

# CPM1/CPM1A

Программируемые контроллеры

## Руководство по установке

*Внимание!*

Данное руководство не является официальной документацией OMRON.

В случае возникновения спорных вопросов, связанных с применением оборудования OMRON, обращайтесь к официальной документации.

# **Программируемые контроллеры СРМ1/СРМ1А**

**Руководство по работе**

*Декабрь 1995*



## *Перед тем, как читать инструкцию:*

Продукция фирмы OMRON создана для использования согласно разрешенным процедурам квалифицированным оператором и только для целей, описанных в данной инструкции.

В данной инструкции приняты следующие обозначения для индикации и классификации предупреждающих сообщений. Обязательно учитывайте эту информацию. Если пренебречь предупреждениями, это может повлечь несчастные случаи с людьми или повреждение оборудования.

**Опасность!** Указывает на информацию, пренебрежение которой с большой долей вероятности повлечет смерть или тяжелые увечья.

**Предупреждение!** Указывает на информацию, пренебрежение которой может повлечь смерть или тяжелые увечья (с меньшей степенью вероятности).

**Внимание!** Указывает на информацию, пренебрежение которой может повлечь относительно серьезные или небольшие травмы, повреждение оборудования или неправильную работу.

## *Справочная информация об изделиях OMRON*

Все изделия OMRON пишутся в данной инструкции с прописной буквы. Слово “Блок” также пишется с большой буквы, когда оно относится к продукции OMRON, независимо от того, появляется оно или нет в собственном имени изделия.

Сокращение “Ch”, которое появляется в некоторых режимах индикации и некоторых продуктах OMRON, часто обозначает “слово” (word) и в документации имеет сокращенное обозначение “Wd”.

Сокращения “ПК” (PC) означают Программируемый Контроллер (Programmable Controller) и в других смыслах не используется.

## *Средства выделения информации*

В левой колонке данной инструкции появляются следующие заголовки для облегчения определения типа информации.

**Замечание** Указывает информацию, заслуживающего особого интереса для эффективной и удобной работы изделий OMRON.

**1, 2, 3,...** Указывает на перечисления того или иного рода, такие, как процедуры, списки и т.д.

© OMRON, 1995

Все права сохранены. Никакую часть данного документа нельзя размножать, загружать в информационно-поисковые системы или передавать в любой форме, механической, электрической, фотокопированием, магнитозаписью или какой-либо другой, без письменного разрешения OMRON.

Патентной ответственности за использование информации в данном документе не несет. Более того, поскольку OMRON постоянно старается улучшить свою продукцию, информация в данной инструкции может измениться без уведомления. При подготовке данной инструкции были приняты все меры предосторожности. Однако OMRON не принимает на себя ответственности за ошибки или пропуски. Не принимается также никакая ответственность за ущерб, нанесенный вследствие применения информации, содержащейся в данном документе.



# Содержание

Введение . . . . .	11
1.1 Характеристики и функции СРМ1 . . . . .	12
1.1.1 Характеристики СРМ1 . . . . .	12
1.1.2 Клеммник входа/выхода - распределение битов IR . . . . .	13
1.1.3 Функции СРМ1 . . . . .	14
1.2 Конфигурация системы . . . . .	17
1.2.1 Конфигурация ЦПУ и блока дополнительных входов/выходов. . . . .	17
1.2.2 Связь HOST LINK . . . . .	17
1.2.3 Линии связи 1:1 . . . . .	18
1.2.4 Связь NT LINK . . . . .	19
1.2.5 Линии связи периферийных устройств. . . . .	20
Глава 2. Технические характеристики блоков и их компоненты . . . . .	21
2.1 Характеристики. . . . .	22
2.1.1 Общие характеристики. . . . .	22
2.1.2 Характеристики. . . . .	22
2.1.3 Характеристики входов/выходов . . . . .	24
2.1.4 Характеристики адаптеров связи . . . . .	26
2.2 Компоненты блоков . . . . .	27
2.2.1 Компоненты ЦПУ. . . . .	27
2.2.2 Компоненты блока расширения . . . . .	29
2.2.3 Компоненты адаптеров связи. . . . .	30
Глава 3. Монтаж и подключение . . . . .	31
3.1 Меры предосторожности. . . . .	32
3.1.1 Подключение питания . . . . .	32
3.1.2 Цепи взаимоблокировки и аварийного ограничения. . . . .	32
3.1.3 Прерывания питания ПК. . . . .	32
3.2 Выбор места установки. . . . .	33
3.2.1 Условия места установки . . . . .	33
3.2.2 Установка на панели/в шкафу. . . . .	33
3.3 Монтаж СРМ1 . . . . .	34
3.3.1 Ориентация СРМ1 . . . . .	34
3.3.2 Монтаж СРМ1 . . . . .	34
3.3.3 Подключение блока расширения входов/выходов. . . . .	36
3.4 Подключение. . . . .	36
3.4.1 Общие меры предосторожности при подключении. . . . .	36
3.4.2 Подключение питания . . . . .	38
3.4.3 Подключение входов. . . . .	38
3.4.4 Подключение выходов . . . . .	41
3.4.5 Соответствие директивам EMC (электромагнитной совместимости) . . . . .	43
3.4.6 Подключение периферийного устройства . . . . .	44
3.4.7 Связь HOST LINK. . . . .	45
3.4.8 Связь ПК "1:1". . . . .	47
3.4.9 Связь NT LINK . . . . .	47
Глава 4. Использование периферийных устройств . . . . .	49
4.1 Использование программатора . . . . .	50
4.1.1 Совместимые программаторы . . . . .	50
4.1.2 Подключение программатора . . . . .	50
4.1.3 Переключение режимов СРМ1 . . . . .	51
4.2 Операции программатора . . . . .	52
4.2.1 Очистка памяти . . . . .	52
4.2.2 Чтение/очистка сообщений об ошибках . . . . .	53

4.3.3	Операции с зуммером . . . . .	54
4.2.4	Установка и чтение адреса памяти программ . . . . .	54
4.2.5	Поиск команды. . . . .	55
4.2.6	Поиск битового операнда . . . . .	55
4.2.7	Вставка и удаление команд . . . . .	56
4.2.8	Ввод или редактирование программ . . . . .	58
4.2.9	Контроль программы . . . . .	60
4.2.10	Просмотр бита, числа, слова . . . . .	61
4.2.11	Просмотр изменения состояния бита . . . . .	63
4.2.12	Просмотр параметра в двоичном виде . . . . .	64
4.2.13	Просмотр трех слов . . . . .	64
4.2.14	Просмотр в виде десятичного числа со знаком . . . . .	65
4.2.15	Просмотр в виде десятичного числа без знака . . . . .	65
4.2.16	Изменение значений трех соседних слов . . . . .	66
4.2.17	Изменение задания таймеров и счетчиков . . . . .	66
4.2.18	Изменение значения в виде 16-ричного или двоично-десятичного числа . . . . .	67
4.2.19	Изменение параметра, представляемого в виде двоичного числа . . . . .	68
4.2.20	Изменение десятичного числа со знаком . . . . .	68
4.2.21	Изменение десятичного числа без знака . . . . .	69
4.2.22	Принудительные установка/сброс бита . . . . .	70
4.2.23	Убрать принудительное состояние включено/выключено . . . . .	70
4.2.24	Изменение индикации с 16-ричного на ASCII . . . . .	71
4.2.25	Индикация времени цикла . . . . .	71
4.3	Пример программирования . . . . .	71
4.3.1	Подготовительные мероприятия . . . . .	71
4.3.2	Пример программы . . . . .	72
4.3.3	Процедуры программирования . . . . .	73
4.3.4	Контроль программы . . . . .	76
4.3.5	Тестовый прогон в режиме MONITOR . . . . .	77
Глава 5.	Тестовые прогоны и обработка ошибок . . . . .	79
5.1	Начальная проверка системы и процедура тестового прогона . . . . .	80
5.1.1	Начальная проверка системы . . . . .	80
5.1.2	Процедура тестового прогона СРМ1 . . . . .	80
5.1.3	Предосторожности при обращении с памятью FLASH . . . . .	80
5.2	Цикл СРМ1 . . . . .	81
5.3	Функции самодиагностики . . . . .	82
5.3.1	Нефатальные ошибки . . . . .	82
5.3.2	Фатальные ошибки . . . . .	82
5.3.3	Опознание ошибок . . . . .	83
5.3.4	Ошибки, определяемые пользователем . . . . .	83
5.4	Ошибки при работе с программатором . . . . .	84
5.5	Ошибки программирования . . . . .	85
5.6	Алгоритмы поиска неисправностей . . . . .	87
Приложение А	. . . . .	95
Стандартные модели	. . . . .	95
ЦПУ	. . . . .	95
Блок расширения входов/выходов	. . . . .	95
Адаптеры связи	. . . . .	95
Периферийные устройства	. . . . .	95
Приложение В	. . . . .	97
Габариты	. . . . .	97
Габариты с подключенными периферийными устройствами	. . . . .	99
Глоссарий	. . . . .	101



## *О данной инструкции:*

СРМ1 - это компактный, быстродействующий программируемый контроллер (ПК), предназначенный для операций управления в системах, требующих от 10 до 50 входов/выходов. Есть две инструкции, в которых описаны установка и работа СРМ1: Инструкция по работе (данная инструкция) и Инструкция по программированию. Кроме того, есть Инструкция по работе со специальными блоками СРМ1-серии.

В данной инструкции описаны конфигурация системы и установка СРМ1, даны краткие описания процедур работы на программаторе и представлены описания SSS (пакет поддержки SYSMAC). Для ознакомления с СРМ1 читайте сначала данную инструкцию.

В Инструкции по программированию (W228) дано детальное описание функций программирования СРМ1. Инструкция по работе с LSS и SSS описывает операции с LSS и SSS ПК серии С. При использовании LSS версия должна быть не ниже 5.0.

Перед тем, как начать установку и работу с СРМ1 внимательно ознакомьтесь с данной инструкцией.

Глава 1 содержит краткое описание действий по разработке системы СРМ1, возможные конфигурации системы и приводит специфические характеристики и функции.

Глава 2 описывает технические характеристики блоков, из которых составляется ПК, а также основные компоненты блоков.

Глава 3 описывает установку и подключение СРМ1.

Глава 4 описывает возможности SSS, подключение программатора и работу с программатором.

Глава 5 описывает, как осуществлять тестовый прогон, диагностику и устранение неисправностей аппаратной и программной части.

В Приложении А даны таблицы блоков СРМ1 и сопутствующей аппаратуры.

В Приложении В даны габариты блоков СРМ1.

***Внимание!*** Если приступить к работе с ПК, не ознакомившись с данной инструкцией, это может привести к несчастным случаям с персоналом, повреждению и поломке изделия. Пожалуйста, прочтите каждую главу перед тем, как проделать любую операцию.



## **Введение**

*В данной главе описаны характеристики и функции СРМ1 и возможные конфигурации системы.*

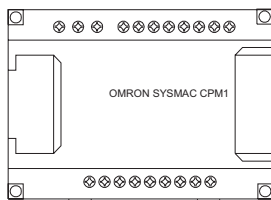
## 1.1 Характеристики и функции СРМ1

### 1.1.1 Характеристики СРМ1

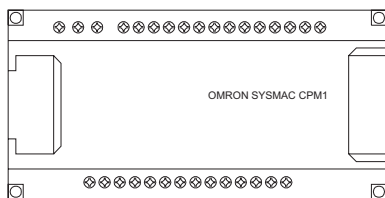
#### Конструкция в виде единого блока.

