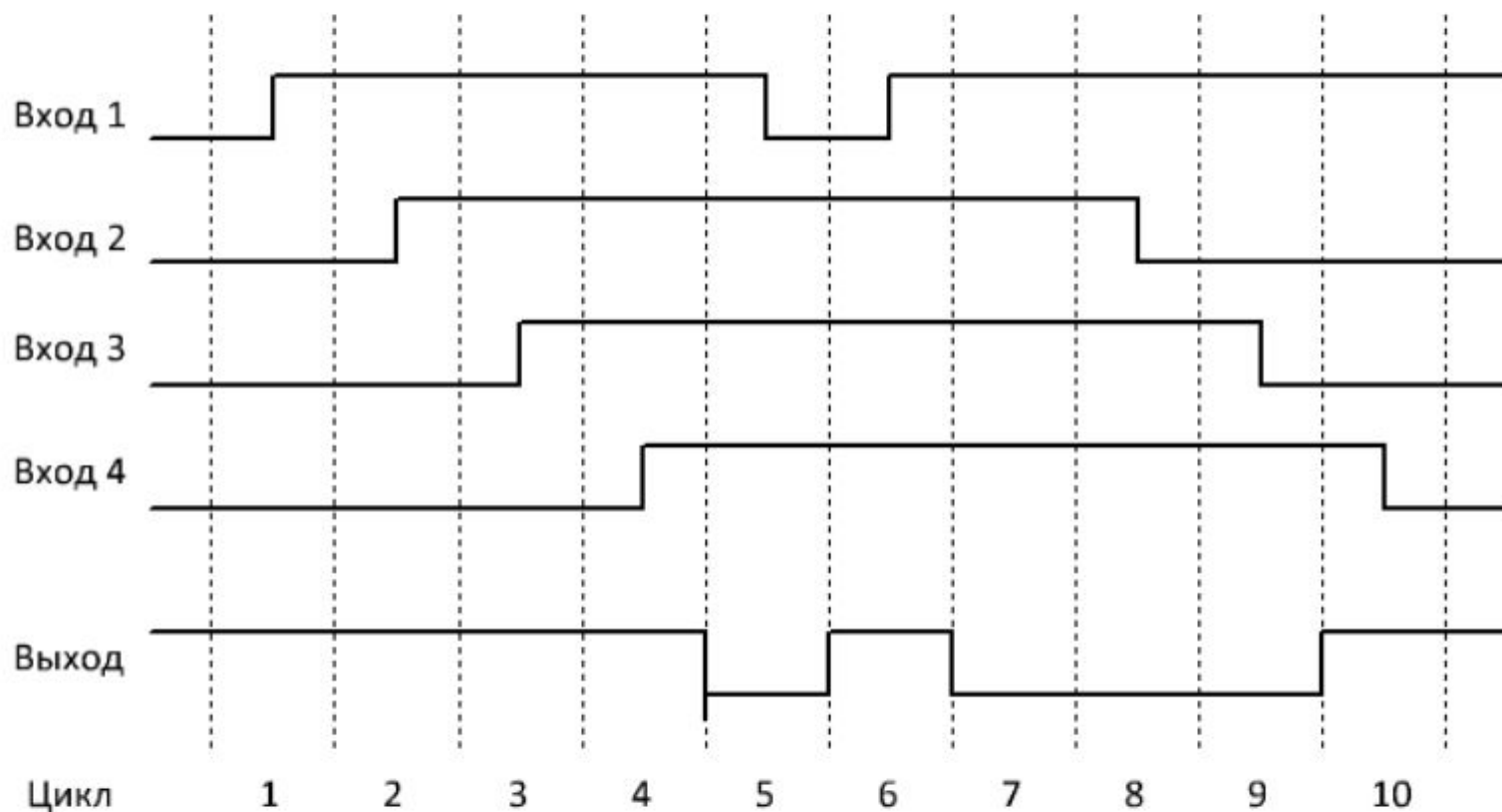


**B001[M1]**



Выход блока переключается в состояние логического нуля, только если логическая единица действует на всех входах блока одновременно.

### Временная диаграмма



### Примечание:

Значения на незадействованных в программе входах блока по умолчанию соответствуют логиче-

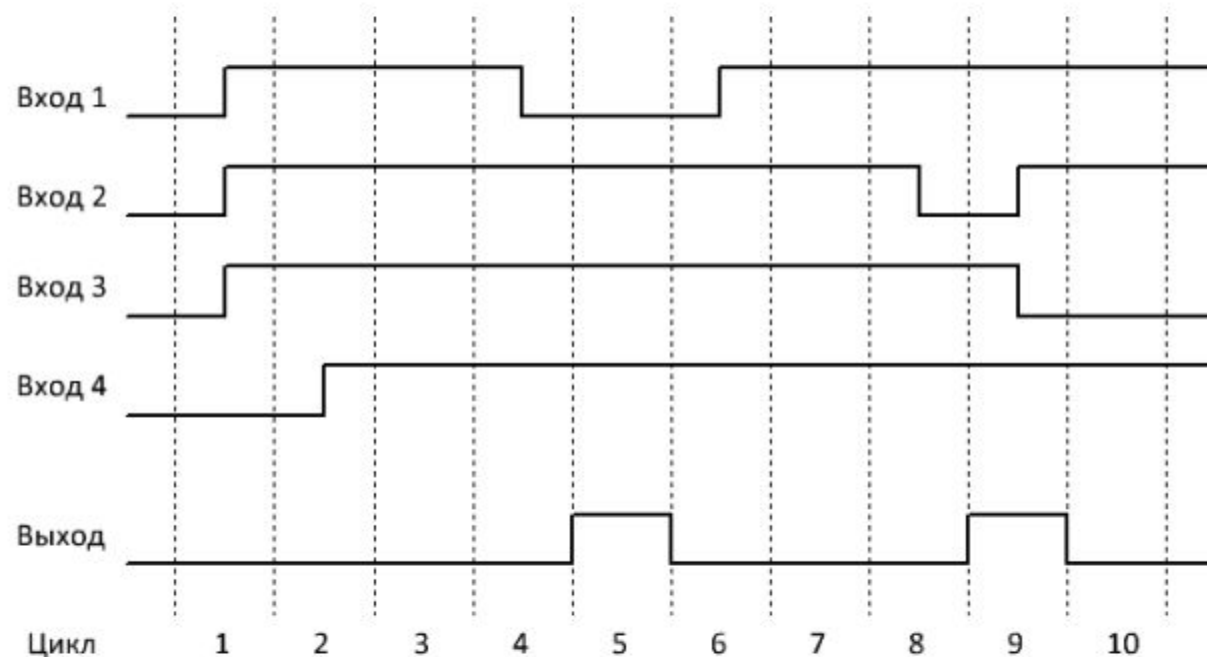
## 2.7.2.4 И-НЕ (по фронту)

B001[M1]



Выход блока переключается в состояние логической единицы на один цикл программы, если логический ноль действует хотя бы на одном входе блока, при условии, что все входы были в состоянии логической единицы в течении предыдущего цикла программы.

