



## Логические модули LOGO!

---



2016

УДК681.5.08(075)  
ББК973.26-04я73

Составил: **С.И. Макаренко**  
Логические модули LOGO! Пособие:/ X:- 2016, 23 с

М31

В пособии представлены сведения о современных модульных технических устройствах LOGO! производства Siemens, использующихся в автоматизированных системах управления технологическими процессами.

ISBN 5-94275-104-8

## Содержание

1. Логические модули LOGO! для промышленной автоматизации . . . . .	3
2. Составление управляющей программы программируемого контроллера . . . . .	10
3. Применение микропроцессорных систем в электротехнике на примере использования ПЛК . . . . .	15
4. Автоматизация системы вентиляции на ПЛК LOGO! . . . . .	18
Литература . . . . .	23

# 1. Логические модули LOGO! для промышленной автоматизации

Микропроцессорные устройства широко применяются при автоматизации общепромышленных, транспортных и бытовых устройств. Благодаря универсальности и низкой стоимости микропроцессорных устройств их доля в устройствах автоматики постоянно увеличивается. На начальном этапе применения микропроцессорных устройств основным сдерживающим фактором являлись, при низкой стоимости самих микроконтроллеров, значительные затраты на создание их программного обеспечения, которое разрабатывалось на языках программирования низкого уровня и требовало высококвалифицированных программистов.

Решить данную проблему позволило создание функционально законченных микропроцессорных модулей со встроенным базовым программным обеспечением и дополнительными модулями расширения. Связь базовых модулей с модулями расширения осуществляется через специальные разъемы, исключающие подключение модулей, которые по каким либо критериям (например, напряжению питания) не могут быть соединены с базовым модулем.

Программирование модулей ведется на специализированных языках высокого уровня типа Step 5 или Step 7, которые позволяют составить программу в виде структурной схемы, или контактной схемы, или в виде системы логических уравнений. Компиляция таких программ в машинные коды ведется с учетом конкретной номенклатуры установленных модулей. Программисту не требуются специальные знания о структуре и командах микропроцессоров, входящих в модули, а требуются лишь знания о функционировании разрабатываемой технической системы.

Фирма разработчик модулей создает специализированное программное обеспечение для персонального компьютера с удобным интерфейсом, которое обеспечивает все этапы разработки системы и программирование микропроцессорных модулей непосредственно через порты персонального компьютера или подключаемое к компьютеру дополнительное устройство. Такая концепция реализована фирмой SIEMENS при создании набора микропроцессорных модулей LOGO!.



Рис.1. Логический модуль LOGO!

**LOGO! - это универсальный логический микропроцессорный модуль фирмы Siemens.**

LOGO! включает в себя устройство управления на микропроцессоре, панель управления и индикации с фоновой подсветкой, блок питания, интерфейс для модулей расширения, интерфейс для программного модуля (плата) и кабель PC.

В модуль LOGO! заложены стандартные готовые функции, часто используемые на практике, например: функции задержки включения и выключения, импульсное реле, программируемые клавиши, часовой выключатель, цифровые и аналоговые флаги, входы и выходы в соответствии с типом устройства.

## Типы устройств LOGO!



Basic (базисный модуль) выпускается для двух классов напряжения: Класс 1 < 24 В, т.е. 12 В пост. тока, 24 В пост. тока, 24 В перемен. тока; Класс 2 > 24 В, т.е. 115...240 В пост. и перем. тока; в вариантах:

с ЖК дисплеем (ЖКД): 8 входов и 4 выхода;

без дисплея ("LOGO! Pure [Чистый LOGO!]"): 8 входов и 4 выхода.

Каждый класс состоит из 4 субблоков (SU), снабжен интерфейсом для подключения расширения и предоставляет в ваше распоряжение 33 готовых к использованию основных и специальных функций для

разработки программы коммутации.

### Модули расширения

- Цифровые модули LOGO! выпускаются на все напряжения и имеют 4 входа и 4 выхода.
- Аналоговые модули LOGO! выпускаются на напряжения 12 и 24 В постоянного тока с двумя аналоговыми входами или двумя входами РТ100.
- Цифровые и аналоговые модули состоят из двух субблоков. Каждый из них имеет два интерфейса расширения для подключения дополнительных модулей. Каждое базовое устройство LOGO! Basic может быть расширено только с помощью модулей расширения того же класса напряжения. Подключению друг к другу устройств, относящихся к различным классам напряжения, препятствует механическое кодирование (штифты в корпусе). Исключение: левый интерфейс аналогового или коммуникационного модуля имеет потенциальную развязку. Поэтому эти модули расширения могут соединяться с устройствами другого класса напряжения.

### Элементы в обозначении LOGO!

Модули LOGO! Отличаются видом (постоянное = или переменное ~) и значением питающего напряжения, типом выходов (релейные или транзисторные), наличием или отсутствием жидкокристаллического дисплея. Многообразие модулей LOGO! позволяет выбрать наиболее подходящий набор, с минимальным избытком технических средств реализующий конкретную техническую задачу.

Обозначение элементов:





- 12 – вариант на 12 В постоянного тока.
- 24 – вариант на 24 В постоянного тока.
- 230 – вариант на 115/240 В переменного тока.
- R – релейные выходы (без R - транзисторные выходы).
- С – встроенный часовой выключатель на 7 дней.
- о – вариант без дисплея.
- DM – цифровой модуль.
- AM – аналоговый модуль.
- CM – коммуникационный модуль (напр., AS interface).
- 

### Варианты модулей LOGO!



Рис.2 Базовый модуль LOGO! с модулями расширения.

Таблица 1. Технические данные модулей LOGO!

	Обозначение	Напряжение	Входы	Выходы	Свойства
	LOGO! 12/24 RC	=12/24 В	8 цифровых <sup>(1)</sup>	4 релейных (10 А)	
	LOGO! 24	=24 В	8 цифровых <sup>(1)</sup>	4 транзисторных 24В /0,3 А	Без часов
	LOGO! 24 RC <sup>(3)</sup>	~/= 24 В	8 цифровых	4 релейных (10 А)	
	LOGO! 230 RC <sup>(2)</sup>	~/= 115...240 В	8 цифровых	4 релейных (10 А)	
	LOGO! 12/24 RC <sub>o</sub>	= 12/24 В	8 цифровых <sup>(1)</sup>	4 релейных (10 А)	Без ЖКД и клавиатуры
	LOGO! 24 <sub>o</sub>	= 24 В	8 цифровых <sup>(1)</sup>	4 транзисторных 24В /0,3 А	Без часов, дисплея и клавиатуры
	LOGO! 24 RC <sub>o</sub> <sup>(3)</sup>	~/= 24 В	8 цифровых	4 релейных (10 А)	Без ЖКД и клавиатуры
	LOGO! 230 RC <sub>o</sub> <sup>(2)</sup>	~/= 115...240 В	8 цифровых	4 релейных (10 А)	Без ЖКД и клавиатуры
<b>Модули расширения</b>					
	Обозначение	Напряжение	Входы	Выходы	
	LOGO! DM 8 12/24 R	= 12/24 В	4 цифровых	4 релейных (5 А)	
	LOGO! DM 8 24	= 24 В	4 цифровых	4 транзисторных 24В/0,3А	
	LOGO! DM 8 24 R <sup>(3)</sup>	~/= 24 В	4 цифровых	4 релейных (5 А)	
	LOGO! DM 8 230 R	~/= 115-240В	4 цифровых <sup>(1)</sup>	4 релейных (5 А)	
	LOGO! AM 2	= 12/24 В	2 аналоговых 0...10 В или 0...20 мА <sup>(4)</sup>	Нет	
	LOGO! AM 2 PT100	= 12/24 В	2 PT100 от -50 °С до +200 °С	Нет	

(1) – из них альтернативно можно использовать 2 аналоговых входа с диапазоном сигнала 0...10 В и 2 быстрых входа. (2) – варианты на 230 В переменного тока - входы двумя группами по 4. Внутри группы возможна только одна и та же фаза, между группами возможны разные фазы. (3) – цифровые входы могут работать с прямой и обратной полярностью. (4) – по можно выбрать диапазон сигнала 0 ... 10 В, или 0 ... 20 мА.