СРМ1 - это ПК, выполненный в виде единого блока с 10, 20 или 30 клеммами входов/выходов, встроенными в ЦПУ.

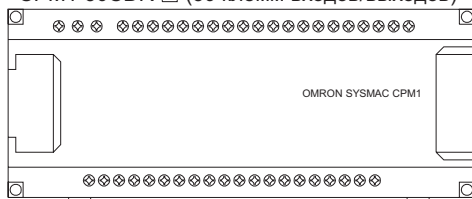
СРМ1-10CDR-□ (10 клемм входов/выходов)



СРМ1-20CDR-□ (20 клемм входов/выходов)



СРМ1-30CDR-□ (30 клемм входов/выходов)



#### Дополнительные входы/выходы

К ЦПУ можно подключить блок дополнительных входов/выходов (20 входов/выходов).

#### Функция входного фильтра

В СРМ1 имеется функция фильтра, служащая для предотвращения некорректной работы, вызванной дребезгом или помехами на входах. Пользователь может выбрать константу времени входа из ряда 1, 2, 4, 8, 16, 32, 64 или 128 мс.

#### Малый объем обслуживания

Память FLASH обеспечивает поддержание памяти без аккумулятора.

#### Соответствие директивам ЕС

СРМ1 отвечает требованиям ЕС (по EMC (электромагнитной совместимости) и низкому напряжению) и, следовательно, может встраиваться в оборудование, предназначенное для ЕС. О подробностях спрашивайте у Вашего представителя OMRON.

#### Входные прерывания

У СРМ1-10CDR-? есть 2 входа прерывания, у СРМ1-20CDR-? и СРМ1-30CDR-? - 4 входа прерывания. В дополнение к обычным входным прерываниям у СРМ1 есть режим счетчика, в котором подсчитываются высокоскоростные входные сигналы и через их кратное число запускается прерывание.

#### Входы быстрого реагирования

Входы быстрого реагирования могут обнаружить входные сигналы с шириной импульса 0.2 мс, независимо от места в цикле ПК. Входы быстрого реагирования и входы прерываний используют одни и те же клеммы.

**Интервальный таймер**

У СРМ1 есть высокоскоростной интервальный таймер, который можно настроить на диапазон 0.5 ... 319 968мс. Таймер можно настроить на запуск отдельного прерывания (режим одиночного прерывания) и постоянных прерываний по расписанию (режим прерываний по расписанию).

**Высокоскоростной счетчик**

У СРМ1 есть высокоскоростной счетчик, который можно использовать в инкрементальном режиме или режиме плюс/минус. Высокоскоростной счетчик можно объединить с входными прерываниями для осуществления управления с помощью сравнения (либо с заданной величиной, либо с зоной, причем на него не будет влиять длина цикла ПК).

**Функция аналогового задания**

На СРМ1 имеются два аналоговых регулятора, которые служат для ручного задания аналоговой величины.

**Связь HOST LINK**

СРМ1 совместим с системой связи HOST LINK, которая позволяет связываться с персональными компьютерами. СРМ1, использующий систему связи HOST LINK, может с помощью команд HOST LINK связываться с программируемым терминалом (ПТ).

Адаптер RS-232C используется для связи 1:1, а адаптер RS-422 используется для связи 1:n.

**Связь 1:1**

Можно создать линию данных с областью общих данных с другим СРМ1, СQM1 или С200HS. Для связи 1:1 служит адаптер RS-232C.

**Связь NT LINK**

Высокоскоростные операции можно осуществить, обеспечив прямой доступ путем подключения СРМ1 к программируемому терминалу с помощью интерфейса NT LINK. Для этого используется адаптер RS-232C.

**Стандартные периферийные устройства**

СРМ1 использует те же программатор, пакет поддержки SYSMAC (SSS) и пакет поддержки лестничных диаграмм (LSS), как и контроллеры типа MINI-H и СQM1.

**1.1.2 Клеммник входа/выхода - распределение битов IR**

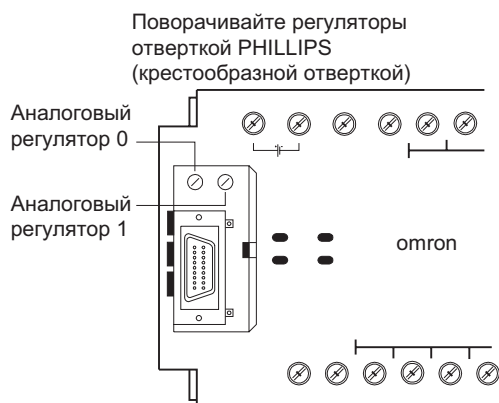
В следующей таблице показано, какие биты IR привязаны к клеммам входа/выхода ЦПУ и блока расширения входов/выходов.

	Количество клемм входа/выхода на ЦПУ				Питание	Номер модели
	Клеммы ЦПУ		Клеммы блока расширения			
	Входы	Выходы	Входы	Выходы		
10	6 точек: 00000 ... 00005	4 точки: 01000 ... 01003	12 точек: 00100 ... 00111	8 точек: 01100 ... 01107	Пер. ток	СРМ1- 10CDR-A
					Пост. ток	СРМ1- 10CDR-D
20	12 точек: 00000 ... 00011	8 точек: 01000 ... 01007	12 точек: 00100 ... 00111	8 точек: 01100 ... 01107	Пер. ток	СРМ1- 20CDR-A
					Пост. ток	СРМ1- 20CDR-D
30	18 точек: 00000 ... 00011 00100 ... 00105	12 точек: 01000 ... 01007 01100 ... 01103	12 точек: 00200 ... 00211	8 точек: 01200 ... 01207	Пер. ток	СРМ1- 30CDR-A
					Пост. ток	СРМ1- 30CDR-D

### 1.1.3 Функции СРМ1

#### Функция аналогового задания

На СРМ1 имеются два аналоговых регулятора, которые можно использовать для ручного задания таймера и счетчика. При повороте регуляторов содержимое соответствующего слова меняется в диапазоне 0 ... 200 (двоично-десятичное значение).



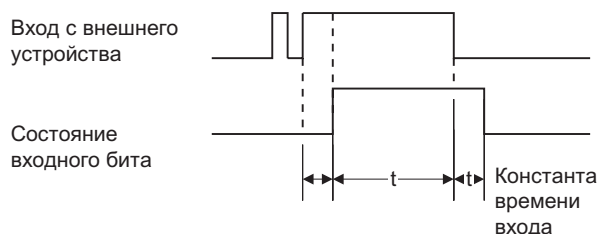
Поворачивайте регуляторы отверткой PHILIPS (крестовой отверткой).

В таблице показано, какие биты IR привязаны к клеммам входа/выхода ЦПУ и блока расширения.

Регулятор	Соответствующее слово IR	Диапазон (двоично-десятичный)
Аналоговый регулятор 0	IR 250	0000 ... 0200
Аналоговый регулятор 1	IR 251	

#### Функция входного фильтра

Константа времени входа для внешних входов СРМ1 может быть задана из ряда 1, 2, 4, 8, 16, 32, 64 или 128 мс. Увеличение задержки уменьшает воздействие дребезга или помех на входах.

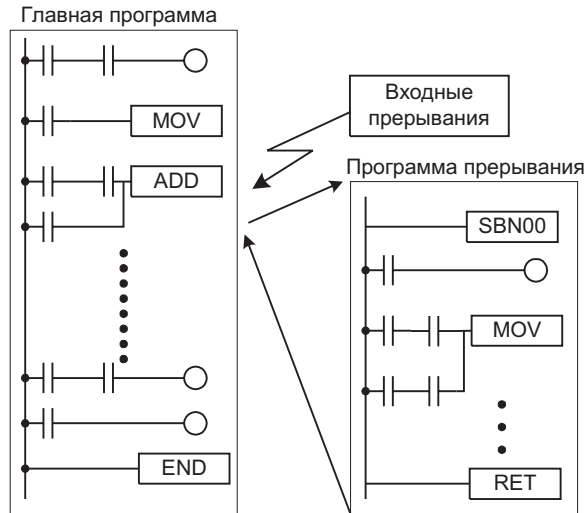


#### Входные прерывания

У СРМ1-10CDR-\_ есть 2 входа прерывания, у СРМ1-20CDR-\_ и СРМ1-30CDR-\_ - 4 входа прерывания. Есть два режима входных прерываний: мгновенное прерывание и режим счетчика.

- 1, 2, 3,... 1. Когда приходит сигнал входного прерывания в режиме мгновенного прерывания, главная программа прерывается и сразу же начинает выполняться программа прерывания, независимо от времени цикла.
2. В режиме счета сигналы внешнего прерывания подсчитываются с высокой скоростью (до 1 кГц) и прерывание вызывается каждый раз, когда счетчик достигает задания (SV). При этом главная программа прерывается и начинает выполняться программа прерывания. Задание может быть установлено в диапазоне 0 ... 65 535.

На диаграмме показано исполнение программы при прерывании.



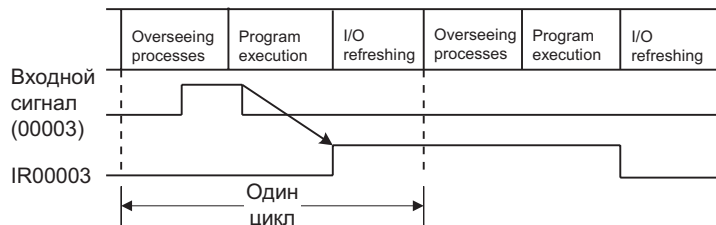
Модель ПК	Входные биты	Время реакции
СРМ1-10СДР-?	IR 00003 ... IR 00004	0.3 мс (1 кГц в режиме счета)
СРМ1-20СДР-?/ СРМ1-30СДР-?	IR 00003 ... IR 00006	

**Замечание** Если входы не используются в качестве входных прерываний, входные биты IR 00003 ... IR 00006 можно использовать как обычные входы.

#### Входы быстрого реагирования

У СРМ1-10СДР-\_ есть 2 клеммы для входов быстрого реагирования, у СРМ1-20СДР-\_ и СРМ1-30СДР-\_ - 4. (Одни и те же клеммы используются и как входы быстрого реагирования, и как входы прерываний).

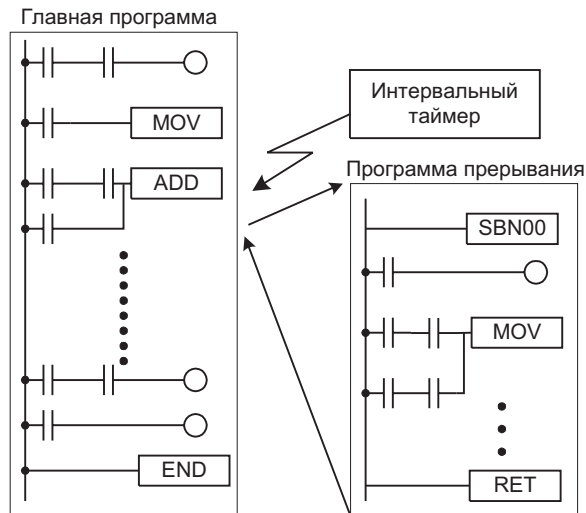
Входы быстрого реагирования имеют внутренний буфер, так что могут быть обнаружены сигналы короче, чем время цикла.