### Временная диаграмма

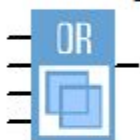


### Примечание:

Значения на незадействованных в программе входах блока по умолчанию соответствуют логической единице.

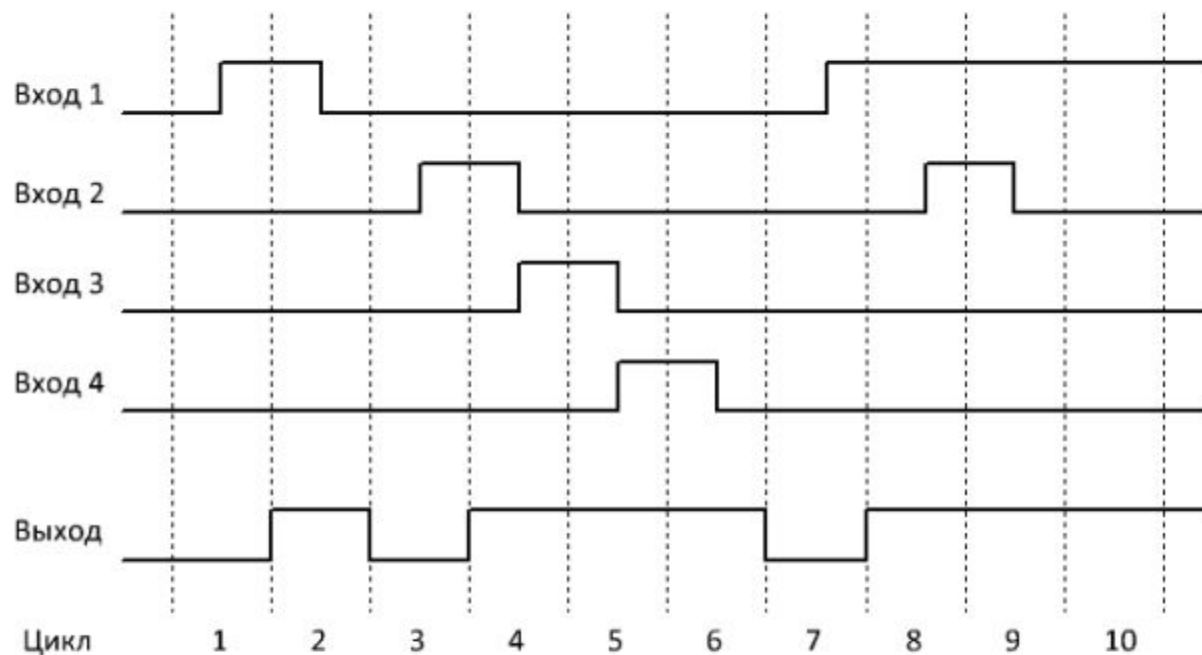
## 2.7.2.5 ИЛИ

B001[M1]



Выход блока переключается в состояние логической единицы, если логическая единица действует как минимум на одном входе блока.

### Временная диаграмма

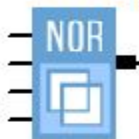


### Примечание:

Значения на незадействованных в программе входах блока по умолчанию соответствуют логическому нулю.

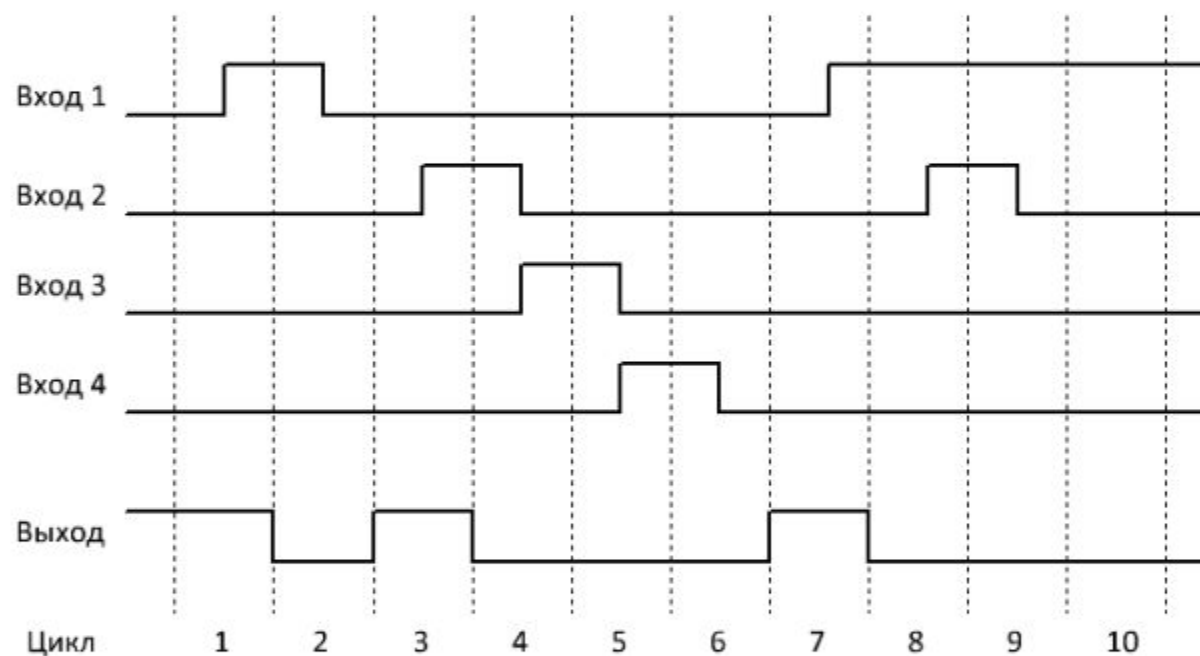
## 2.7.2.6 ИЛИ-НЕ

B002[M2]



Выход блока переключается в состояние логической единицы, только если все входы блока переведены в состояние логического нуля.