### Функции LOGO!

LOGO! в режиме программирования предоставляет в ваше распоряжение различные элементы, которые разделены на списки:

- CO – список соединительных элементов (Входов/выходов)
- GF – список основных функций AND [И], OR [ИЛИ],
- SF – список специальных функций
- BN – список готовых к использованию в коммутационной программе блоков.

Все списки отображают элементы, доступные в LOGO!. Обычно это все соединительные элементы, все основные функции и все специальные функции, которые известны LOGO!. Сюда включаются также все блоки, которые вы создали в LOGO! ко времени вызова списка. LOGO! отображает не все элементы, если нет свободного места в памяти, или достигнуто максимально возможное количество блоков. В этом случае нельзя вставить следующий блок.

Константы и соединительные элементы (Co) - это входы, выходы, биты памяти и фиксированные уровни напряжения (константы).

### Входы:

1) Цифровые входы

Цифровые входы обозначаются буквой I. Номера цифровых входов (I1, I2,...) соответствуют номерам входных выводов на LOGO! Нумерация входов на базовом модуле и модулях расширения сквозная в порядке установки модулей.

2) Аналоговые входы

У вариантов LOGO! 24, LOGO! 24<sub>o</sub>, LOGO! 12/24RC и LOGO! 12/24RC<sub>o</sub> имеются входы I7 и I8, которые могут быть также запрограммированы для использования в качестве аналоговых входов AI1 и AI2. Если эти входы используются как I7 и I8, то входной сигнал интерпретируется как цифровая величина. Если они используются как AI1 и AI2, то сигналы интерпретируются как

аналоговые величины. При подключении аналогового модуля его входы получают номера, следующие за существующими аналоговыми входами.

В случае специальных функций, которые на стороне входов имеет смысл соединять только с аналоговыми входами, при выборе в режиме программирования входного сигнала предлагаются только аналоговые входы AI1...AI8, аналоговые флаги AM1...AM6, аналоговые выходы модулей, которые их имеют, нумеруются как выходы AQ1 и AQ2.

#### Выходы:

##### 1) Цифровые выходы

Цифровые выходы обозначаются буквой Q. Номера выходов (Q1, Q2,... Q16) соответствуют номерам выходных выводов на LOGO!. Номера выходов имеют сквозную нумерацию, начиная с базового модуля и далее в порядке установки модулей. Кроме того, имеется возможность использования 16 неподключенных к блокам выходов. Они обозначены символом X и не могут повторно использоваться в коммутационной программе (в отличие, например, от флагов). В списке появляются все запрограммированные неподключенные выходы, а также один еще не запрограммированный неподключенный выход. Использование неподключенного выхода имеет смысл, например, у специальной функции «Тексты сообщений», если только текст сообщения имеет значение для коммутационной программы.

##### 2) Аналоговые выходы

Аналоговые выходы обозначаются буквами AQ. Имеются в распоряжении два аналоговых выхода, а именно, AQ1 и AQ2. К аналоговому выходу можно подключать только аналоговую величину, т.е. функцию с аналоговым выходом или аналоговый флаг AM.

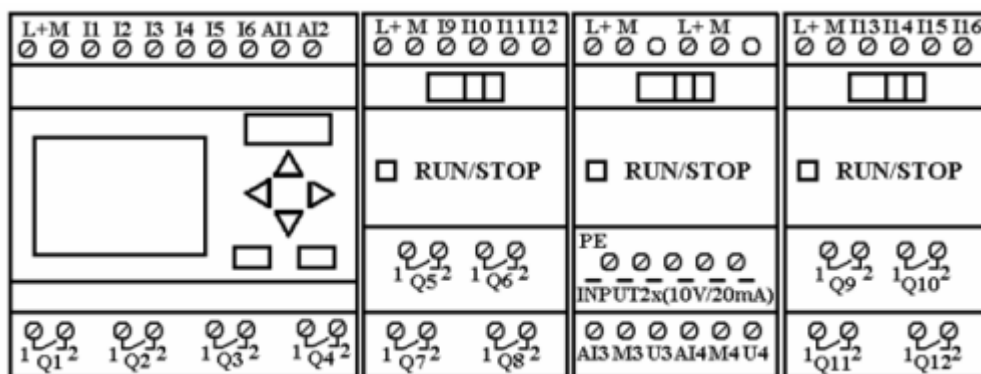


Рис. 3. Вид на лицевые панели модулей LOGO!

#### Флаги

Флаги обозначаются буквами M или AM. Это виртуальные выходы, которые имеют на своем выходе такое же значение, как и на своем входе. В LOGO! имеется 24 цифровых флага M1 ... M24 и 6 аналоговых флагов AM1 ...AM6.

Флаг запуска M8 устанавливается в первом цикле работы программы пользователя и, следовательно, может использоваться в вашей коммутационной программе как флаг запуска. Он автоматически сбрасывается после первого цикла обработки программы. Во всех последующих циклах флаг M8 может использоваться таким же образом, как и другие флаги.

#### Логические уровни сигналов

Уровни сигналов обозначаются hi и lo. Если на блоке должно постоянно иметь место состояние «1» = hi или «0» = lo, то на вход подается фиксированный уровень или постоянное значение hi или lo. Открытые соединительные элементы Если соединительный элемент блока не используется, то его можно обозначить символом x.

или lo. Открытые соединительные элементы Если соединительный элемент блока не используется, то его можно обозначить символом x.

#### Список основных функций - GF

Основные функции - это простые логические элементы булевой алгебры.

Список GF содержит блоки основных функций, которые вы можете использовать в своей коммутационной программе. Имеются следующие основные функции:

Представление на коммутационной схеме	Представление в LOGO	Наименование функции
 Последовательное соединение		<b>AND (И)</b>
		<b>AND (И) с анализом фронта</b>
 Параллельное соединение размыкающих контактов		<b>NAND (И-НЕ)</b>
		<b>NAND (И-НЕ) с анализом фронта</b>
 Параллельное соединение замыкающих контактов		<b>OR (ИЛИ)</b>
 Последовательное соединение размыкающих контактов		<b>NOR (ИЛИ-НЕ)</b>
 Двойной перекидной контакт		<b>XOR (исключающее ИЛИ, сумма по модулю 2)</b>
 Размыкающий контакт		<b>NOT (отрицание, инверсия)</b>

**Рис.4. Функции модулей**

#### Список специальных функций – SF

При вводе коммутационной программы в LOGO! вы найдете блоки для специальных функций в списке SF. Входы специальных функций можно инвертировать по отдельности, т.е. коммутационная программа преобразует логическую «1» на входе в логический «0»; а логический «0» она преобразует в логическую «1». В таблице указано, обладает ли соответствующая функция параметрируемой сохраняемостью (REM).

Имеются следующие специальные функции:

- Задержка включения
- Задержка выключения
- Задержка включения / выключения
- Задержка включения с запоминанием
- Интервальное реле времени (формирование короткого импульса)
- Интервальное реле времени, запускаемое фронтом
- Асинхронный генератор импульсов
- Генератор случайных импульсов
- Выключатель света на лестничной клетке
- Двухфункциональный выключатель
- Семидневный часовой выключатель
- Двенадцатимесячный часовой выключатель
- Реверсивный счетчик
- Счетчик рабочего времени
- Пороговый выключатель
- Аналоговый пороговый выключатель
- Аналоговый разностный пороговый выключатель
- Аналоговый компаратор
- Контроль аналоговых величин
- Аналоговый усилитель
- Самоблокирующее реле (RS-триггер)
- Импульсное реле
- Программный выключатель
- Регистр сдвига

## Программное обеспечение LOGO!

В качестве программного пакета для ПК имеется программа LOGO!Soft Comfort. Это программное обеспечение включает в себя следующие функции:

- графический интерфейс для создания коммутационной программы в режиме offline в виде цепной логической схемы (контактного плана / коммутационной схемы) или функциональной блок-схемы (функционального плана);
- имитация вашей коммутационной программы на компьютере;
- генерирование и распечатка блок-схемы для коммутационной программы;
- сохранение программы на жестком диске или ином средстве хранения;
- сравнение коммутационных программ;
- удобная параметризация блоков;
- передача коммутационной программы из LOGO! в ПК и из ПК в LOGO!;
- считывание счетчика рабочего времени;
- установка времени;
- переход с летнего времени на зимнее время и обратно;
- тестирование в режиме online, отображение состояний и текущих значений переменных LOGO! В режиме RUN;
- остановка выполнения коммутационной программы из ПК (STOP).