Модель ПК	Входные биты	Ширина мин. импульса
СРМ1-10СДР-?	IR 00003 ... IR 00004	0.2 мс
СРМ1-20СДР-?/ СРМ1-30СДР-?	IR 00003 ... IR 00006	

#### Функция интервального таймера (Прерывания по расписанию)

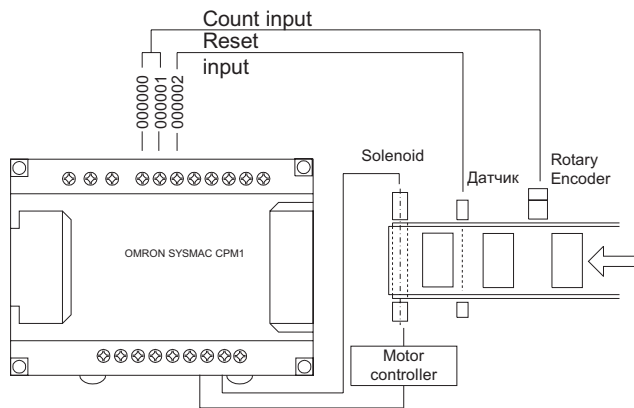
У СРМ1 есть высокоскоростной интервальный таймер, который можно настроить на диапазон 0.5 ... 319968 мс с дискретностью 0.1 мс. Таймер можно настроить на запуск отдельного прерывания (режим одиночного прерывания) и постоянных прерываний по расписанию (режим прерываний по расписанию).



Режим	Функция
Режим одиночного прерывания	Вызывает одно прерывание сразу после отсчета таймера.
Режим прерываний по расписанию	Вызывает прерывание каждый раз после отсчета таймера.

### Высокоскоростной счетчик

У СРМ1 есть высокоскоростной счетчик, который можно использовать в инкрементальном режиме или режиме плюс/минус. Высокоскоростной счетчик можно объединить с входными прерываниями для осуществления управления с помощью сравнения (либо с заданной величиной, либо с зоной), причем на него не будет влиять длина цикла ПЛК.



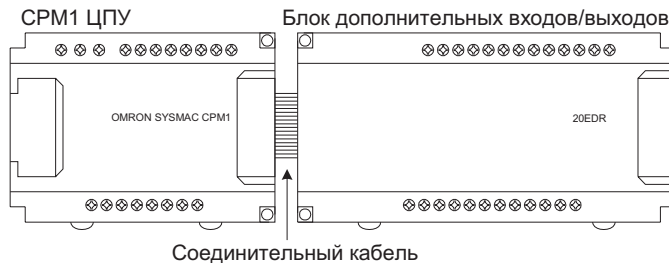
Режим	Инкрементальный	Плюс/минус
Функции входа	00000: вход фазы А 00001: вход фазы В 00002: вход фазы Z	00000: Счетный вход 00001: см. прим. 00002: вход сброса
Метод ввода	Разность фаз	Индивидуальные входы
Частота счета	2.5 кГц	5.0 кГц
Диапазон счета	-32 767 ... 32767	0 ... 65 535
Методы управления	Управление сравнением по конечному значению: Вводятся до 16 конечных значений и номеров подпрограмм прерываний.  Управление сравнением по зоне: Вводятся до 8 границ зон (верхняя и нижняя граница) и номеров подпрограмм прерываний.	



*Замечание* В инкрементальном режиме данный вход (00001) используется как обычный вход.

## 1.2 Конфигурация системы

### 1.2.1 Конфигурация ЦПУ и блока дополнительных входов/выходов



#### ЦПУ CPM1

В таблице приведены шесть типов ЦПУ. Все выходы релейные.

Число входов/выходов	Входы	Выходы	Питание	Номер модели
10	6	4	Пер. ток	CPM1-10CDR-A
			Пост. ток	CPM1-10CDR-D
20	12	8	Пер. ток	CPM1-20CDR-A
			Пост. ток	CPM1-20CDR-D
30	18	12	Пер. ток	CPM1-30CDR-A
			Пост. ток	CPM1-30CDR-D

#### Блок дополнительных входов/выходов

В таблице описан блок дополнительных входов/выходов. Все выходы релейные.

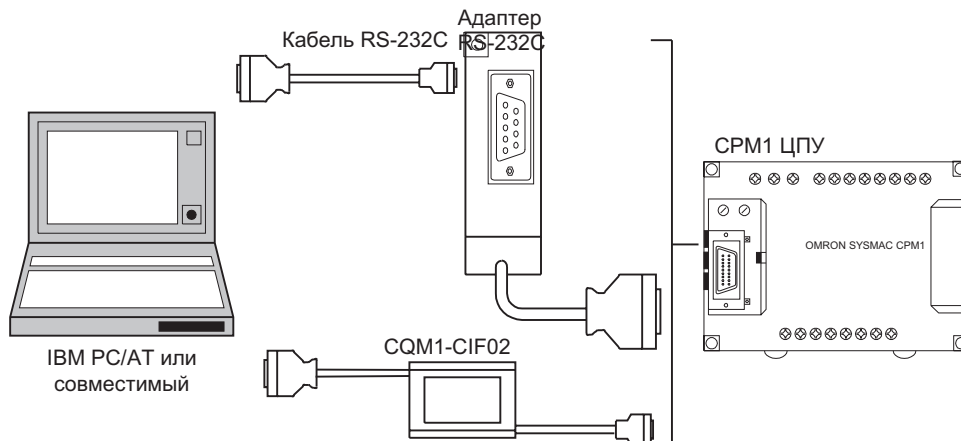
Число входов/выходов	Входы	Выходы	Номер модели
20	12	8	CPM1-20EDR

### 1.2.2 Связь HOST LINK

Связь HOST LINK позволяет управлять с управляющего компьютера до 32 ПК OMRON. Соединения компьютер - ПК можно осуществить такими адаптерами, как RS-232C и RS-422.

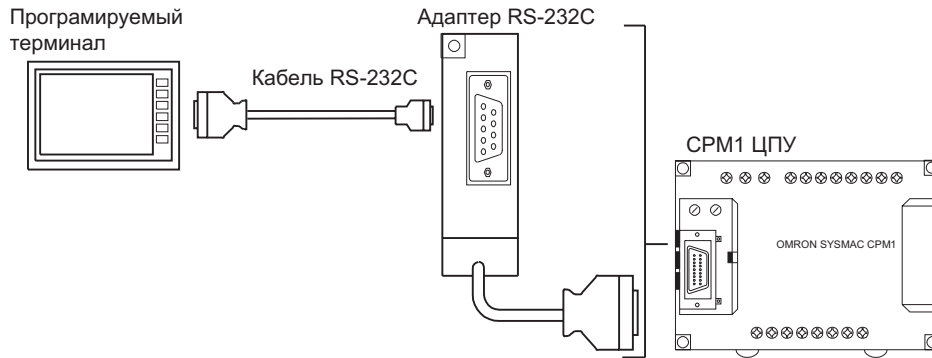
#### Связь 1:1

На следующей схеме показаны возможные методы связи 1:1 между CPM1 и IBM PC/AT или совместимым компьютером.



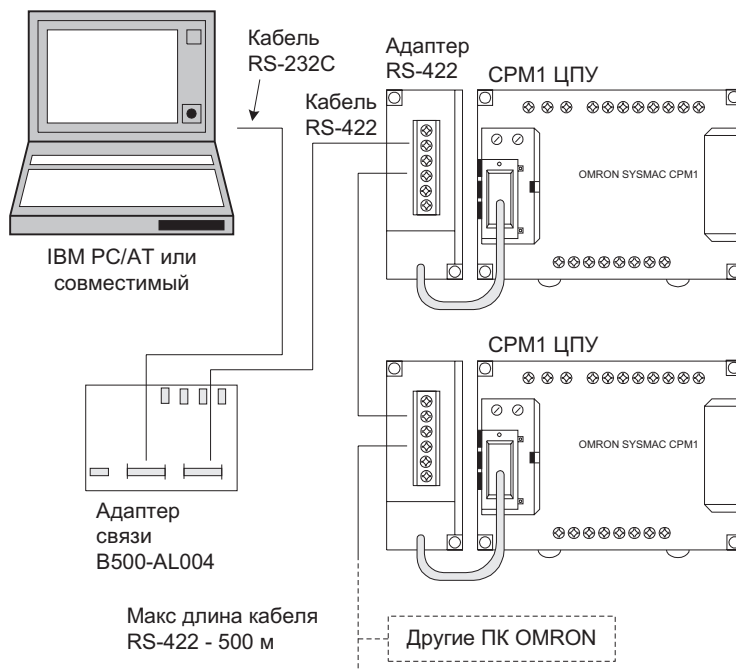
### Подключение к программируемому терминалу

На схеме показаны возможные методы соединения между CPM1 и программируемым терминалом (индицирующим устройством).



### Связь 1:N

На схеме показано, как подключить к IBM PC/AT или совместимому компьютеру до 32 ПК OMRON.



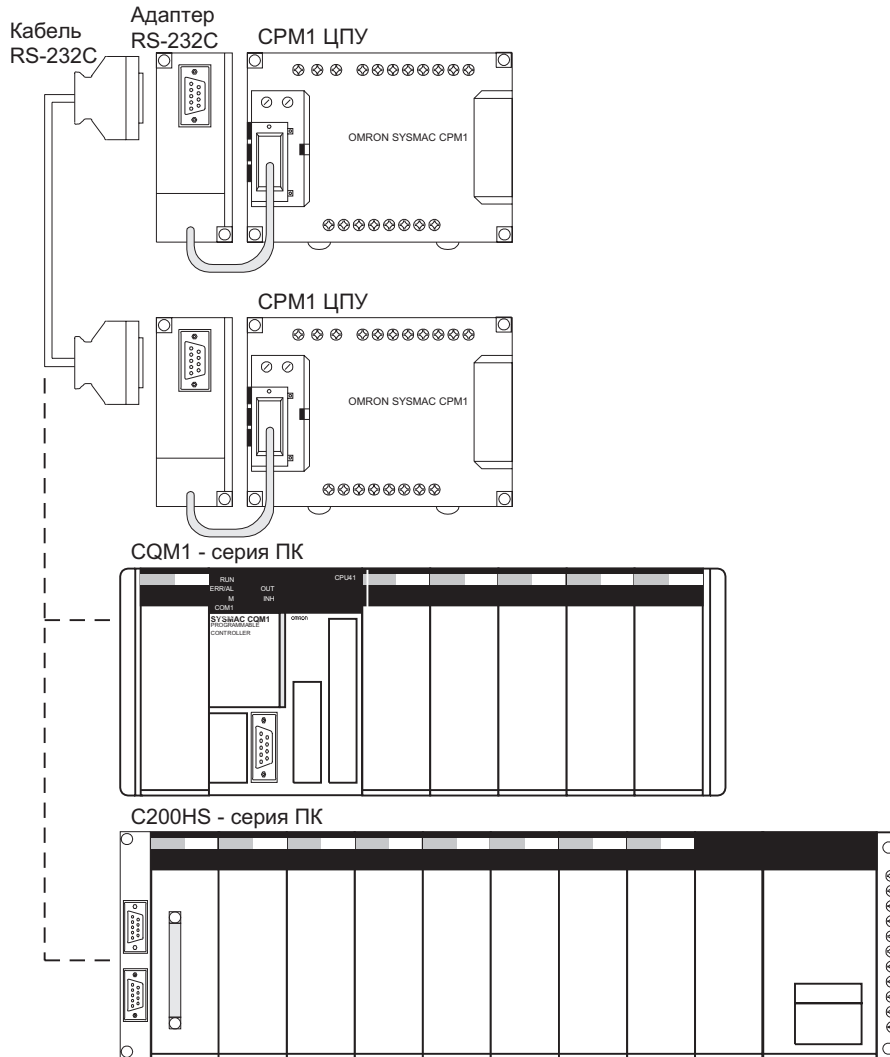
### Адаптеры и кабели

В следующей таблице перечисляются адаптеры и кабели, используемые при связи HOST LINK.