Временная диаграмма



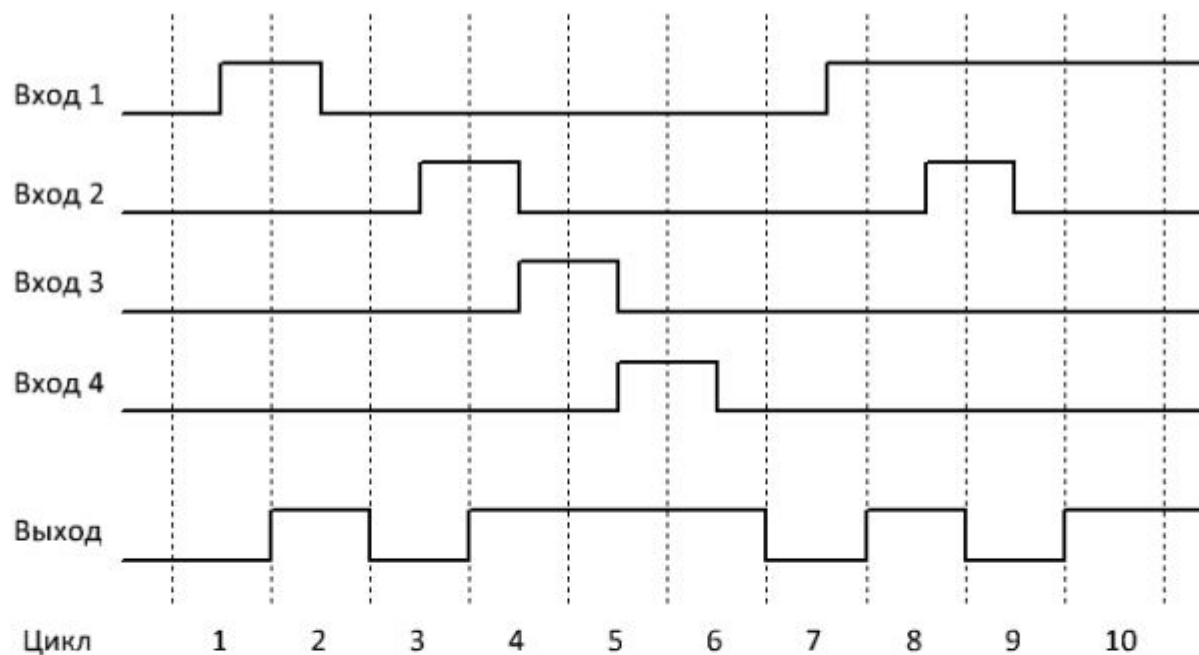
**Примечание:**

Значения на незадействованных в программе входах блока по умолчанию соответствуют логическому нулю.

## 2.7.2.7 Исключающее ИЛИ

<p><b>В001[M1]</b></p> 	<p>Выход блока переключается в состояние логической единицы, если логическая единица действует только на одном входе модуля.</p>
--	--

Временная диаграмма



**Примечание:**

Значения на незадействованных в программе входах блока по умолчанию соответствуют логическому нулю.

## 2.7.2.8 НЕ



### Временная диаграмма

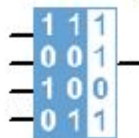


### Примечание:

Значения на незадействованных в программе входах блока по умолчанию соответствуют логическому нулю.

## 2.7.2.9 Настраиваемая логика

В001[M1]



Настраиваемый блок логика работы которого определяется таблицей истинности заданной пользователем в процессе разработки проекта.

На рисунке ниже приведен пример конфигурации таблицы истинности, который соответствует логике элемента 4И (Логическое И по четырем входам).

В001[M1][Настраиваемая булева логика] ✕

Параметры | Комментарии

Имя блока:   Показывать параметры

Варианты

Выход 0 если результат TRUE

Выход 1 если результат TRUE

Индекс	Вход 1	Вход 2	Вход 3	Вход 4	Выход
1	0	0	0	0	0
2	1	0	0	0	0
3	0	1	0	0	0
4	1	1	0	0	0
5	0	0	1	0	0
6	1	0	1	0	0
7	0	1	1	0	0
8	1	1	1	0	0
9	0	0	0	1	0
10	1	0	0	1	0
11	0	1	0	1	0
12	1	1	0	1	0
13	0	0	1	1	0
14	1	0	1	1	0
15	0	1	1	1	0
16	1	1	1	1	1

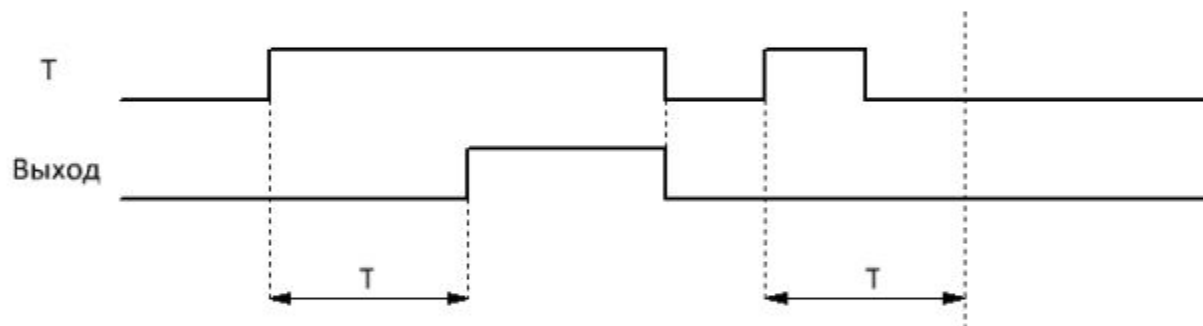
## 2.7.3 Специальные функции

### 2.7.3.1 Временные

#### 2.7.3.1.1 Задержка включения

<b>В001 [M1]</b> 	<p>Функциональный блок обеспечивает задержку появления сигнала на выходе с момента появления логической единицы на входе Т на время заданной уставки Т.</p> <p>Если длительность входного сигнала менее заданной уставки, переключения выхода блока не произойдет.</p>
---	--

#### Временная диаграмма



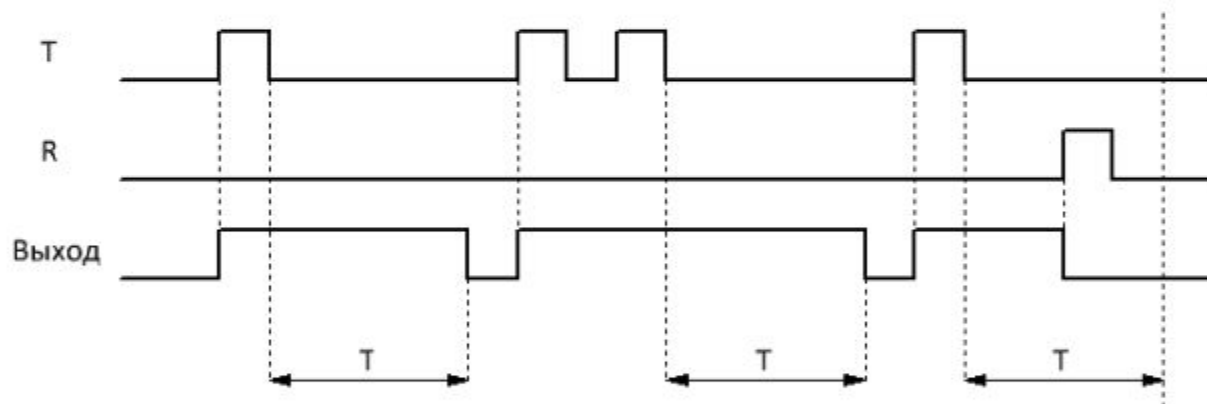
Уставка задержки включения (Т) задается на вкладке "параметры" в окне свойств блока, и может быть определена как постоянная, или как переменная величина.