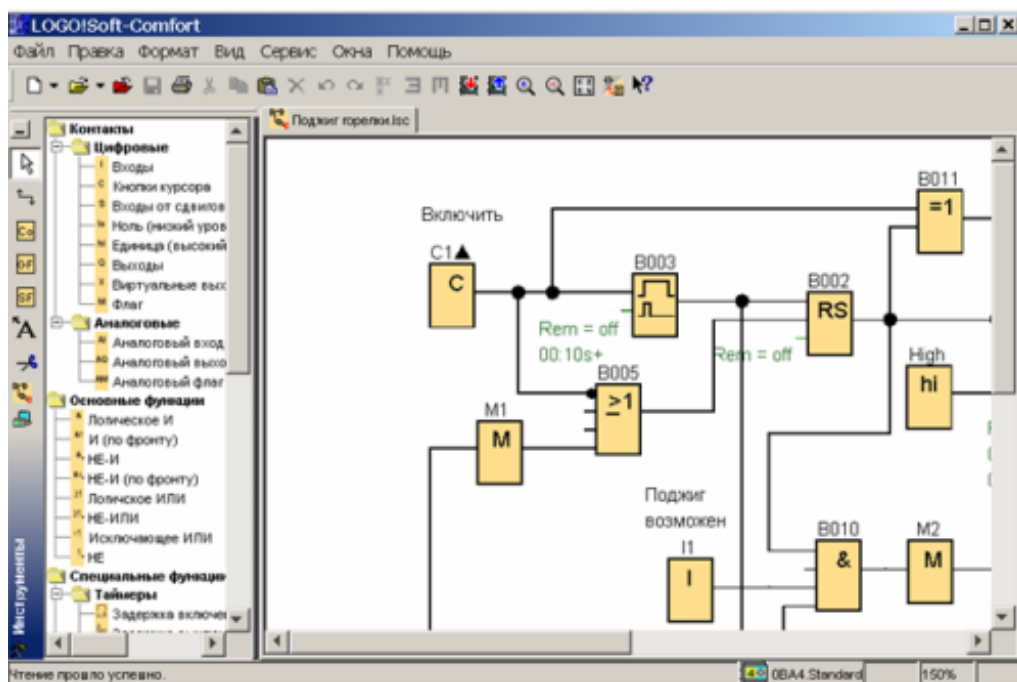


Рис.5. Основное окно программы LOGO!Soft Comfort в режиме функциональной блок-схемы (редактор FBD)

## Пример. Модель электрической сети в LOGO!Soft Comfort

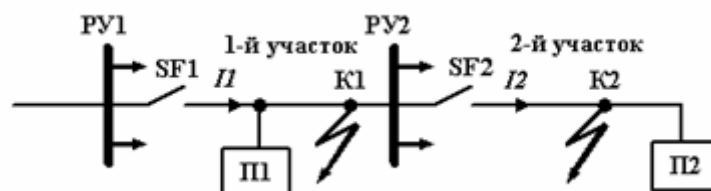


Рис.6. Конфигурация защищаемой сети PY1, PY2 – распределительные устройства; П1, П2 – первая и вторая группа потребителей; SF1, SF2 – первый и второй автоматические выключатели; K1, K2 первая и вторая точки короткого замыкания; I1, I2 – токи в участках сети



От распределительного устройства РУ1 отходит несколько электрических линий, одна из которых защищается выключателем SF1. От этой линии питается распределительное устройство РУ2, одна из отходящих линий которого защищается выключателем SF2. Короткое замыкание может произойти на участке 1 (точка K1) или на участке 2 (точка K2), при этом короткое замыкание (к.з.) должно отключаться ближайшим к точке к.з. выключателем. Однако, если ближайший выключатель неисправен, то к.з. должно отключиться выключателем, расположенным ближе к источнику питания. Модель электрической сети в LOGO! Soft Comfort приведена на рис.7. расположенным ближе к источнику питания.

Автоматический выключатель SF1 моделируется кнопкой C1 и блоками B001,...B006 и Q1. Кнопка C1 соответствует ручке включения/выключения автомата. Триггер B001 моделирует механическую защелку автомата, удерживающую контакты в замкнутом или разомкнутом состоянии.

Блок B002 имитирует «ломающийся рычаг», позволяющий отключить автомат при включенном положении ручки включения/выключения.

Инвертор B003 обеспечивает отключение автомата при выключенном положении ручки.

Блок B005 соответствует расцепителю, который через блок B004 отключает автомат при подаче «1» на его вход Trg. Расцепитель срабатывает с временной задержкой, которая состоит из нерегулируемой и регулируемой частей.

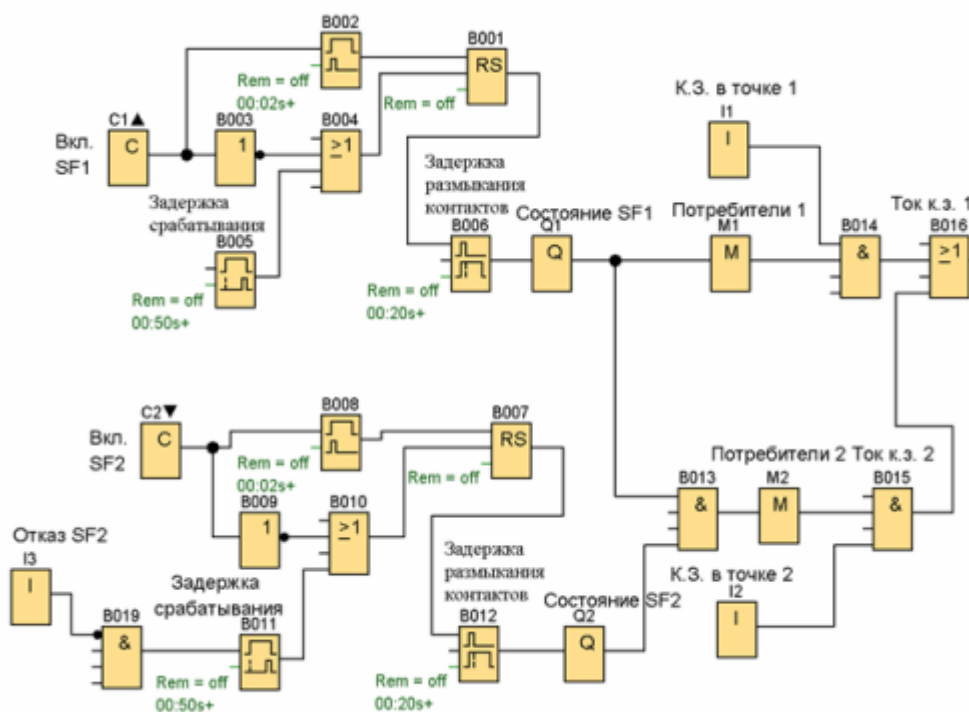


Рис. 7. Модель электрической сети в LOGO!Soft Comfort

Состояние контактов автомата SF1 определяется выходом Q1. Блок B006 моделирует время движения контактов до полного размыкания цепи.

Блок I1 моделирует к.з. в точке K1, блок M1 показывает наличие напряжения потребителей первой группы, блок B016 моделирует аварийный ток на первом участке.

Второй участок сети моделируется аналогично, но с помощью входа I3 моделируется отказ расцепителя автомата SF2.

## 2. Составление управляющей программы программируемого контроллера

Программируемые контроллеры предназначены для циклового программного управления станками и различным технологическим оборудованием, оснащенным датчиками и исполнительными устройствами, которые работают по двухпозиционному принципу: "включен-

выключен". В статье процесс составления управляющей программы рассмотрен на примере контроллера модели МКП-1.

В зависимости от исполнения этот контроллер позволяет управлять 16, 32 или 48 устройствами. Количество входных цепей для подключения датчиков соответствует числу выходов. Каждый вход и выход имеет свой адрес.

Контроллер обеспечивает управление исполнительными устройствами, прием информации от датчиков состояния оборудования, формирование выдержек времени, организацию условных и безусловных переходов по программе управления, а также реализует другие функции.

Проектирование управляющего устройства сводится к двум этапам: 1 – составление схемы подключения датчиков и исполнительных устройств к контроллеру, 2 – составление управляющей программы по схеме алгоритма.

### Подключение датчиков

Двухпозиционные кнопки и датчики подключаются к входным разъемам контроллера в соответствии с Таблицей 2. Каждому входу соответствует свой адрес.