Наименование	Применение	Номер модели
Адаптер RS-232C	Осуществляет преобразования в уровень периферийного порта	CPM1-CIF01
Адаптер RS-422		CPM1-CIF11
Соединительные кабели	Служат для подключения IBM PC/AT или совместимых компьютеров (длина: 3.3 м)	CPM1-CIF02
Адаптер связи	Осуществляет преобразования между форматами RS-232C и RS-422	NT-AL001

### 1.2.3 Линии связи 1:1

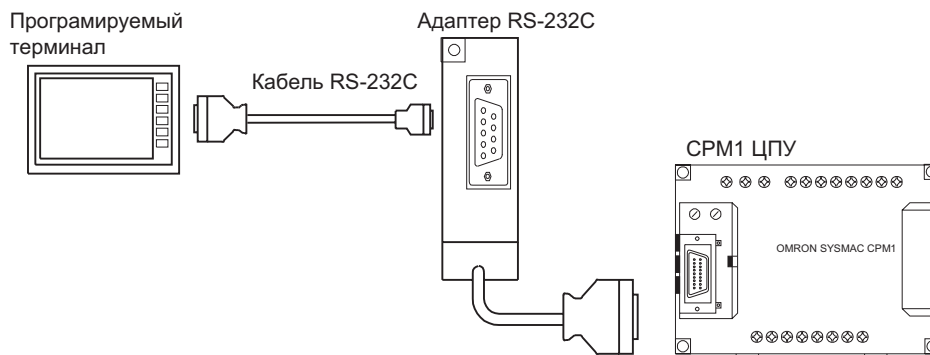
Можно создать линию связи (DATA LINK) с областью данных в другом CPM1, CQM1 или C200HS. Для связи 1:1 используется адаптер RS-232C.



Наименование	Применение	Номер модели
Адаптер RS-232C	Осуществляет преобразование в формат периферийного порта	CPM1-CIF01

### 1.2.4 Связь NT LINK

Используя связь NT CPM1 через адаптер CPM1 можно подключить к программируемому терминалу (интерфейс NT LINK) через адаптер RS-232C.



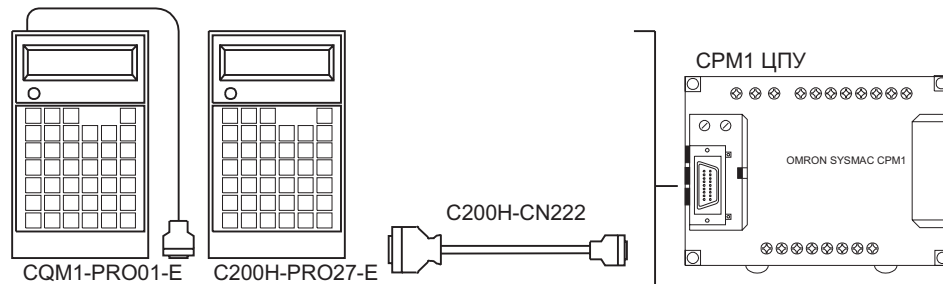
Наименование	Применение	Номер модели
Адаптер RS-232C	Осуществляет преобразование в формат периферийного порта	CPM1-CIF01

### 1.2.5 Линии связи периферийных устройств

Программу CPM1 можно написать и отредактировать с помощью программатора или персонального компьютера с использованием SSS или LSS.

#### Программатор

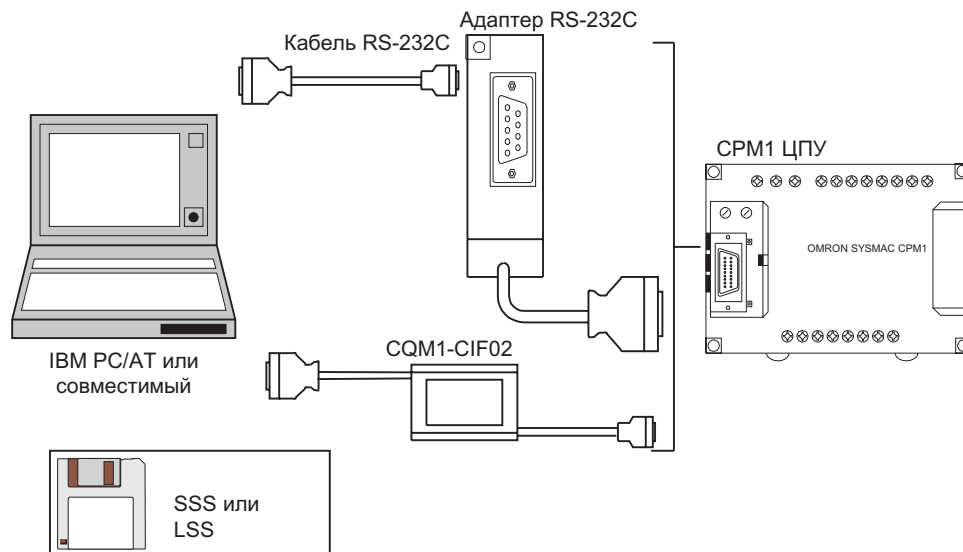
К CPM1 можно подключить программаторы CQM1-PRO01-E или C200H-PRO27-E, как показано на схеме.



Наименование		Номер модели
Программатор серии CQM1 (Поставляется с кабелем)		CPM1-PRO01-E
Программатор серии C200H		CPM1-PRO27-E
Соединительные кабели серии C200H	Длина кабеля: 2м	CPM1-CN222
	Длина кабеля: 4м	CPM1-CN422

#### SSS, LSS

К CPM1 можно подключить персональный компьютер с пакетом программ SSS или LSS, как показано на схеме. Разводку кабеля RS-232C см. 3-4-7. Версия SSS должна быть не ниже 2. Подробности о SSS см. 4.1



Наименование	Применение	Номер модели
Адаптер RS-232C	Осуществляет преобразование в формат периферийного порта	CPM1-CIF01
Соединительный кабель	Служит для подключения компьютеров IBM PC/AT или совместимых (Длина: 3.3 м).	CPM1-CIF02 (см. прим.)
SSS	Для компьютеров IBM PC/AT или совместимых (дискета 3.5", 2DD).	CPM1-ZL3AT1-E

*Замечание* При покупке указывайте номер модели.

## **Глава 2. Технические характеристики блоков и их компоненты**

*В данной главе приведены технические характеристики блоков, составляющих ПК, и описаны основные компоненты блоков.*

## 2.1 Характеристики

### 2.1.1 Общие характеристики

Параметр		CPM1-10CDR-	CPM1-20CDR-	CPM1-30CDR-
Напряжение питания	перем. тока	100 ... 240 В перем. тока, 50/60 Гц		
	пост. тока	24 В пост. тока		
Рабочее напряжение	перем. тока	85 ... 264 В перем. тока		
	пост. тока	20.4 ... 26.4 В пост. тока		
Энергопотребление	перем. тока	макс. 60 ВА		
	пост. тока	макс. 20 Вт		
Пусковой ток		макс. 60 А		
Сервисный выход (только для CPM1 перем. тока)		24 В пост. тока 300 мА (см. прим.)		
Сопротивление изоляции		Мин. 20 мОм (при 500 В пост. тока) между клеммами внешней сети перем. тока и клеммами защитного заземления.		
Электрическая прочность		2 300 В пер. тока 50/60 Гц (1 минута) между клеммами внешней сети перем. тока и клеммами защитного заземления. Ток утечки: макс. 10 мА.		
Помехоустойчивость		1 500 В двойной амплитуды, ширина импульса: 0.1 ... 1 мкс, время нарастания: 1 нс (при имитации помехи).		
Виброустойчивость		10 ... 57 Гц, амплитуда 0,075 мм, 57 ... 150 Гц, ускорение: 9.8 м/с <sup>2</sup> (1G) в направлении X, Y, Z по 80 мин. на каждое. (Временной коэф.: 8 мин. x коэф. 10 = общее время 80 мин).		
Сопротивление удару		147 м/с <sup>2</sup> (15 G) 3 раза в каждом направлении X, Y, Z		
Температура окружающей среды		При работе: 0 ... 55 С При хранении: -20 ... 75 С		
Влажность		10 % ... 90 % (без конденсации)		
Атмосфера		Без корродирующих газов		
Размер винтов клемм		M3		
Заземление		менее 100 Ом		
Устойчивость к прерыванию питания		Перем. тока: 10 мс. минимум Пост. тока: 2 мс. минимум (Прерывание по питанию случается при снижении питания менее 85% номинала на время, большее времени прерывания питания).		
Степень защиты		IEC IP50		
Вес ЦПУ	перем. тока	макс. 600 г	макс. 800 г	макс. 900 г
	пост. тока	макс. 500 г	макс. 700 г	макс. 800 г
Вес блока дополнительных входов/выходов		макс. 600 г		

*Замечание* Когда внешний источник питания выдает излишний ток или закорочен, напряжение внешнего источника питания падает. Однако ПК будет продолжать работу.

### 2.1.2 Характеристики

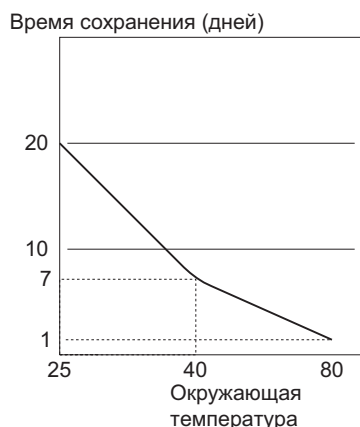
Параметр	CPM1-10CDR-__	CPM1-20CDR-__	CPM1-30CDR-__
Метод управления	По записанной программе		
Метод управления входами/выходами	Циклическое сканирование с прямым выходом; Обработка с непосредственным обновлением.		