### 2.7.3.1.2 Задержка выключения

<p><b>V001 [M1]</b></p> 	<p>Функциональный блок обеспечивает задержку переключения выхода в состояние логического нуля с момента исчезновения логической единицы на входе Т на время заданной уставки Т.</p> <p>Если в течении временного интервала задержки отключения на вход Т блока будет вновь подан сигнал логической единицы, отсчет прекратится и возобновиться заново при очередном переходе входа из состояния логической единицы в состояние логического нуля.</p> <p>Вход R выполняет сброс таймера и переводит выход блока в состояние логического нуля.</p>
---	--

Временная диаграмма





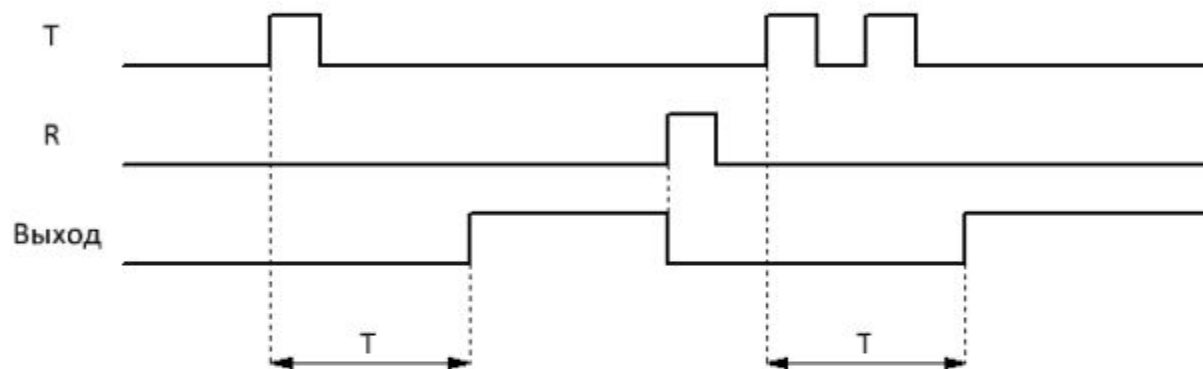
**B001 [M1]**



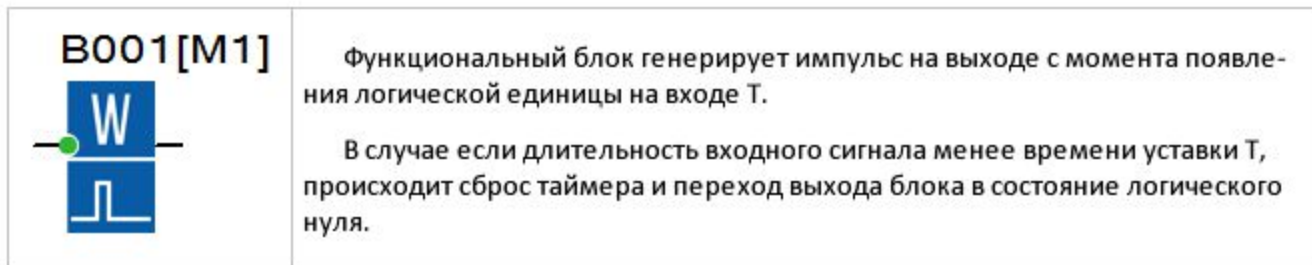
Функциональный блок обеспечивает задержку появления сигнала на выходе с момента появления логической единицы на входе Т на время заданной уставки Т, но в отличие от таймера задержки включения, последующие изменения состояния входа Т в течении отсчета времени не влияют на работу таймера.

Вход R выполняет сброс таймера и переводит выход блока в состояние логического нуля.

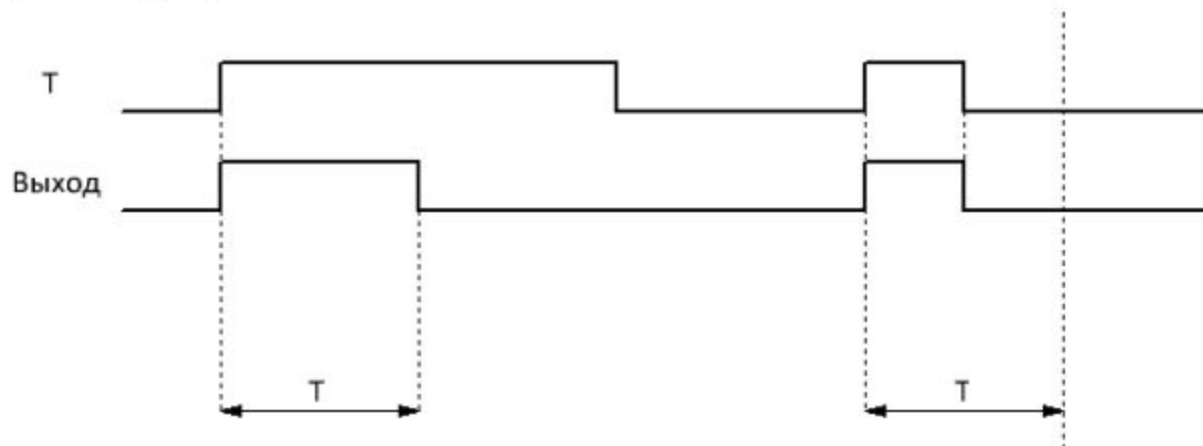
### Временная диаграмма



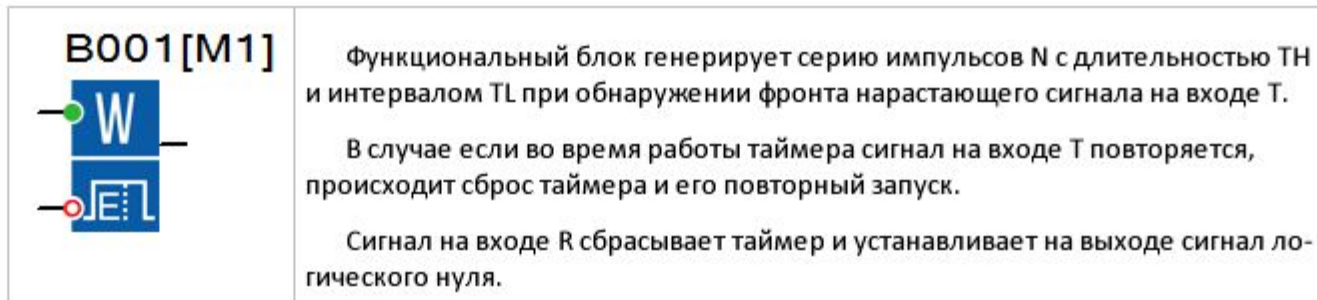
### 2.7.3.1.5 Генератор одиночного импульса