Для питания входных цепей необходим источник питания с выходным напряжением  $U_n = 20 \dots 30$  В.

Срабатыванию датчика соответствует замыкание входной цепи (уровень двоичной 1), разомкнутое состояние цепи эквивалентно уровню двоичного 0.

Пример подключения контакта датчика к входу контроллера приведен на рис.8

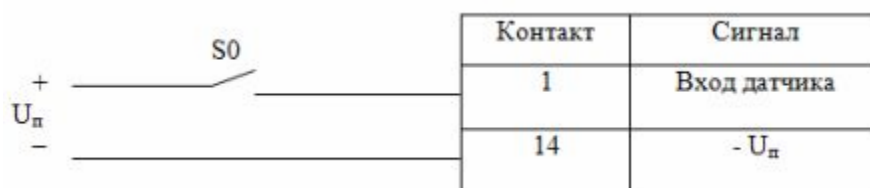


Рис.8. Схема подключения контакта датчика

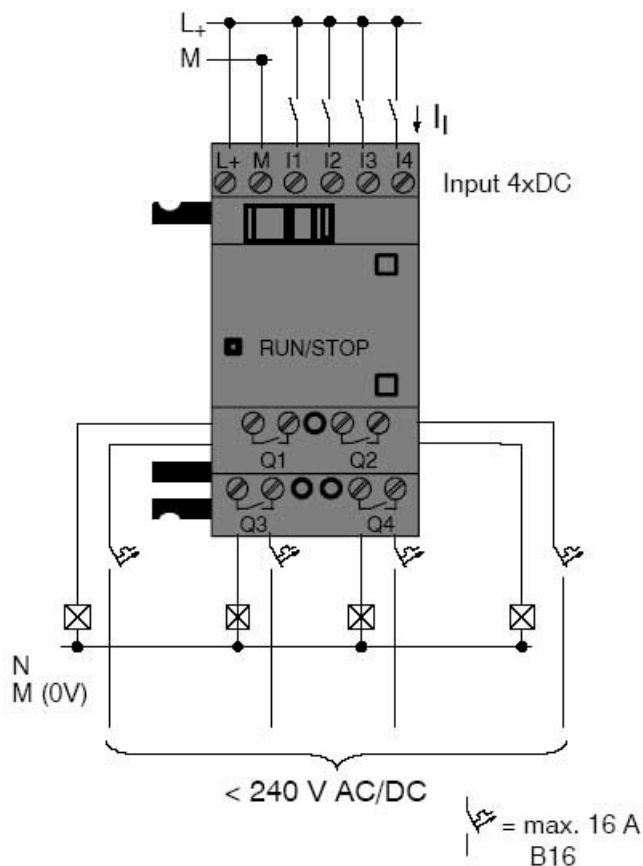


Рис.9. Схема подключения LOGO!

Таблица 2. Цепи входных сигналов контроллера

Адрес датчика	Контакт разъема		Адрес датчика	Контакт разъема	
	Датчик	- U <sub>н</sub>		Датчик	- U <sub>н</sub>
00	1	14	08	9	19
01	2	25	09	10	20
02	3	26	0A	11	28
03	4	15	0B	12	29
04	5	27	0C	13	30
05	6	16	0D	22	31
06	7	17	0E	23	21
07	8	18	0F	24	32

### Подключение исполнительных устройств

Исполнительные устройства (катушки реле, входные цепи бесконтактных аппаратов) подключаются к выходным разъемам контроллера в соответствии с таблицей 3.

Таблица 3. Цепи выходных сигналов контроллера

Адрес устройства	Контакт разъема			Адрес устройства	Контакт разъема		
	Устройство	- U <sub>н</sub>	+U <sub>н</sub>		Устройство	- U <sub>н</sub>	+ U <sub>н</sub>
00	24	17	30	08	13	18	25
01	7	17	30	09	8	18	25
02	2	14	29	0A	12	19	28
03	1	14	29	0B	9	19	28
04	11	20	26	0C	3	21	27
05	10	20	26	0D	23	21	27
06	5	16	31	0E	22	15	32
07	6	16	31	0F	4	15	32

Пример подключения термодатчика к LOGO! приведен на рисунке 10.

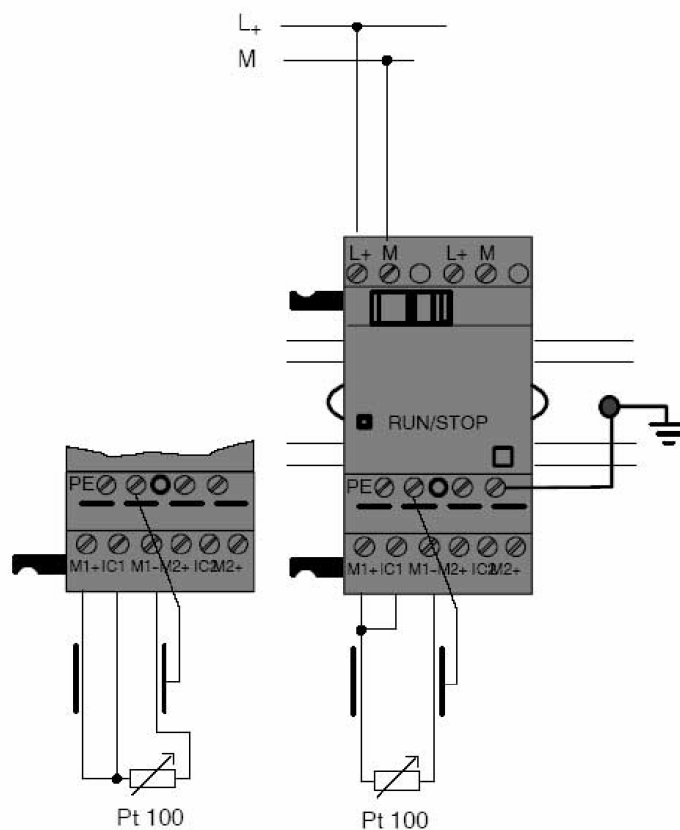


Рис.10. Схема подключения аналоговых цепей к LOGO!

Пример подключения катушек реле к выходам контроллера приведен на рис.11.



Рис.11. Схема подключения катушек реле

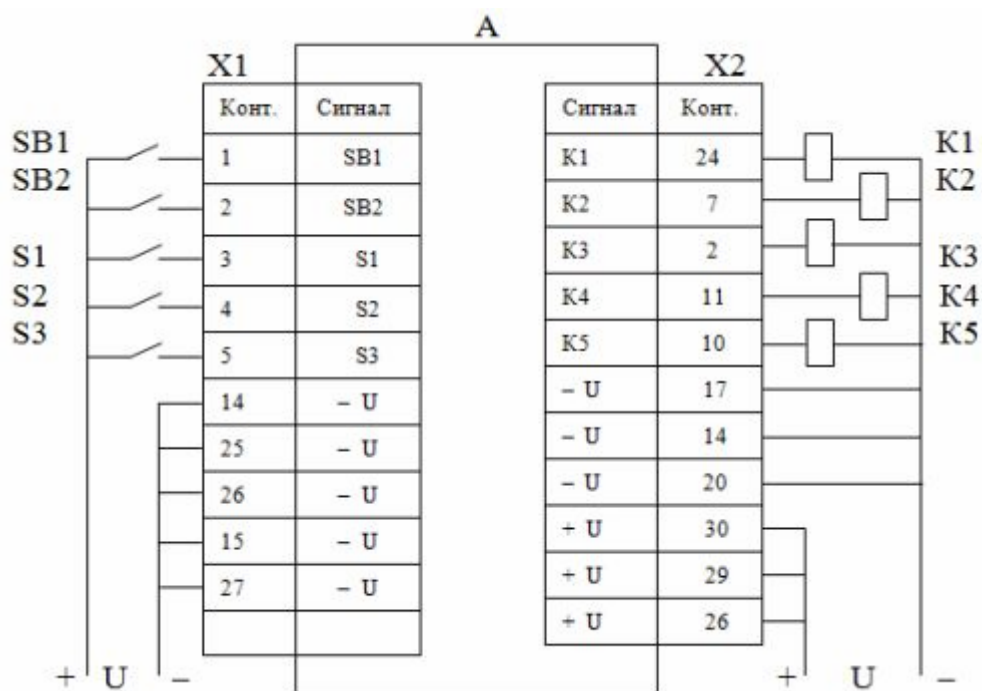


Рис.12. Пример полной схемы подключения внешних устройств к контроллеру

### Система счисления контроллера

Контроллер оперирует с цифрами, выраженными в шестнадцатеричной системе счисления. Основанием системы является десятичное число 16, алфавит состоит из десяти цифр (0...9) и шести латинских букв (A, B, C, D, E, F). Буквы соответствуют десятичным числам 10, 11, 12, 13, 14, 15.