Параметр	CPM1-10CDR-__	CPM1-20CDR-__	CPM1-30CDR-__
Язык программирования	Релейно-контактные схемы		
Длина команды	1 шаг на команду, 1 ... 5 слов на команду		
Типы команд	Базовых команд: 14 Специальных команд: 77 типов, 134 команды		
Время исполнения	Базовые команды: 0.72 ... 16.2 мкс Специальные команды: 16.3 мкс (команда MOV)		
Объем памяти программы	2 048 слов		
Биты входа	Биты входа: 00000 ... 00915 (Биты, не задействованные в качестве входных, можно использовать в качестве рабочих битов).		
Биты выхода	Биты выхода: 00000 ... 00915 (Биты, не задействованные в качестве выходных, можно использовать в качестве рабочих битов).		
Рабочие биты	640 бит 20000 ... 23915 (слова IR 200 ... IR 239)		
Специальные биты (область SR)	256 бит 24000 ... 25507 (слова IR 240 ... IR 255)		
Биты временного хранения (область TR)	8 бит TR0 ... TR7		
Биты долговременного хранения (область HR)	320 бит HR 0000 ... HR 1915 (слова HR 00 ... HR 19)		
Вспомогательные биты (область AR)	256 бит AR 0000 ... AR 1515 (слова AR 00 ... AR 15)		
Биты связи (область LR)	256 бит LR 0000 ... LR 1515 (слова LR 00 ... LR 15)		
Таймеры/счетчики	128 таймеров/счетчиков (TIM/CNT 000 ... TIM/CNT 127). 100-мс таймеры: TIM 000 ... TIM 127 10-мс таймеры (высокоскоростные счетчики): TIM 000 ... TIM 127 (номера таймеров - те же, что и для 100 мс таймеров) Декрементирующие счетчики и реверсивные счетчики		
Память параметров	Чтение/запись: 1 024 слова (DM 0000 ... DM 1023) Только чтение: 512 слов (DM 6144 ... DM 6655)		
Обработка прерываний	Внешних прерываний - 4 (Время реакции: макс. 0.3 мс)	Внешних прерываний - 4 (Время реакции: макс. 0.3 мс)	
Защита памяти	При отключении питания поддерживаются: содержимое HR, AR, DM, значение счетчиков.		
Независимое питание памяти	Память FLASH: Программа и память DM только для чтения сохраняются без аккумулятора. Сохранение от конденсатора: Область чтения/записи DM, область HR, область AR и область счетчиков поддерживается конденсатором 20 дней при 25 °С. Время зависит от окружающей температуры. Подробности см. на графике.		
Функции самодиагностики	Неисправности ЦПУ (контрольный таймер), сбой шины входов/выходов, сбой памяти.		
Проверки программы	Нет команды END, ошибки программирования (проверяется постоянно при работе).		
Высокоскоростной счетчик	Один высокоскоростной счетчик: 5 кГц однофазный или 2.5 кГц двухфазный (метод линейного счета) Инкрементальный режим: 0 ... 65 535 (16 бит) Режим плюс/минус: -32 767 ... 32 767 (16 бит)		
Входы быстрого реагирования	Те же самые входы служат в качестве входов быстрого реагирования и входов внешних прерываний. (Мин. ширина импульса: 0.2 мс).		

Параметр	CPM1-10CDR-__	CPM1-20CDR-__	CPM1-30CDR-__
Константа времени входа	Может быть установлен на 1, 2, 4, 8, 16, 32, 64 или 128 мс.		
Аналоговые регуляторы	2 (0 ... 200 - двоично-десятичные)		

### Зависимость времени сохранения памяти от температуры

Конденсатор в ЦПУ служит для сохранения программы, значений счетчика и содержимого областей DM, HR и AR. Время сохранения зависит от окружающей температуры, как показано на следующем графике.



Если содержимое области программ ЦПУ утеряно, при пуске в область программ будет считана программа, записанная в памяти FLASH. Хотя программа будет восстановлена, значения счетчиков и областей DM, HR и AR будут утеряны.

### 2.1.3 Характеристики входов/выходов

#### Характеристики входов ЦПУ

Параметр	Спецификация
Входное напряжение	24 В пост. тока +10% / -15 %
Входной импеданс	Входы IN 0000 ... IN 00002: 2 кОм, другие входы 4.7 кОм
Входной ток	Входы IN 0000 ... IN 00002: 12 мА другие входы 5 мА
Напряжение уровня 1	Минимум 14.4 В пост. тока
Напряжение уровня 0	Максимум 5.0 В пост. тока
Задержка включения в 1	Максимум 8 мс.
Задержка выключения в 0	Максимум 8 мс.
Функциональная схема	<p>Прим.: Цифры в скобках - для IN00000 ... IN00002</p>

*Замечание* При помощи установочных параметров можно задавать 1, 2, 4, 8, 16, 32, 64 или 128 мс. Если входы IN 0000 ... IN 00002 используются в качестве входов высокоскоростного счетчика, задержки указаны в таблице.

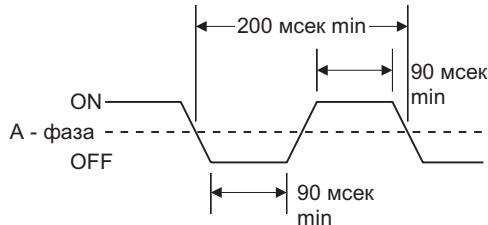
Вход	Режим инкрементального счета	Режим разности фаз
IN00000 (фаза A)	5 кГц	2.5 кГц



Вход	Режим инкрементального счета	Режим разности фаз
IN00000 (фаза В)	Нормальный вход	
IN00000 (фаза Z)	Установка в 1: минимум 100 мкс, Установка в 0: минимум 500 мкс.	

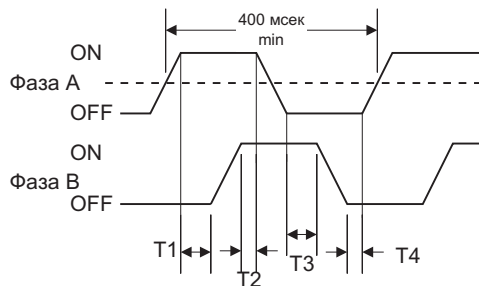
Минимальная задержка имеет следующий вид:

#### Режим инкремента (макс. 5 кГц)

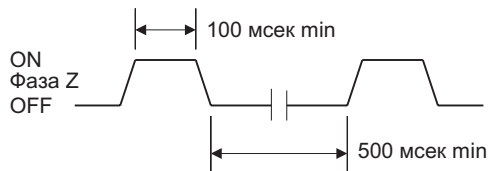


#### Режим разности фаз (макс. 2.5 кГц)

IN00000 (фаза А), IN00001 (фаза В)



IN00000 (фаза Z)



Когда IN00003 ... IN00006 используются, как входы прерываний, макс. задержка = 0.3 мс. Задержка измеряется от времени, когда вход устанавливается в 1 до выполнения программы прерывания.

#### Характеристики входов блока расширения

Параметр	Значение
Входное напряжение	24 В пост. тока +10% / -15 %
Входной импеданс	4.7 кОм
Входной ток	5 мА
Напряжение уровня 1	Минимум 14.4 В постоянного тока
Напряжение уровня 0	Максимум 5.0 В постоянного тока
Задержка включения в 1	Максимум 8 мс.
Задержка выключения в 0	Максимум 8 мс.
Функциональная схема	

**Замечание** При помощи установочных параметров можно задавать 1, 2, 4, 8, 16, 32, 64 или 128 мс.

### Характеристики выходов ЦПУ и блока расширения

Параметр	Значение
Тип выхода	Все выходы релейные
Макс. переключаемая нагрузка	2 А, 250 В пер. тока ( $\cos \phi = 1$ ) 2 А, 24 В пост. тока (4А / общий)
Мин. переключаемая мощность	10 мА, 5 В пост. тока
Тип реле	G6D-1A
Прочность реле	Электрическая: 300 000 операций (активная нагрузка) 100 000 операций (индуктивная нагрузка) Механическая: 20 000 000 операций
Задержка включения в 1	макс. 10 мс
Задержка выключения в 0	макс. 5 мс
Функциональная схема	

### 2.1.4 Характеристики адаптеров связи

#### Характеристики адаптера RS-232C

Параметр	Значение
Функции	Согласование форматов CMOS (КМОП) (со стороны ЦПУ ПК) и формата RS-232C (со стороны периферийного устройства)
Развязка	RS-232C (со стороны периферийного устройства) развязан преобразователем DC/DC и оптронной парой.
Питание	Подается от ЦПУ ПК
Энергопотребление	макс. 0.3 А
Скорость передачи	макс. 38 400 бод
Расстояние передачи	Общая длина: 15 м макс.
Виброустойчивость	10 ... 57 Гц; амплитуда 0,075 мм 57 ... 150Гц, ускорение: 9.8 м/с <sup>2</sup> (1G) в направлении X,Y,Z по 80 мин. на каждую ось. (Временной коэф.: 8 мин. x коэф. 10 = общее время 80 мин).
Сопrotивление удару	147 м/с <sup>2</sup> (15 G) 3 раза в каждом направлении X,Y,Z
Температура окружающей среды	При работе: 0 ... 55 °C При хранении: -20 ... 75 °C
Влажность	10 % ... 90 % (без конденсации)
Атмосфера	Без корродирующих газов
Вес	Макс. 200 г

#### Характеристики адаптера RS-422

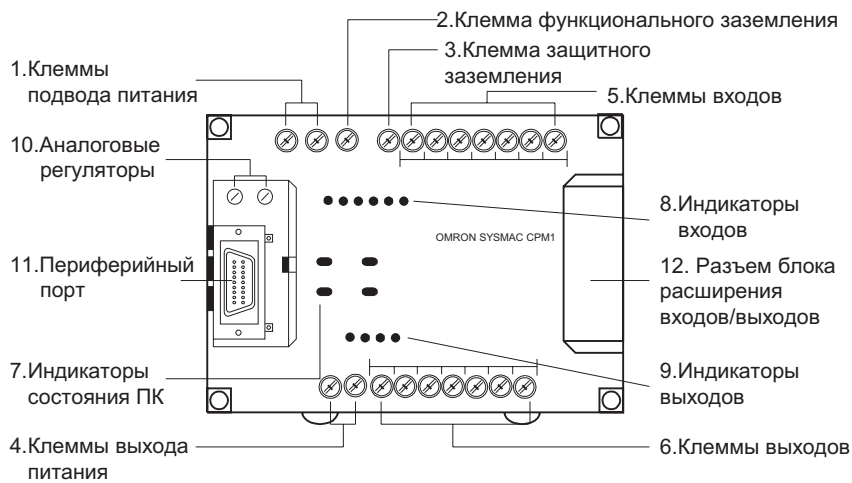
Параметр	Значение
Функции	Согласование форматов CMOS (КМОП) (со стороны ЦПУ ПК) и формата RS-422 (со стороны периферийного устройства)

Параметр	Значение
Развязка	RS-422 (со стороны периферийного устройства) развязан преобразователем DC/DC и оптронной парой.
Питание	Подается от ЦПУ ПК
Энергопотребление	макс. 0.3 А
Скорость передачи	макс. 38 400 бод
Расстояние передачи	Общая длина: 500 м макс.
Виброустойчивость	10 ... 57 Гц; амплитуда 0,075 мм 57 ... 150Гц, ускорение: 9.8 м/с <sup>2</sup> (1G) в направлении X,Y,Z по 80 мин. на каждую ось. (Временной коэф.: 8 мин. x коэф. 10 = общее время 80 мин).
Соппротивление удару	147 м/с <sup>2</sup> (15 G) 3 раза в каждом направлении X,Y,Z
Температура окружающей среды	При работе: 0 ... 55 °С При хранении: -20 ... 75 °С
Влажность	10 % ... 90 % (без конденсации)
Атмосфера	Без корродирующих газов
Вес	Макс. 200 г