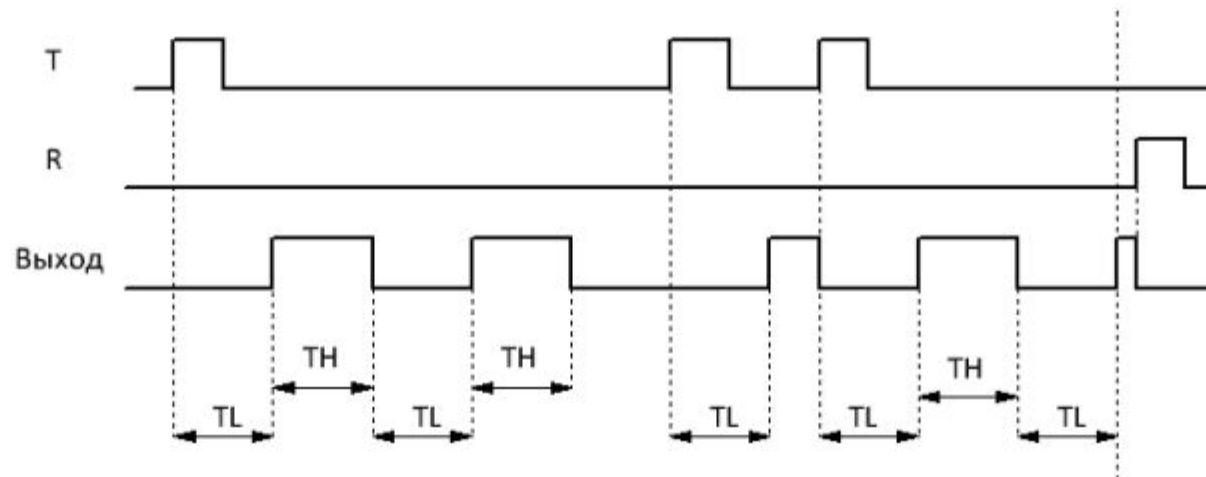
Временная диаграмма



### 2.7.3.1.6 Генератор серии импульсов



**Временная диаграмма**



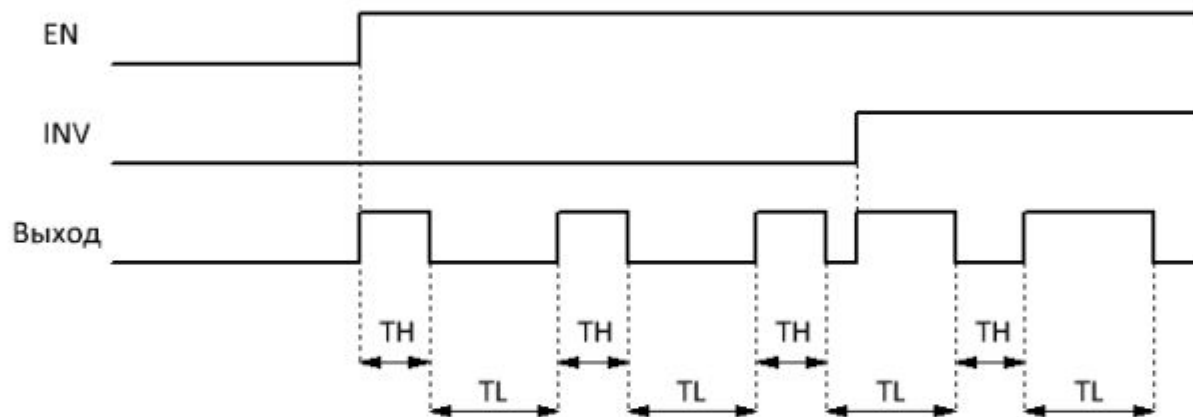
**В001 [M1]**



При появлении сигнала логической единицы на входе EN блока, на выходе формируется импульсная последовательность с заданными параметрами TH и TL.

Вход INV используется для выполнения инверсии выходного сигнала при его переключении в состояние логической единицы.

**Временная диаграмма**



### 2.7.3.1.9 Выключатель освещения

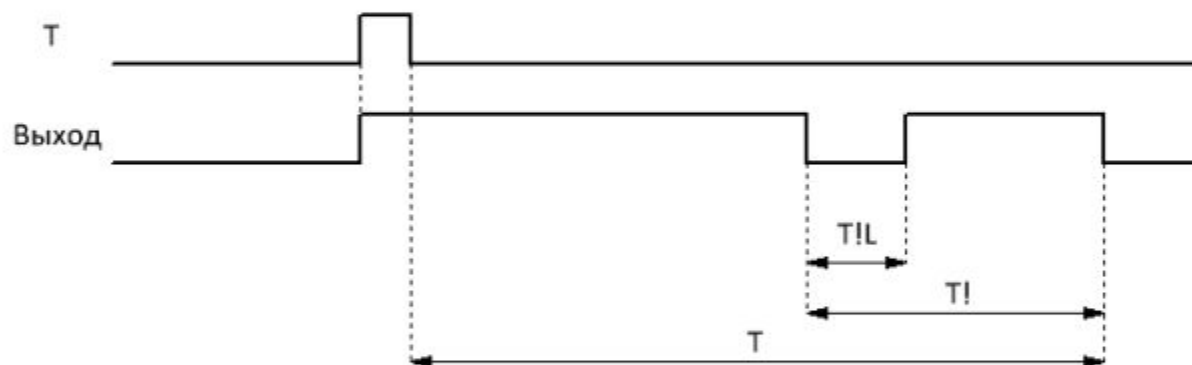
**B001 [M1]**



При обнаружении фронта нарастающего сигнала на входе  $T$ , функциональный блок формирует логическую единицу на выходе в течении времени, заданного уставкой  $T$ .

Дополнительно предусмотрена функция предупреждения о предстоящем выключении, которая предусматривает перевод выхода блока в состояние логического нуля на время  $T!L$  за время  $T!$  до окончания основного интервала.

**Временная диаграмма**



## **23.Примеры выполнения программ**



# Управление освещением лестничных клеток

## Описание задачи

Как правило, освещение лестничных клеток многоэтажных зданий включено постоянно независимо от времени суток, что противоречит общим тенденциям по повышению энергоэффективности. Поэтому в настоящее время, в качестве основной меры по экономии электроэнергии, довольно часто используются различные датчики движения. Которые отключают освещение на лестничной площадке при отсутствии движения.

Не смотря на кажущуюся простоту и экономичность, такие схемы имеют недостатки. Например при индивидуальном включении ламп, неизбежен момент, когда, поднимаясь по лестнице, вы попадаете с освещённой лестничной площадки на неосвещённую, т.к. датчик, который должен включить свет, еще не обнаружил движения. Подобная ситуация не только вызывает дискомфорт, но и опасна, особенно при полном отсутствии иных источников света и в ночное время. В случае если датчики движения включены параллельно и включают одновременно освещение на всех этажах, передвижение становится более комфортным, но данное решение является избыточным, так как не все передвижения ведут на последний этаж. При этом снижается энергоэффективность и срок службы ламп, как следствие включения лишних потребителей и дополнительной коммутации питания.