Подробнее о шестнадцатеричной системе счисления: [Системы счисления чисел](#)

При программировании все числовые величины задаются шестнадцатеричными значениями. В таблице 4 приведен ряд чисел в шестнадцатеричной системе N<sub>16</sub> и их десятичные эквиваленты N<sub>10</sub>.

Таблица 4. Числа в шестнадцатеричной системе счисления

N <sub>10</sub>	N <sub>16</sub>	N <sub>10</sub>	N <sub>16</sub>	N <sub>10</sub>	N <sub>16</sub>	N <sub>10</sub>	N <sub>16</sub>	N <sub>10</sub>	N <sub>16</sub>
0...9	0...9	16	10	24	18	32	20	40	28
10	A	17	11	25	19	33	21	41	29
11	B	18	12	26	1A	34	22	42	2A
12	C	19	13	27	1B	35	23	43	2B
13	D	20	14	28	1C	36	24	44	2C
14	E	21	15	29	1D	37	25	45	2D
15	F	22	16	30	1E	38	26	46	2E
		23	17	31	1F	39	27	47	2F

### Система команд контроллера

Программируемый контроллер оснащен системой команд, предназначенной для решения задач программного управления. В таблице 5 приведены небольшая часть команд контроллера. Команда состоит из двух частей: кода выполняемой операции (КОП) и операнда, в котором указывается адрес объекта, над которым производится операция. При этом в качестве такого объекта выступают как датчики и исполнительные устройства, так и команды самой программы. При задании временных интервалов операндом является длительность этих интервалов.

Таблица 5. Система команд контроллера

Команда		Выполняемые действия
Код операции (КОП)	Операнд	
02	Адрес входа (номер датчика)	Ожидание появления сигнала датчика. Выполнение следующей команды происходит только после появления этого сигнала.
04	Адрес входа (номер датчика)	Проверка входа на наличие сигнала датчика. Если сигнал есть, то в ОЗУ контроллера формируется бит условия $БУ = 1$ . Если сигнала нет, то устанавливается $БУ = 0$ .
05	Адрес выхода (номер исполнительного устр-ва)	Включение исполнительного устройства под заданным номером.
06	Адрес выхода (номер исполнительного устр-ва)	Выключение исполнительного устройства под заданным номером
07	Временной интервал T	Задание выдержки времени $\Delta t$ . Длительность временного интервала кратно 0,1 сек, минимально возможному для задания отрезку времени ( $\Delta t = T \cdot 0,1$ ).
09	Адрес команды	Безусловный переход к выполнению указанной команды.
0A	Адрес команды	Выполнение заданной команды, если бит условия $БУ = 1$ . Если $БУ = 0$ , то выполняется следующая, записанная в программе после 0A команда.

### Составление схемы алгоритма

Порядок работы любого устройства может быть описан с помощью графических символов, образующих схему алгоритма. При составлении схемы могут быть использованы символы четырех типов, называемые вершинами (рис.13).

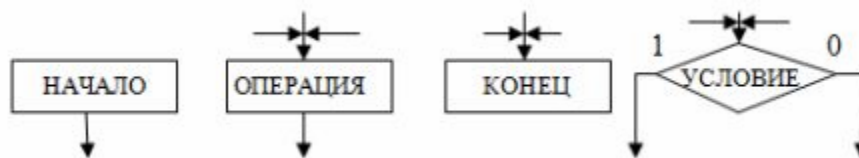


Рис.13. Вершины схемы алгоритма

Вершина «Начало» соответствует исходному состоянию устройства управления до воздействия на него органами управления, например, кнопкой «Пуск».

Вершина «Операция» соответствует выполнению определенной элементарной операции над приборами, входящими в состав устройства управления, например включение или выключение реле.

Вершина «Конец» соответствует окончанию процесса управления, например, после нажатия кнопки «Стоп».

Операционная вершина соответствует выполнению определенной элементарной операции над приборами, входящими в состав устройства управления, например включение или выключение реле. На схеме внутри значка вершины записывается выполняемая операция.

Условная вершина определяет условие перехода от одной операционной вершины к другой. Условие задается датчиком, кнопкой управления или другим устройством. Состояние датчика или кнопки и соответственно выходы вершины обозначаются цифрами 1 или 0. Например: путевой выключатель «включен» – 1; «выключен» – 0. Составление схемы алгоритма сводится к соединению вершин в соответствии с требуемым порядком работы автоматизируемого устройства. Фрагмент схемы алгоритма показан на рис.14. На схеме символом X1 обозначен переключатель,  $\Delta t$  – временной интервал.

### Составление управляющей программы

Каждая команда в программе записывается под своим порядковым номером, который является ее адресом. Составление программы выполняется по схеме алгоритма и должна содержать набор команд, реализующих все операции, указанные в схеме.

Перед разработкой программы необходимо составить схему подключения датчиков и исполнительных устройств. В зависимости от места подключения этих устройств они получают свой номер, который в программе является их адресом.

Составление программы следует начинать с вершины схемы "Начало" и далее последовательно программировать операции вплоть до вершины "Конец".

Если выполнение какой-либо операции производится после срабатывания кнопки, путевого выключателя или другого датчика, то задается команда 02, а в качестве операнда записывается номер данного датчика. При этом контроллер будет выполнять команду включения или выключения исполнительных аппаратов только после получения сигнала о срабатывании этого датчика.

Включение или выключение аппаратов выполняется командами соответственно 05 или 06. В операнде записывается номер коммутируемого аппарата

Задание временных интервалов осуществляется командой 07. В операнде записывается коэффициент, который при умножении на 0,1 сек. дает требуемое время задержки.

Например, при задании  $t = 2,6$  сек. в операнде записывается число 1A (26 в десятичной системе счисления). Максимальное значение выдержки времени, задаваемое одной командой 07, равно 25,5 сек (команда 07 FF). Если необходимо получить временную задержку большую, чем 25,5 секунд, то в управляющей программе следует последовательно включить несколько команд 07, суммарно обеспечивающих требуемый временной интервал.

Для реализации в программе условных переходов (в схеме алгоритма условная вершина с операциями как по «1», так и по «0») следует сначала задать команду проверки этой вершины 04.

Если датчик, соответствующий этой вершине, находится в состоянии «1», то сформируется бит условия  $БУ = 1$ . Если датчик в состоянии «0», то сформируется  $БУ = 0$ .

Следующей задается команда 0A, которая, если в предыдущей команде установлен  $БУ = 1$ , переведет контроллер на выполнение команды, указанной в операнде этой команды.

При  $БУ = 0$  контроллер будет выполнять команду, идущую следом за командой 0A.

При составлении программы рекомендуется сначала записать последовательность команд, которую будет выполнять контроллер при  $БУ = 0$ , не указывая операнда в команде 0A. После записи всех команд, выполняемых контроллером по условию «0», в программу заносится команда, выполняемая по условию «1». Адрес этой команды указывается в операнде команды 0A.

ПРИМЕЧАНИЕ: Для бита условия исходным состоянием является  $БУ = 1$ , которое устанавливается после включения питания контроллера и после исполнения команд условных переходов.

Пример записи программы для фрагмента схемы алгоритма по рис.14 показан в таблице 6.