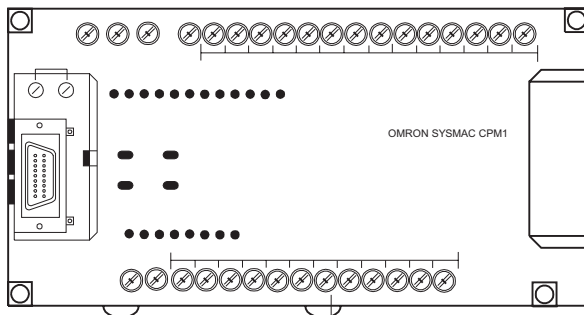
## 2.2 Компоненты блоков

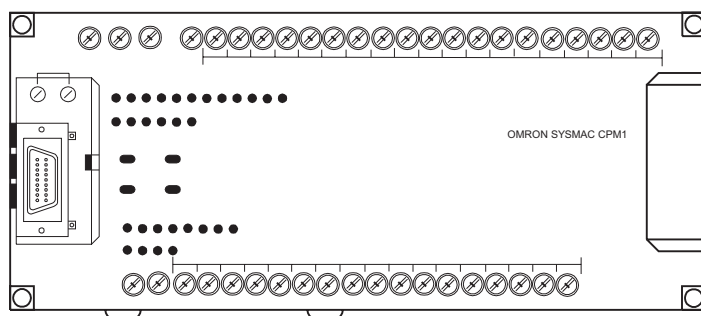
### 2.2.1 Компоненты ЦПУ

#### CPM1-10CDR-\_\_



#### CPM1-20CDR-\_\_



**CPM1-30CDR-****Описание компонентов ЦПУ**

- 1, 2, 3,... 1. Клеммы подвода питания  
 К данным клеммам подключается питание (100 ...240 В перем. тока или 24 В пост. тока)
2. Клемма функционального заземления  
 Обязательно заземлите данную клемму (только при ПК для перем. тока) для защиты от помех и уменьшения риска поражения током.
3. Клемма защитного заземления  
 Обязательно заземлите данную клемму для уменьшения риска поражения током.
4. Клеммы выхода питания  
 В CPM1 с питанием переменным током есть клеммы выдачи питания 24 В постоянного тока для запитки входных устройств.
5. Клеммы входов  
 Подключаются ко входным цепям.
6. Клеммы выходов  
 Подключаются к выходным цепям.
7. Индикаторы состояния ПК

Индикаторы указывают состояние работы ПК, как показано в таблице

Индикатор	Состояние	Значение
POWER (зеленый) (питание)	1	Питание подано на ПК
	0	Питание не подано на ПК
RUN (зеленый) (работа)	1	ПК находится в режиме RUN или MONITOR
	0	ПК находится в режиме PROGRAM или случилась фатальная ошибка
ERROR/ALARM (красный) (ошибка)	1	Произошла фатальная ошибка (работа ПК прекращается)
	Мигает	Произошла нефатальная ошибка (работа ПК продолжается)
	0	Нормальная работа
COMM (оранжевый) (связь)	1	Данные передаются по периферийному порту.
	0	Данные не передаются по периферийному порту.

## 8. Индикаторы входов

Горят, когда состояние соответствующего входа = 1. При фатальной ошибке входные индикаторы срабатывают следующим образом:

Ошибка ЦПУ или шины входов/выходов: Входные индикаторы = 0

Ошибка памяти или системная: Входные индикаторы сохраняют состояние, которое было до ошибки, даже если фактическое состояние входов поменялось.

#### 9. Индикаторы выходов

Горят, когда состояние соответствующего выхода = 1.

#### 10. Аналоговые регуляторы

Регулированием данных регуляторов меняется содержание IR 250 и IR 251 от 0 до 200.

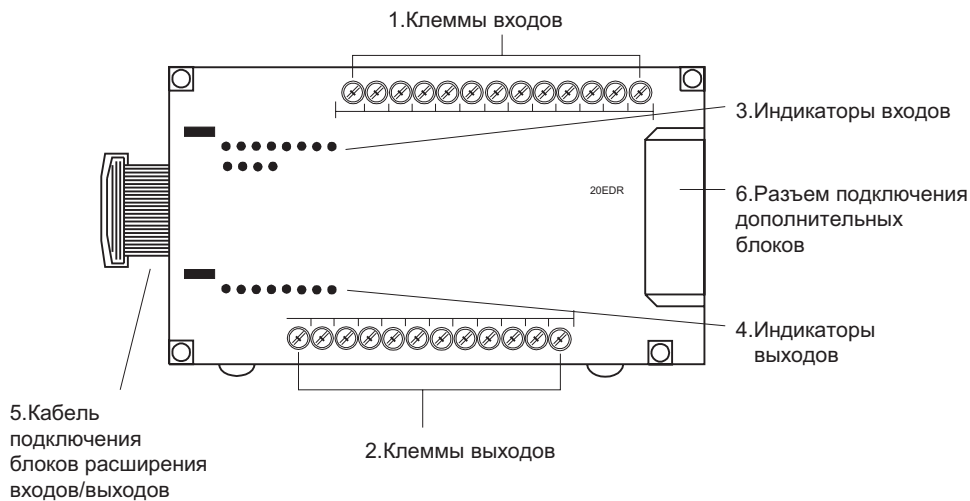
#### 11. Периферийный порт

Соединяет ПК с периферийным устройством, адаптером RS-232C или RS-422.

#### 12. Разъем блока расширения входов/выходов

Соединяет ПК с блоком расширения входов/выходов для добавления 12 входов и 8 выходов. Компоненты блока расширения входов/выходов

### 2.2.2 Компоненты блока расширения



1, 2, 3,... 1. Клеммы входов.

Подключаются ко входным цепям.

2. Клеммы выходов.

Подключаются к выходным цепям.

3. Индикаторы входов.

Горят, когда состояние соответствующего входа = 1.

4. Индикаторы выходов.

Горят, когда состояние соответствующего выхода = 1.

5. Кабель подключения блоков расширения входов/выходов.

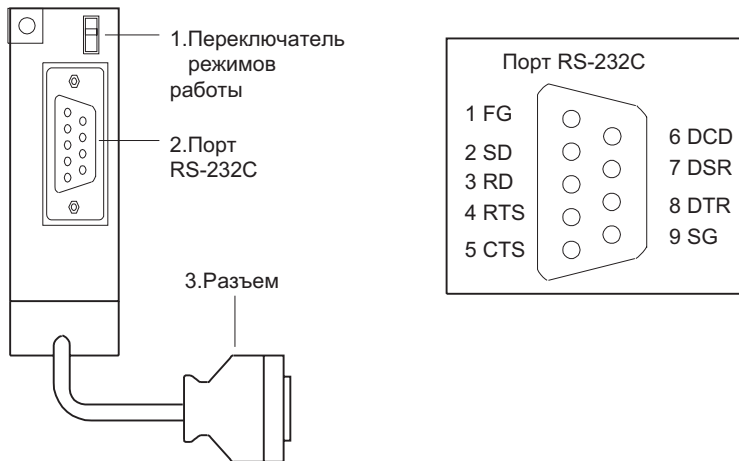
Подключает блок расширения входов/выходов к ЦПУ.

6. Разъем подключения дополнительных блоков

Подключает к блоку расширения входов/выходов специальный блок входов/выходов (пока нет).

### 2.2.3 Компоненты адаптеров связи

#### Адаптер RS-232C



1, 2, 3,... 1. Переключатель режимов работы.

При связи HOST LINK с персональным компьютером установите переключатель в состояние HOST (ведущий). При связи с программируемым терминалом или 1:1 установите переключатель в состояние "NT".

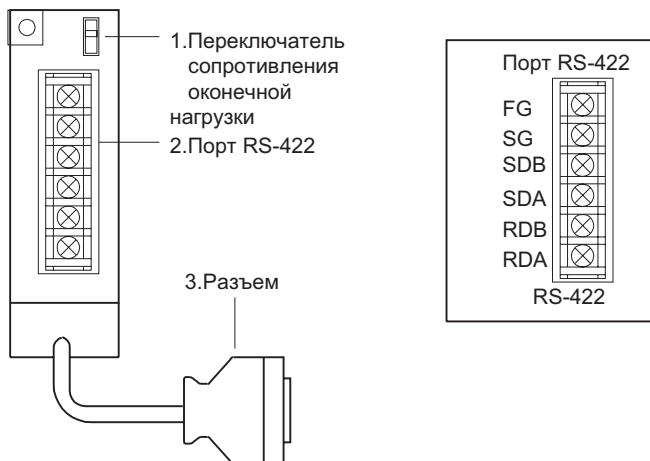
2. Разъем

Подключается к периферийному порту.

3. Порт RS-232C

Подключает к RS-232C кабель от другого устройства, таких как персональный компьютер, периферийное устройство или программируемый терминал.

#### Адаптер RS-422



1, 2, 3,... 1. Переключатель сопротивления оконечной нагрузки.

Установите Переключатель сопротивления оконечной нагрузки в ON (вверх) для адаптеров связи на обеих сторонах системы HOST LINK и для адаптера RS-422.

2. Разъем

Подключается к периферийному порту.

3. Порт RS-422

Подключается к сети HOST LINK.

## **Глава 3. Монтаж и подключение**

*В данной главе приведена информация о монтаже и подключении СРМ1. Следуйте указаниям данной инструкции при установке СРМ1 на панели или в шкафу, подключении питания и входов/выходов.*

### 3.1 Меры предосторожности

При разработке систем управления с СРМ1 соблюдайте следующие меры предосторожности.

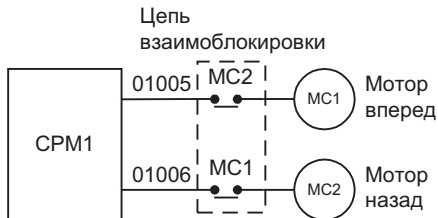
#### 3.1.1 Подключение питания

Отделяйте подвод питания от линий системы управления, системы СРМ1 и линий входов/выходов постоянного тока.

#### 3.1.2 Цепи взаимоблокировки и аварийного ограничения

Установите внешние цепи взаимоблокировки на выходах, когда выходы СРМ1 используются для осуществления взаимоисключающих операций (например, включение вперед и назад двигателя) или если некорректные операции ПК могут вызвать повреждения. Также устанавливайте внешние аварийные ограничения для предотвращения выезда за заданные границы в системах управления позиционированием.

На схеме показан пример цепи взаимоблокировки.



#### 3.1.3 Прерывания питания ПК

##### Снижение напряжения питания

При снижении питания ниже 85 % номинального ПК прекращает работу и выходы выключаются в 0.

##### Обнаружение мгновенных провалов питания

Мгновенный провал питания менее 10 мс при питании переменным током и 2 мс при питании постоянным током не обнаруживается и ЦПУ продолжает работу.

Мгновенный провал питания более 10 мс при питании переменным током и 2 мс при питании постоянным током могут обнаруживаться, а могут и не обнаруживаться.