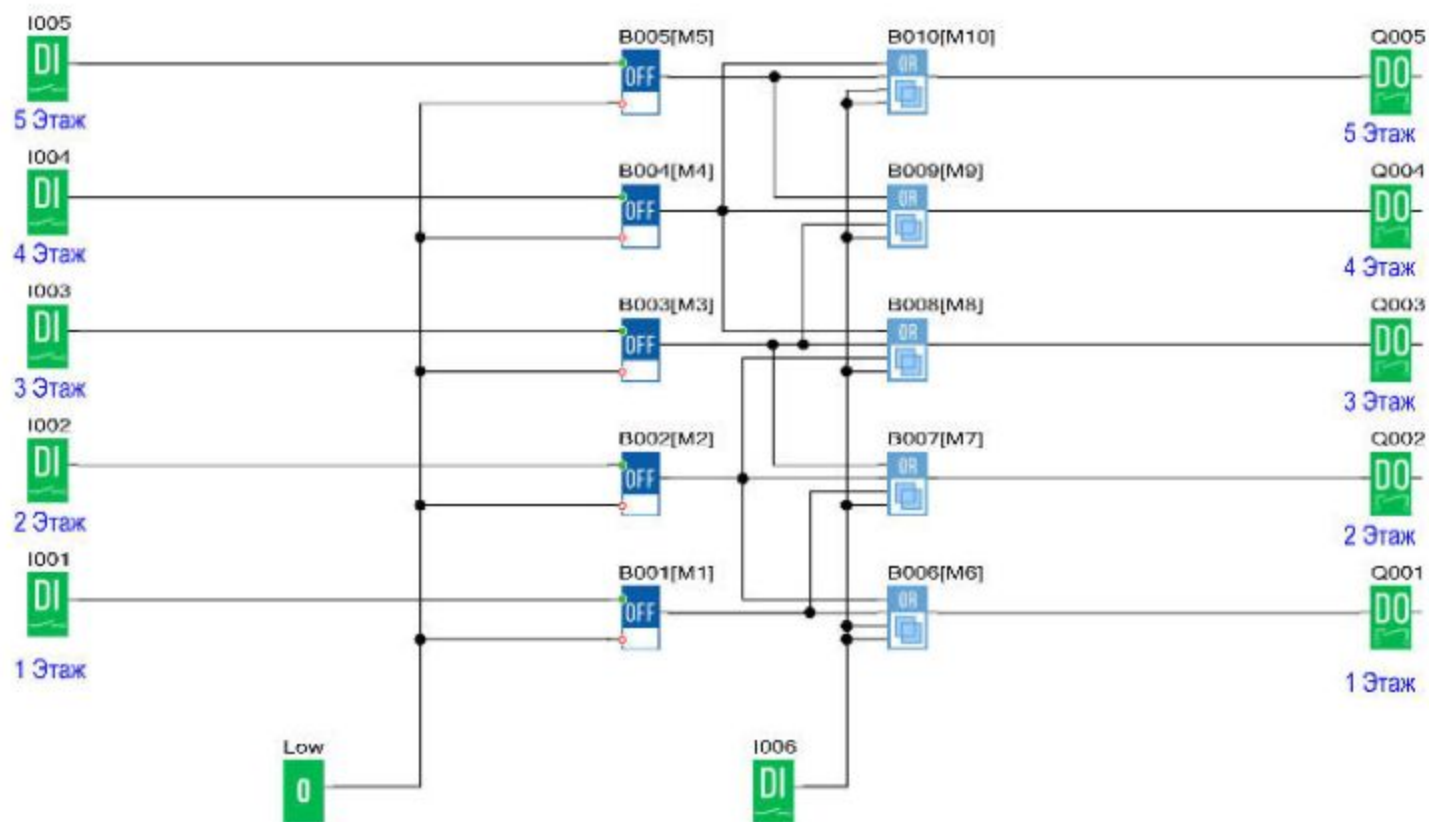
## Предлагаемое решение

С помощью программируемого логического реле ONI PLR-S можно повысить эффективность управления освещением, сделав его более интеллектуальным. Ниже представлен пример программы реализации управления освещением для 5 этажного здания и описание алгоритма работы.