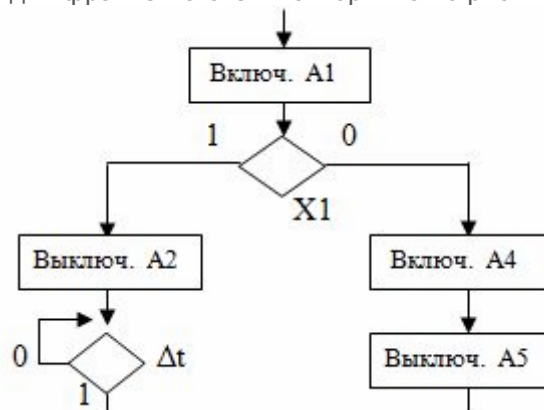


Рис.14. Фрагмент схемы алгоритма

Адрес команды	Код операции	Операнд	Содержание операции
06	05	01	Включение A1 (адрес 01)
07	04	00	Проверка X1 (адрес 00)
08	0A	0E	Переход к команде 0E при X = 1
09	05	04	Включение A4 (адрес 04)
0A	06	05	Выключение A5 (адрес 05)
0E	06	02	Выключение A2 (адрес 02)
0F	07	1A	Выдержка времени $\Delta t = 2,6$ сек
----	----	----	-----

Таблица 6. Фрагмент управляющей программы

### 3. Применение микропроцессорных систем в электротехнике на примере использования ПЛК

Говорить о применении [микропроцессорных систем](#), это значит говорить практически обо всех технических устройствах, которые нас окружают. В любой области электротехники: в электроснабжении, электроприводе, электроосвещении применяются от простейших схем под управлением 8-разрядных микроконтроллеров до сложнейших микропроцессорных систем с многоуровневым сетевым управлением.

Обратим внимание на [программируемые контроллеры \(ПЛК\)](#) (их также называют программируемые реле) LOGO! Siemens, предназначенные для построения наиболее простых устройств автоматического управления. Почему LOGO! Siemens? Потому что для работы с ним не требуется специальных знаний по микропроцессорной технике и языкам программирования, а достаточно [основ электротехники](#) и цифровой электроники (тоже основ). Кроме того, программные продукты компании Siemens распространяются в свободном доступе.

На рисунке 15 показан внешний вид контроллера LOGO! Basic и модуля расширения. Алгоритм функционирования модулей задаётся программой, составленной из набора встроенных функций - FBD (Function Block Diagram) - графический язык программирования. Программирование модулей может производиться либо с компьютера, оснащенного программой LOGO Soft Comfort, либо установкой запрограммированного модуля памяти, либо с их клавиатуры (при её наличии) без использования дополнительного программного обеспечения.



Рис.15 – Конструкция контроллера LOGO! Basic и модуля расширения

Стоимость контроллера и модулей расширения не высока, что даёт возможность применять их даже для автоматизации и не сложных процессов. Возьмем пример, предложенный самой компанией Siemens, смесительное устройство. На рисунке 16 представлена блок-схема смесительного устройства.

Формулировка задания:

По команде запуска (SB1) открыть вентиль Y1 и заполнить бак до отметки SL2. Закрыть вентиль Y1, открыть вентиль Y2 и заполнить бак до отметки SL1. Закрыть вентиль Y2 и на 15 минут включить смеситель. Открыть вентиль Y3 и слить полученную смесь. По сигналу датчика SL3 закрыть вентиль Y3 и привести схему в исходное состояние.

**Исполнительные устройства:**

- M – электродвигатель смесителя
- Y1 – вентиль подачи компонента 1
- Y2 – вентиль подачи компонента 2
- Y3 – вентиль выпуска готовой смеси

**Датчики и органы ручного управления:**

- SL1 – датчик полного заполнения бака
- SL2 – датчик заполнения бака компонентом 1
- SL3 – датчик опустошения бака
- SB1 – кнопка запуска установки

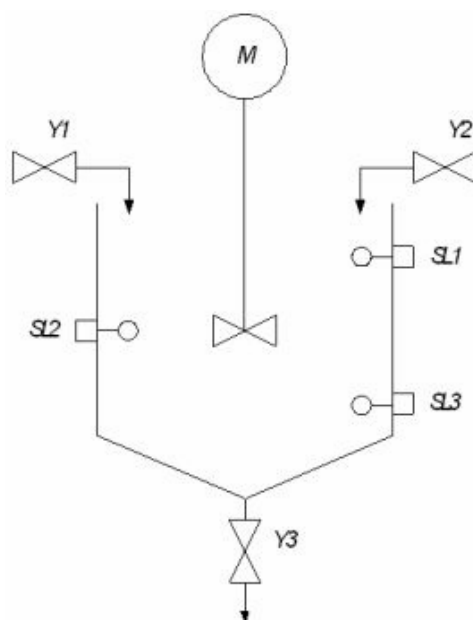


Рис. 16 – Блок-схема смесительного устройства

Исходя из технического задания, составим классическую релейно-контакторную схему (рис.17). Традиционно устанавливаем кнопку «Стоп» SB1, поэтому кнопка запуска установки становится SB2.



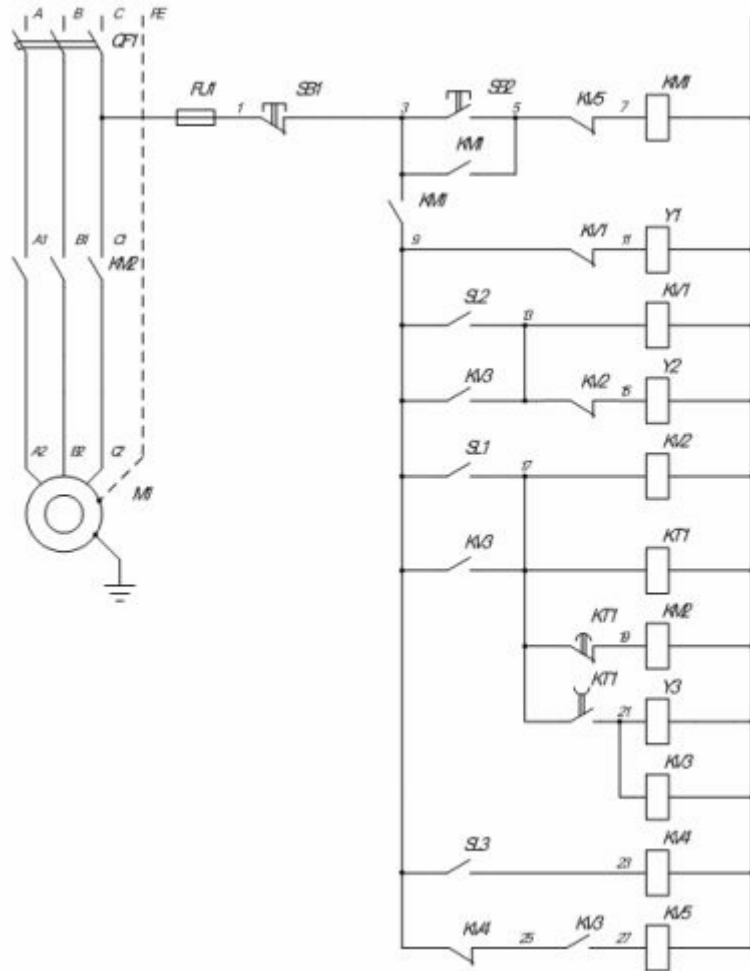


Рис.17 – Релейно-контакторная схема смесительного устройства.

Эта же схема, выполненная на ПЛК LOGO! (рисунок 18). Однозначно проще, а ведь использована только малая часть возможностей контроллера. Кроме самого контроллера в схеме из элементов присутствуют только датчики, органы управления и исполнительные механизмы. Значит схема намного надёжнее своего классического аналога.

Маркировка контроллера LOGO!230RC указывает: питающее напряжение - 115-240 В постоянного или переменного тока, выходы релейного типа (ток нагрузки - 3 А для индуктивной нагрузки).

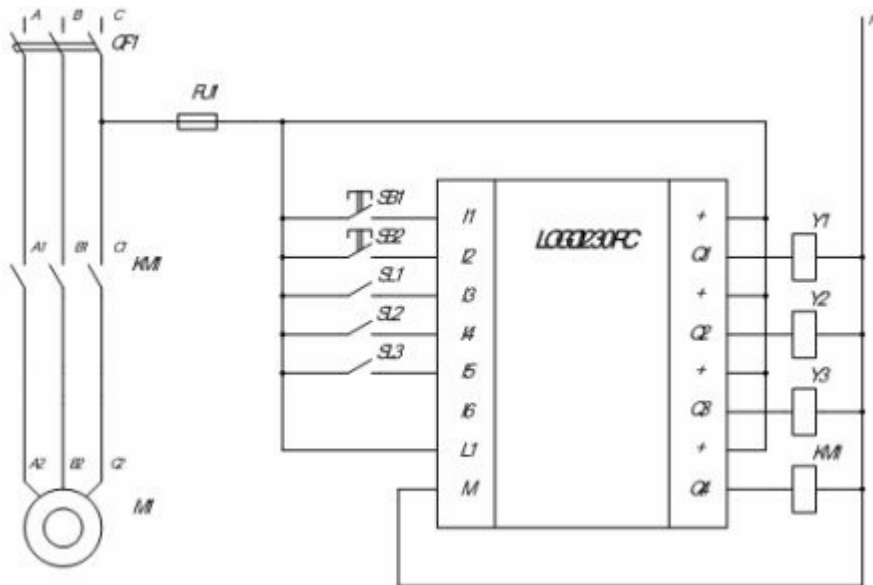


Рис.18 – Схема смесительного устройства на контроллере LOGO!