При обнаружении сбоя питания ПК прекращает работу и выходы выключаются в 0.

##### Автоматическое возобновление работы

Работа возобновится автоматически при восстановлении напряжения свыше 85 % номинального.



*Замечание* СРМ1 может повторять операции старт-стопа, если напряжение питания плавно меняется возле 85% номинального.

Если это отрицательно влияет на оборудование, обеспечьте внешнюю защитную цепь, которая выключит выходы при падении напряжения ниже номинального.

##### Время от включения питания до начала операции

Время от включения питания до начала операции варьирует в зависимости от условий операции, таких как напряжение питания, конфигурация, окружающая температура и т. д. Минимальное время - около 300 мс.



## 3.2 Выбор места установки

СРМ1 устойчив к неблагоприятным окружающим условиям и высоконадежен, но установка ПК в благоприятных условиях увеличивает надежность и срок службы.

### 3.2.1 Условия места установки

**Не устанавливайте ПК в места со следующими условиями:**

- Прямой солнечный свет.
- Окружающая температура выходит за зону 0 ... 55 °С.
- Относительная влажность выходит за зону 10% ... 90%.
- Из-за резких изменений температуры появляется конденсация.
- Присутствие корродирующих газов, воспламеняющихся газов или соли.
- Наличие брызг воды, масла или химикатов.
- Прямая вибрация и удары.

Условия места установки должны удовлетворять характеристикам СРМ1. Подробности см. 2.1.1.

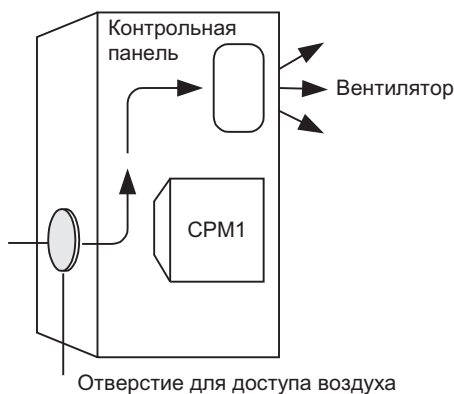
### 3.2.2 Установка на панели/в шкафу

Учитывайте условия работы ПК, условия обслуживания и окружающие условия при выборе установки ПК на панели или в шкафу

#### Перегрев

Диапазон рабочих температур СРМ1 0 ... 55 °С. Обеспечивайте необходимую вентиляцию.

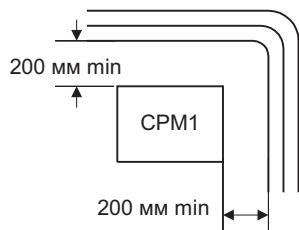
- Оставляйте достаточно места для циркуляции воздуха.
- Не устанавливайте СРМ1 над оборудованием, излучающим большое количество тепла, таким, как нагреватели, трансформаторы или большие резисторы.
- При превышении температуры 55 °С устанавливайте вентилятор или систему охлаждения.



#### Электрические помехи

Линии питания и линии высокого напряжения могут вызывать помехи в ПК.

- Не устанавливайте СРМ1 на панели или шкафу с высоковольтным оборудованием.
- Оставляйте не менее 200 мм между СРМ1 и ближайшими линиями питания.



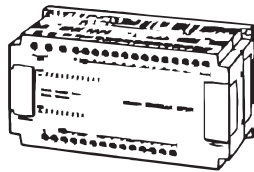
**Доступ**

Обеспечьте доступ к СРМ1 для работы и обслуживания.

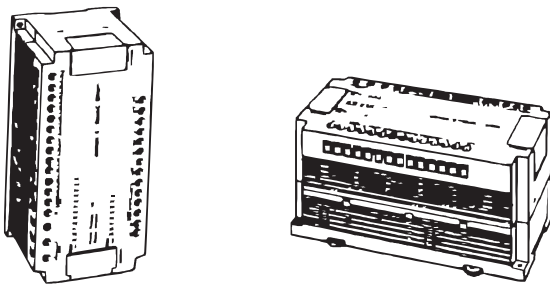
- К СРМ1 должен быть обеспечен легкий доступ для работы и обслуживания. Если доступ к СРМ1 преграждают высоковольтные линии или провода питания, это опасно.
- Доступ к ПК будет облегчен при его установке на панели или в шкафу на высоте 3 ... 5 футов (0.9 ... 1.5 м) от пола.

**3.3 Монтаж СРМ1****3.3.1 Ориентация СРМ1**

Для надлежащего охлаждения следует монтировать СРМ1 в положении, указанном на рисунке:



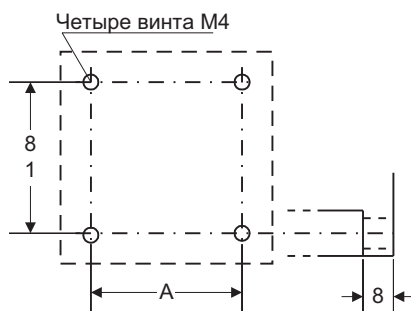
Не устанавливайте СРМ1 в следующих позициях:

**3.3.2 Монтаж СРМ1**

СРМ1 можно монтировать на ровной плоскости или на рейке DIN.

**Монтаж на плоскости**

При установке СРМ1 на ровной плоскости пользуйтесь следующим рисунком:



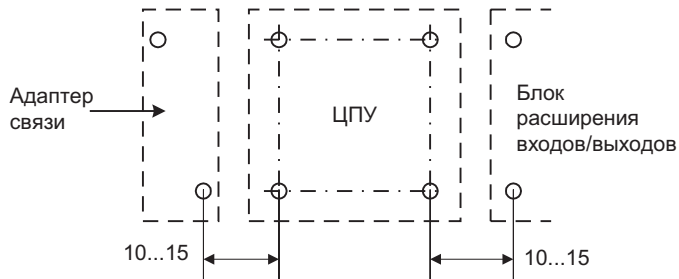
Используйте винты М4 x 15

Ширина (А) между монтажными отверстиями зависит от модели СРМ1.

Номер модели	Ширина (А)
ЦПУ СРМ1-10СDR-Ъ	121 мм
ЦПУ СРМ1-20СDR-Ъ	171 мм

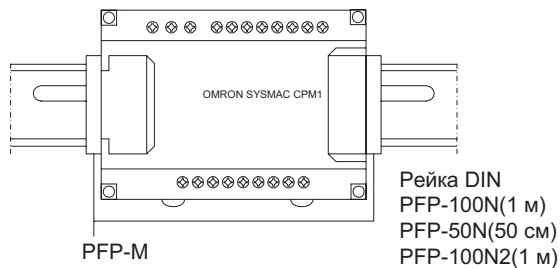
Номер модели	Ширина (А)
ЦПУ СРМ1-30СDR-Т	221 мм
Блок расширения СРМ1-20ЕDR	171 мм
Адаптер RS-232С	21 мм
Адаптер RS-422	21 мм

При установке СРМ1, блока расширения и адаптеров связи на ровной плоскости пользуйтесь следующим рисунком:



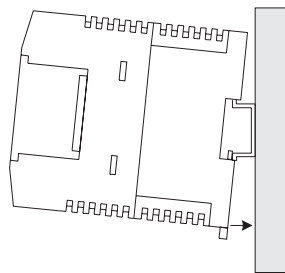
### Установка на рейке DIN

СРМ1 может быть установлен на рейке DIN 35 мм.



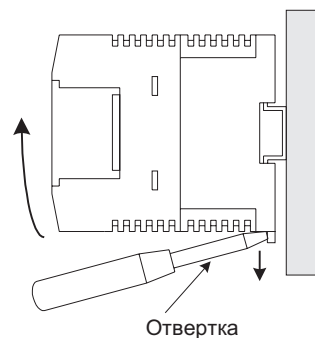
### Монтаж

Опустите СРМ1, чтобы паз на задней стороне ПК наделся на верх рейки DIN. Нажмите ПК, пока защелка не защелкнется.



### Демонтаж

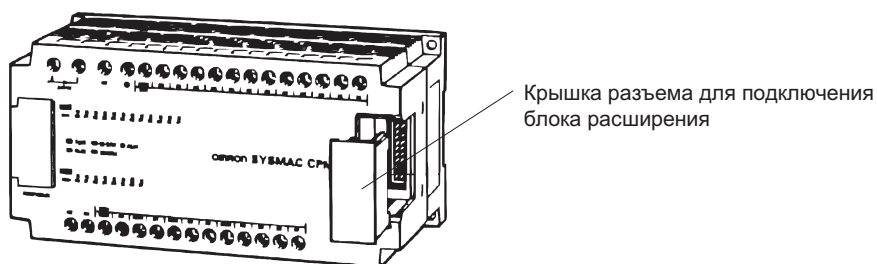
Отожмите отверткой защелку вниз и поверните ПК вверх для того, чтобы снять.



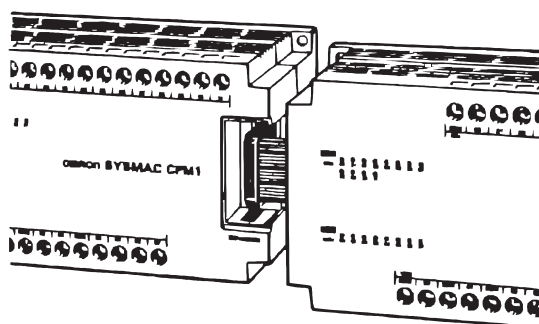
### 3.3.3 Подключение блока расширения входов/выходов

К ЦПУ можно подключить один блок расширения входов/выходов. При подключении блока расширения входов/выходов используйте следующие процедуры.

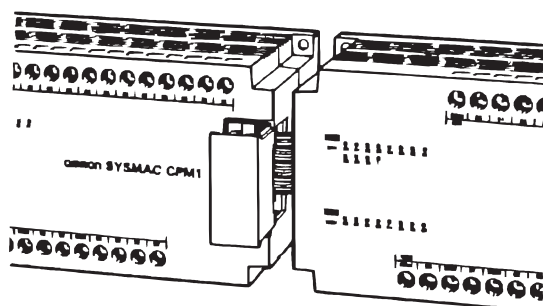
*Замечание* 1. Снимите крышку с разъема ЦПУ блока расширения входов/выходов.



2. Вставьте кабель блока расширения входов/выходов в разъем, который находится на ЦПУ.



3. Снова поставьте крышку на разъем блока расширения входов/выходов.



## 3.4 Подключение

В данной главе приведена основная информация о подключении блока питания и блока расширения входов/выходов и периферийных устройств.

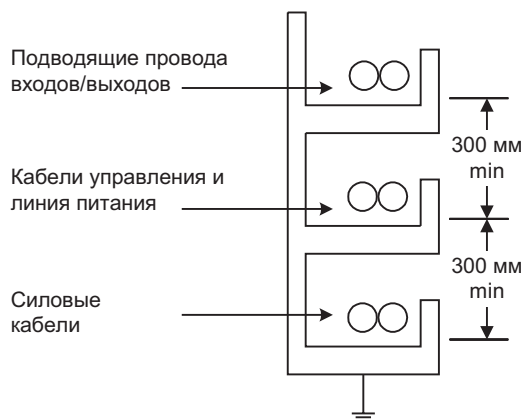
### 3.4.1 Общие меры предосторожности при подключении

#### Помехи на линиях входов/выходов

Не располагайте подводящие провода входов/выходов в одном канале с линиями питания.