Этажные датчики движения

Таймеры задержки отключения

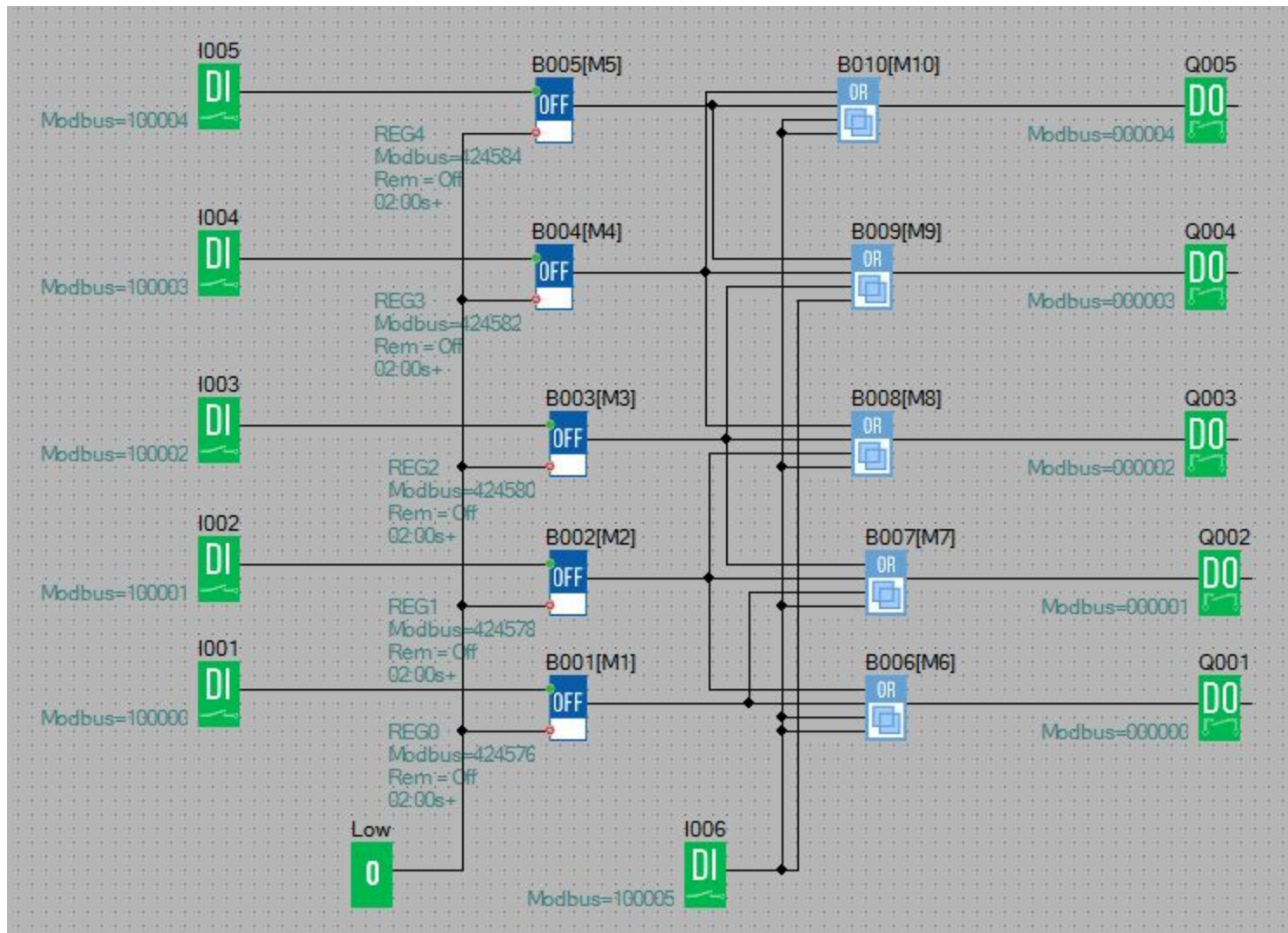
Лампы освещения



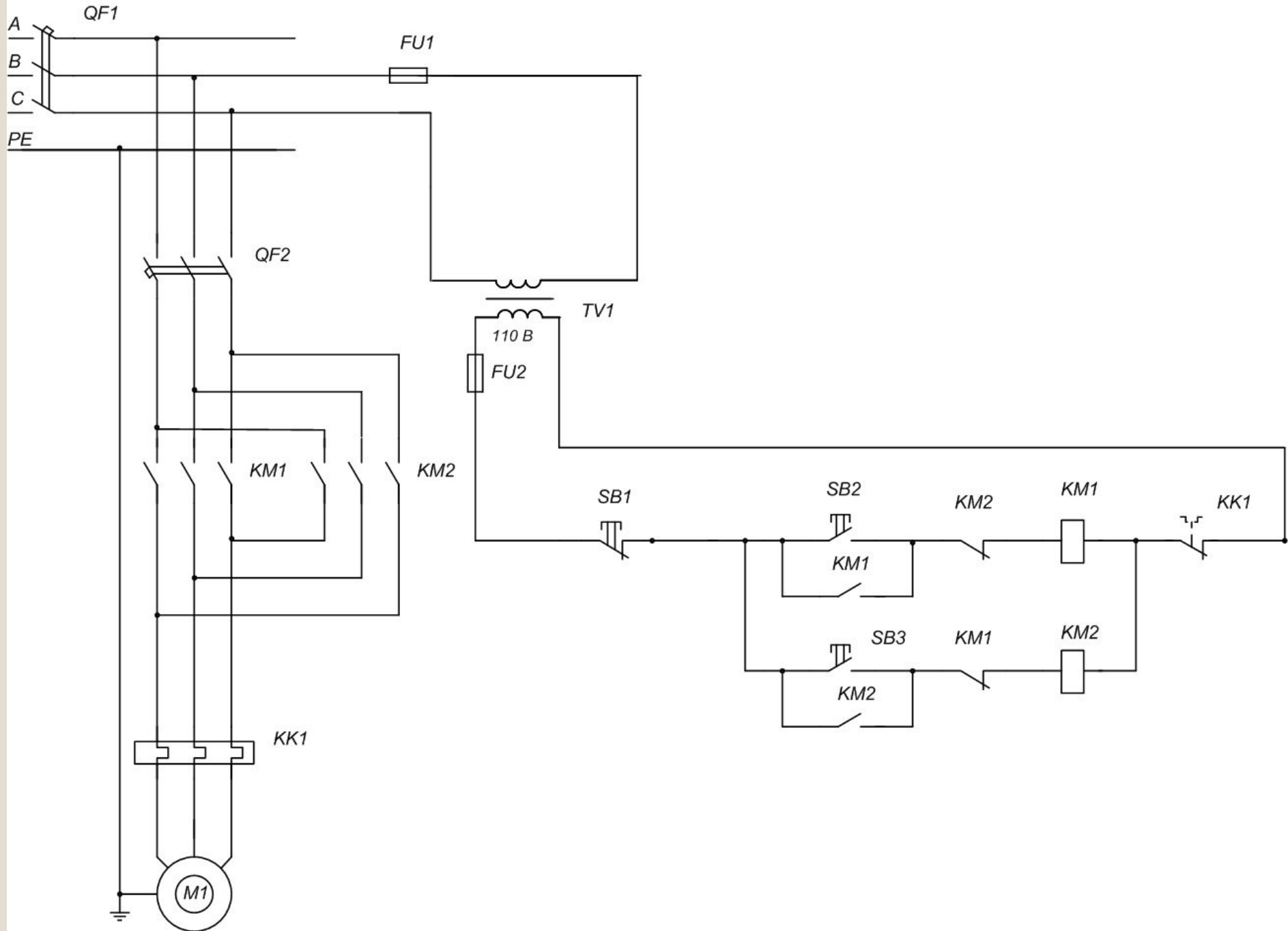
Принудительное включение освещения

## **Алгоритм работы программы**

Срабатывание этажного датчика движения включает освещение не только на этаже, где датчик непосредственно установлен, но одновременно и на смежных этажах. Тем самым повышается комфорт передвижения, и не задействуются излишне избыточные ресурсы, как в случае с одновременным включением освещения на всех этажах. Дополнительно предусмотрено включение освещения вручную, нажатием кнопки-выключателя.