Для программирования ПЛК LOGO! необходимо создание коммутационной программы. Коммутационная программа создается с помощью пакета LOGO!Soft Comfort - программное средство для программирования LOGO!, которое используется для простого и быстрого создания, тестирования, изменения, сохранения и распечатки коммутационных программ.

У LOGO! есть входы и выходы. Входы обозначаются буквой I и номером. Выходы обозначаются буквой Q и номером.

Цифровые входы и выходы могут принимать состояние «0» или «1». «0» означает, что на входе нет напряжения; «1» означает, что оно есть.

Блок в LOGO! – это функция, которая преобразует входную информацию в выходную информацию.

На рисунке 19 представлен вариант коммутационной схемы контроллера смесительного устройства, созданный в среде LOGO!Soft Comfort. При создании коммутационной программы связываем соединительные элементы с блоками. Простейшими блоками являются [логические операции](#). Также в схеме использованы триггеры и блок задержки выключения.

Коммутационная программа отражает алгоритм (логику) работы схемы управления. Графически реализованная схема на типовых блоках и соединительных элементах в дальнейшем преобразуется в логическую структуру контроллера.

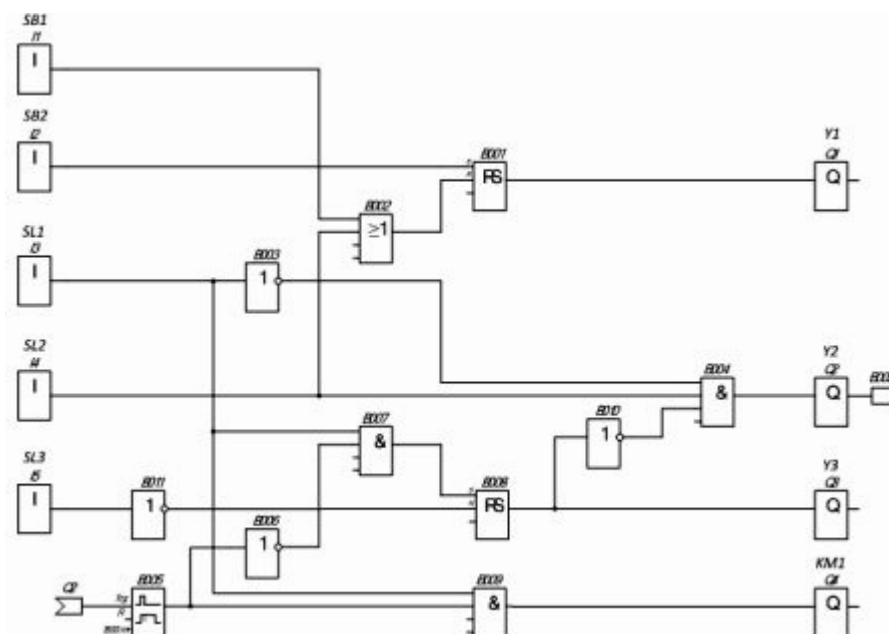


Рис.19 – Коммутационная схема смесительного устройства на контроллере LOGO!

## 4. Автоматизация системы вентиляции на ПЛК LOGO!

Программируемые логические контроллеры (ПЛК) получили широкое применение для автоматизации технологического оборудования и процессов в промышленном производстве. Это связано с их преимуществами по сравнению с устройствами электроавтоматики, выполненными по принципу жесткой логики. В частности, ПЛК позволяют хранить в памяти несколько программ и использовать их по мере необходимости.

Наиболее простыми являются моноблочные контроллеры. Ярким примером данной группы являются моноблочные ПЛК серии LOGO! фирмы Siemens. Они позволяют легко и удобно организовать дискретное управление различными технологическими процессами.



Рис.20. ПЛК LOGO! 230RC

Программным пакет LOGO! Soft Comfort позволяет легко и эффективно создавать программы управления, а затем передавать их в контроллер. Одним особенно удобным для пользователя свойством является возможность имитации программы, что позволяет отредактировать схему управления надлежащим образом перед ее вводом в эксплуатацию. Кроме того, программное обеспечение позволяет обширно документировать созданные программы.

Пример программирования ПЛК LOGO!:

Программирование ПЛК LOGO! с помощью программы LOGO! Soft Comfort:

### Требования к системе вентиляции

Назначение вентиляционной системы состоит в том, чтобы подавать свежий воздух в помещение и вытягивать застоявшийся воздух из помещения. Рассмотрим следующий пример:



Рис.21. План расположения оборудования

В помещении имеется вытяжной вентилятор и приточный вентилятор свежего воздуха. Оба вентилятора контролируются датчиком потока. В помещении никогда не должно возникать избыточное давление.

Приточный вентилятор должен включаться только при условии, что датчик потока сигнализирует о надежной работе вытяжного вентилятора. Если один из вентиляторов выходит из строя, то загорается предупреждающая лампа. Коммутационная схема для предыдущего решения имеет следующий вид:

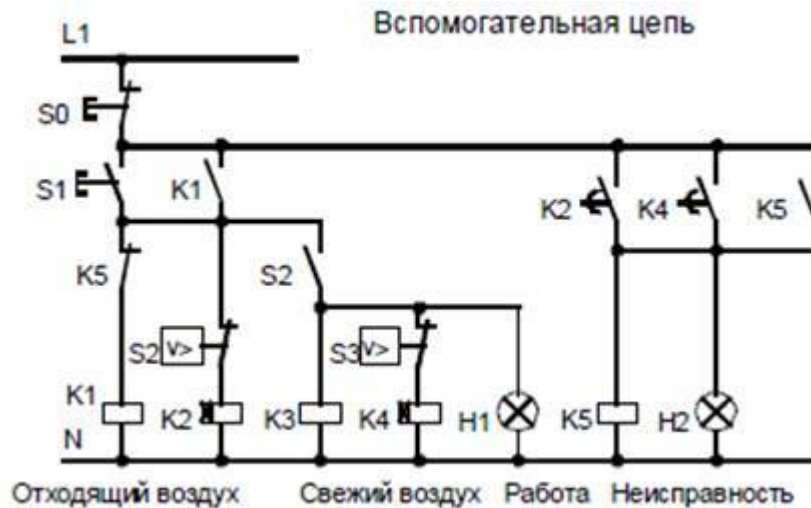


Рис.22 – Релейно-контакторная схема вытяжной системы

Вентиляторы контролируются датчиками потока. Если после короткой задержки воздушный поток не регистрируется, то система выключается и выдается сообщение о неисправности, которое квитируется нажатием кнопки выключения. Контроль вентиляторов требует кроме датчиков потока наличия анализирующей схемы с несколькими переключающими устройствами. Анализирующая схема может быть заменена одним модулем LOGO!.

#### Подключение вентиляционной системы с LOGO! 230RC

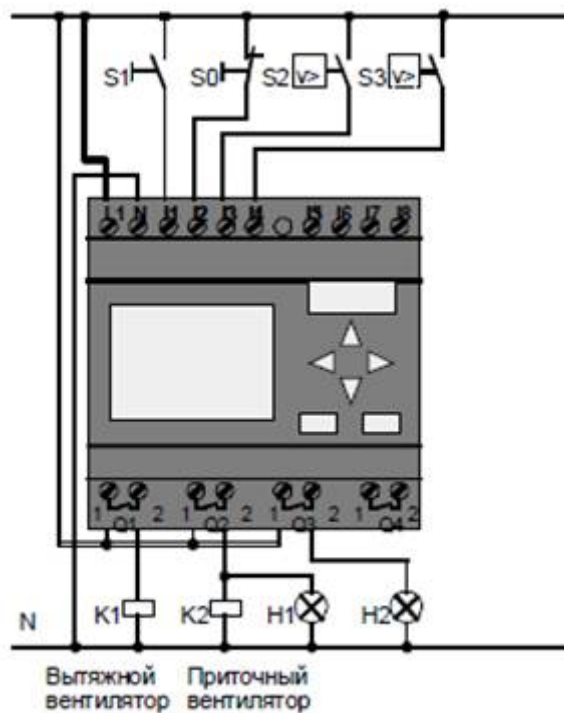


Рис.23 – Коммутационная схема системы вентиляции на контроллере LOGO! 230RC.

Используемые компоненты:

- K1 - Линейный контактор
- K2 - Линейный контактор
- S0 (НЗ контакт) - Кнопка останова
- S1 (НО контакт) - Кнопка пуска
- S2 (НО контакт) - Датчик потока

- S3 (НО контакт) - Датчик потока
- Н1 - Индикаторная лампа
- Н2 - Индикаторная лампа

### Блок-схема решения с LOGO!

Блок-схема системы управления вентиляцией с помощью LOGO! имеет следующий вид:

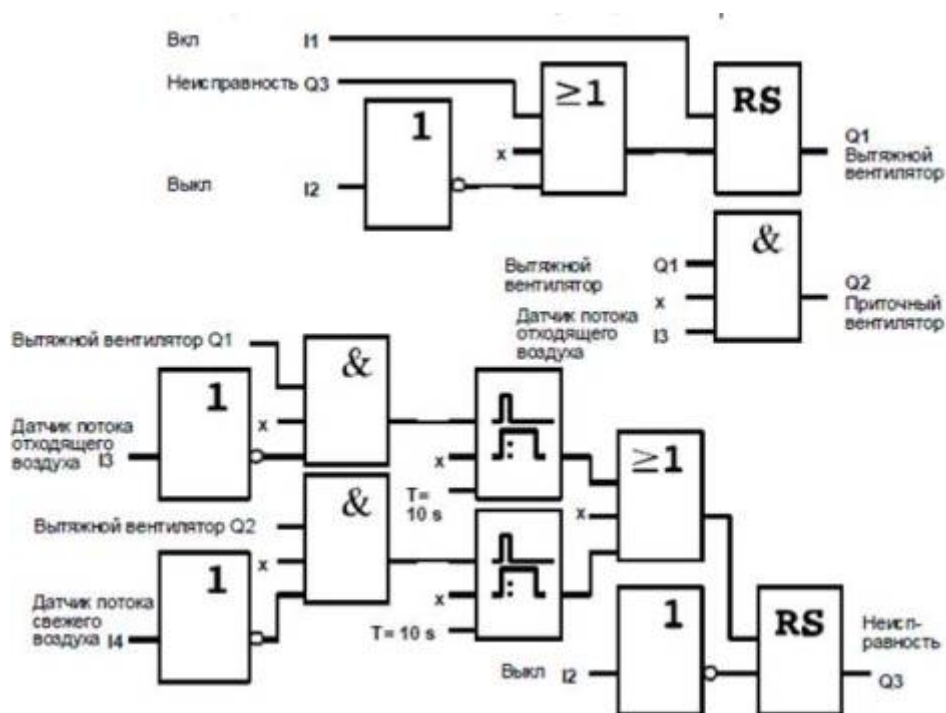


Рис.24 – Блок-схема системы управления вентиляцией на контроллере LOGO!

### Преимущества использования LOGO!

При использовании модуля LOGO! вам нужно меньше переключающих устройств. Это экономит время на монтаж и пространство в шкафу управления. При определенных обстоятельствах это, возможно, даже позволит вам использовать шкаф управления меньших размеров.

### Дополнительные возможности при использовании LOGO!

Свободный выход (Q4) может быть использован как потенциально развязанный контакт для сигнализации о неисправности или сбое по питанию. Возможно ступенчатое отключение вентиляторов.

Эти функции могут быть реализованы без дополнительной коммутационной аппаратуры.

### Функциональная блок-схема расширенного решения с использованием LOGO!

Вентиляторы Q1 и Q2 включаются и выключаются, как показано с помощью следующей схемы:

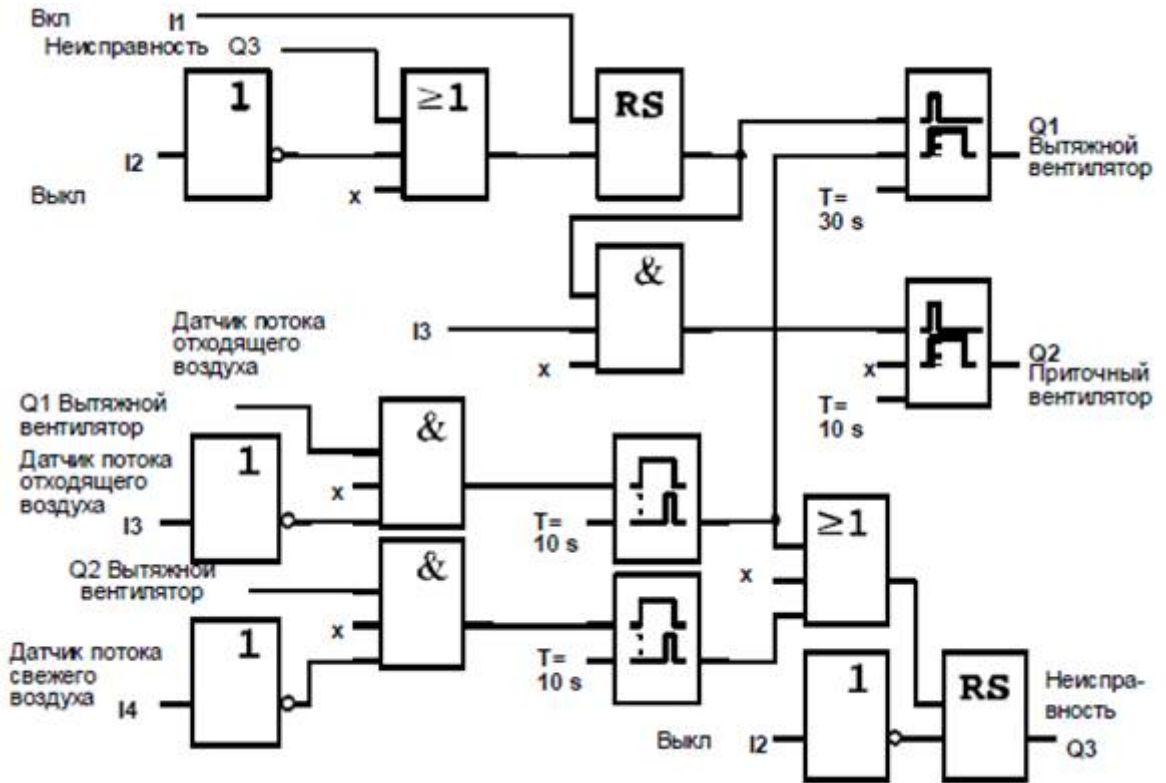


Рис.25. Функциональная блок-схема расширенного решения с использованием LOGO!

Кроме того, через выход Q4 можно сгенерировать сообщение:

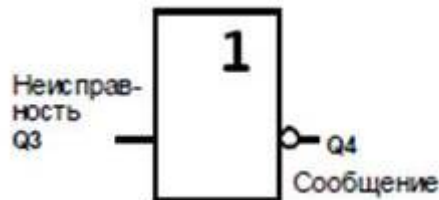


Рис.26. Сообщение о неисправности вентустановки

Релейные контакты выхода Q4 при работе системы всегда замкнуты. Реле Q4 не отпускает, пока не происходит потеря питания или неисправность в системе. Это контакт может быть использован, например, для дистанционной индикации.

## ЛИТЕРАТУРА

1. Минаев И.Г., Самойленко В.В. Программируемые логические контроллеры, руководство/-Ставрополь: изд-во «Агрус»,2009,-100 с.
2. Парр Э. Программируемые контроллеры, руководство для инженера.- М: Бином, 2007, - 516 с.
3. Петров И.В. Программируемые контроллеры. Стандартные языки Программирования. – М: СЛОН-Пресс, 2004. – 246 с.
4. Siemens, LOGO! Руководство пользователя A5E01248543-01.- Siemens: LOGO! Manual, 2013. – 290 с.