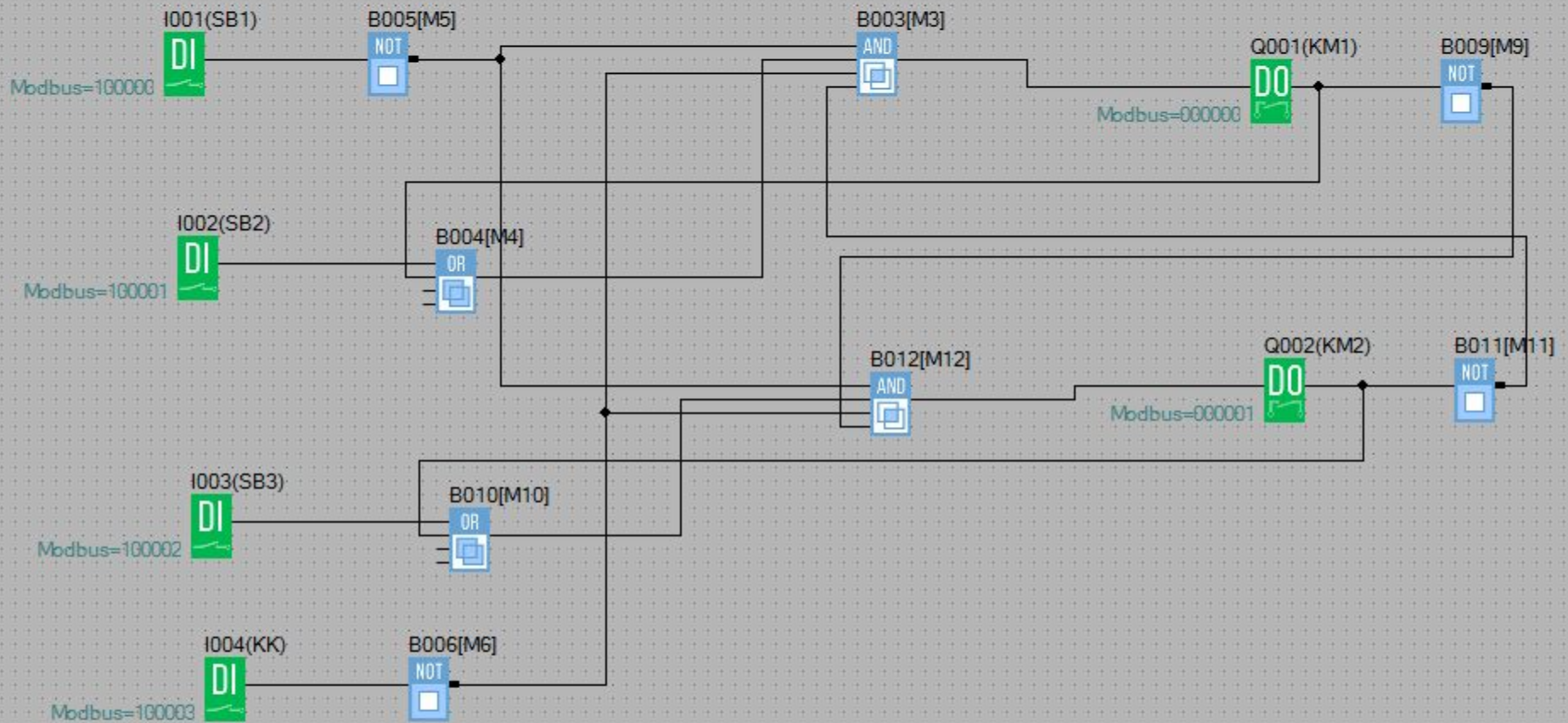


# **Управление асинхронным двигателем**

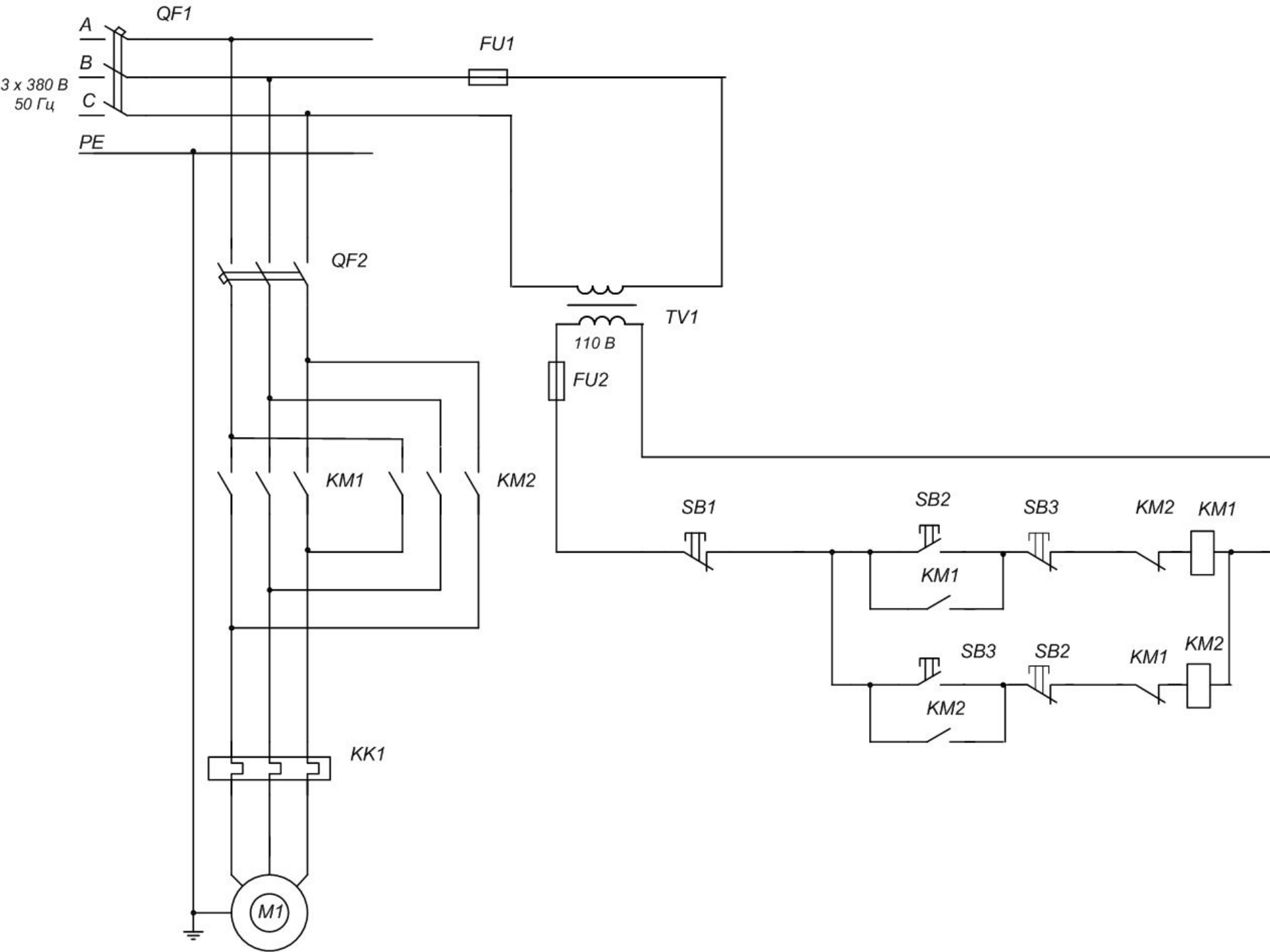


$$KM1 = \overline{SB1} \cdot (SB2 + KM1) \cdot \overline{KM2} \cdot \overline{KK1}$$

$$KM2 = \overline{SB1} \cdot (SB3 + KM2) \cdot \overline{KM1} \cdot \overline{KK1}$$







$$KM1 = \overline{SB1} \cdot (SB2 + KM1) \cdot \overline{SB3} \cdot \overline{KM2} \cdot \overline{KK1}$$

$$KM2 = \overline{SB1} \cdot (SB3 + KM2) \cdot \overline{SB2} \cdot \overline{KM1} \cdot \overline{KK1}$$