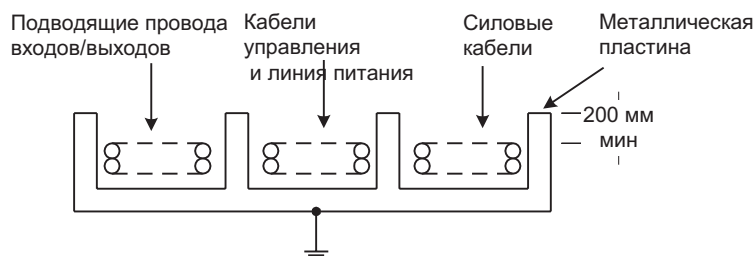
#### Подвесные кабельные каналы

Оставляйте не менее 300 мм между силовыми кабелями и проводами входов/выходов или управления.



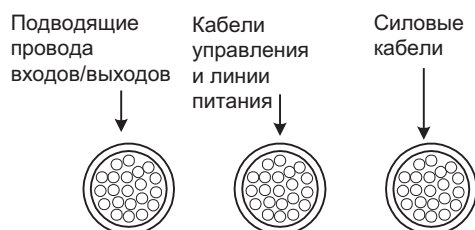
#### Кабельные каналы, расположенные на полу

Оставляйте не менее 200 мм между проводами и верхней точкой кабельного канала, как показано на схеме.



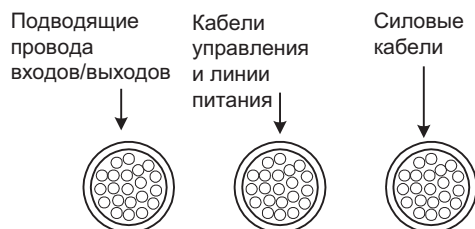
#### Прокладка в трубах

Располагайте линии входов/выходов, линии питания и управления СРМ1 и силовые кабели в соответствии со схемой.



#### Заземление

Заземлите клеммы функционального и защитного заземления с сопротивлением не более 100 Ом для защиты от поражения током и некорректных операций из-за помех. Обязательно используйте для заземления провода сечением не менее 2 мм<sup>2</sup>.



#### Обжимные наконечники

Всегда используйте обжимные наконечники для подключения линий питания СРМ1 и входов/выходов, в противном случае используйте одножильный провод (вместо многожильного). Торчащая отдельная жила провода может вызвать короткое замыкание.

Используйте винты М3 для клемм и надежно их зажимайте (0.5 ... 0.6 Нм).

Рекомендуемые наконечники:

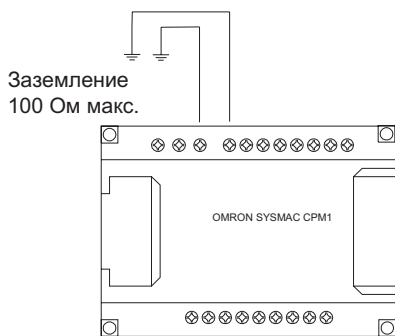
### 3.4.2 Подключение питания

#### Питание 100 ... 240 В переменного тока

Для питания СРМ1 используйте отдельную цепь, чтобы не было падения напряжения, вызванного пусковым током при включении другого оборудования.

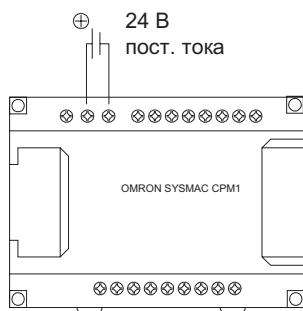
Когда используются несколько СРМ1, рекомендуется подключать питание по отдельным цепям, чтобы не было падения напряжения из-за пускового тока или при неправильном срабатывании выключателя

Для предотвращения влияния помех от линий питания используйте витые провода питания. Добавление развязывающего трансформатора 1:1 еще больше уменьшает влияние помех.



#### Питание 24 В постоянного тока

Используйте источник питания постоянного тока достаточной мощности, с малыми пульсациями и усиленной изоляцией.



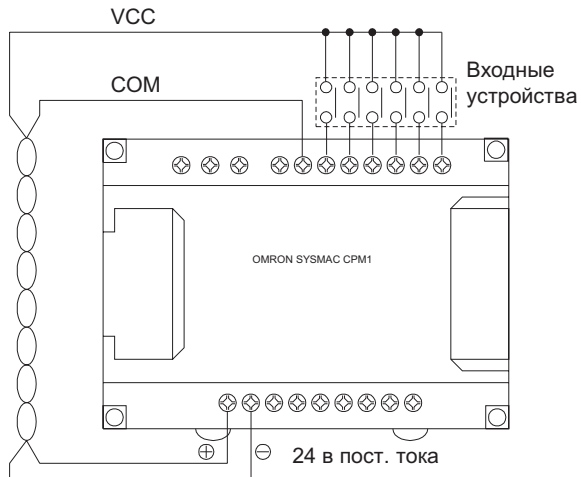
**Внимание!** Не проводите испытания на диэлектрическую прочность ПК постоянного тока. Тест может повредить внутренние компоненты ПК.

### 3.4.3 Подключение входов

Входы ЦПУ СРМ1 и блоков расширения подключайте в соответствии со следующими схемами. Используйте обжимные наконечники или одножильные провода для подключения к ПК. У ПК переменного тока можно использовать клеммы “выдача питания”.

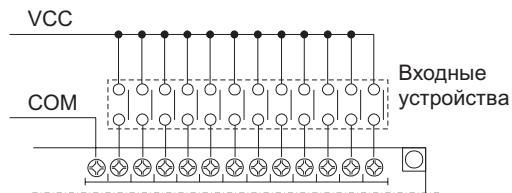
#### СРМ1-10СДР-\_\_

На схеме показано ЦПУ переменного тока. На ЦПУ постоянного тока нет выходов “выдача питания”.



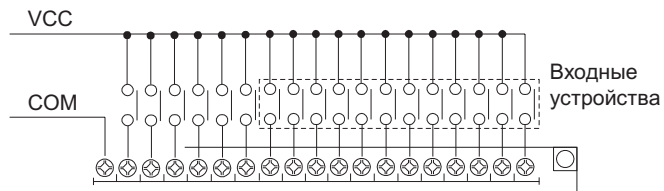
#### CPM1-20CDR-   и блок расширения CPM1-20EDR

На схеме показана конфигурация входов для ЦПУ CPM1-20CDR-   и блока расширения CPM1-20EDR.



#### CPM1-30CDR-

На схеме показана конфигурация входов для ЦПУ CPM1-30CDR-  .



### Входные устройства

В таблице показано подключение входных устройств различных типов.

Устройство	Схема подключения
Релейный выход	
Открытый коллектор NPN	
Токовый выход NPN	
Токовый выход PNP	
Выход напряжения	

### Ток утечки (24 В пост. тока)

Ток утечки может вызвать ложное срабатывание входа при использовании 2-проводных датчиков (бесконтактных выключателей или фотоэлектрических выключателей) или конечных выключателей со светодиодами.

Ложных срабатываний не произойдет, если ток утечки менее 1.0 мА (2.5 мА для входов IN00000 ... IN00002), но если он выше, поставьте шунтирующий резистор для уменьшения входного импеданса, как показано на схеме.



$$R = \frac{L_c \times 5.0}{I \times L_c - 5.0} \text{ кОм макс.}, W = \frac{2.3}{R} \text{ Вт мин.}$$

Где:

I: Ток утечки устройства (мА)

$L_c$ : входной импеданс CPM1 (кОм)

R: Сопротивление шунта (кОм)



$I_c$ : входной ток СРМ1 (мА)  
 $W$ : Мощность шунта (Вт)  
 $E_c$ : напряжение 0 СРМ1 (В) = 5.0 В  
 $U_c$ : входное напряжение (24 В)

Вышеприведенные уравнения выведены из следующих уравнений:

$$I \times \frac{R \times \frac{U_c}{I_c}}{R + \frac{U_c}{I_c}} \leq E_c, W \geq \frac{U_c}{R} \times U_c \times T$$

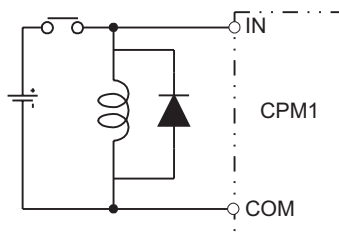
Подробности о значениях  $L_c$ ,  $I_c$  и  $E_c$  см. 2.1.3

Входной импеданс, входной ток и напряжение нуля могут меняться в зависимости от номера входа (значения IN00000 ... IN00002 отличаются от остальных).

#### Индуктивная нагрузка

При подключении ко входу индуктивной нагрузки параллельно с нагрузкой следует подключить диод, который должен удовлетворять следующим требованиям:

- 1, 2, 3,... 1. Напряжение пробоя должно минимум в 3 раза превышать напряжение нагрузки.
2. Ток выпрямления - не менее 1 А.



#### 3.4.4 Подключение выходов

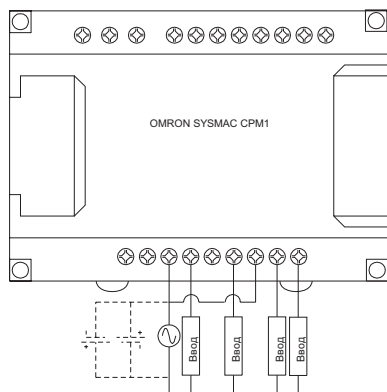
Выходы ЦПУ СРМ1 и блоков расширения подключайте в соответствии со следующими схемами. Используйте обжимные наконечники или одножильные провода для подключения к ПК. У ПК переменного тока можно использовать клеммы “выдача питания”.

- Всегда используйте либо одножильные провода, или (при использовании многожильных) напрессованные наконечники.
- Не превышайте выходную мощность или общий максимальный ток. Подробности см. 2.1.3.

Параметр	Значение
Выходная мощность	2А (250 В перем. тока или 24 В пост. тока)
Максим. мощность на общем (COMMON)	4 А/ общий

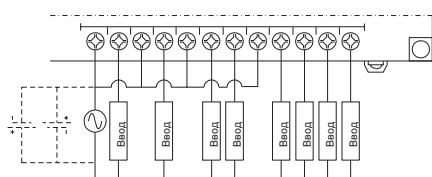
#### ЦПУ СРМ1-10СDR-\_\_

На схеме показано ЦПУ переменного тока. На ЦПУ постоянного тока нет выходов “выдача питания”.



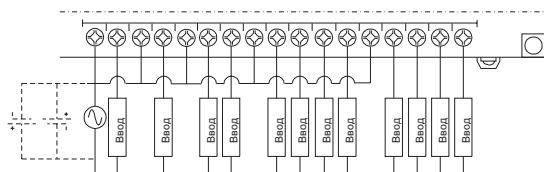
### CPM1-20CDR-\_\_ и блок расширения CPM1-20EDR

На схеме показана конфигурация выходов для ЦПУ CPM1-20CDR-Т и блока расширения CPM1-20EDR.



### CPM1-30CDR-\_\_

На схеме показана конфигурация входов для ЦПУ CPM1-30CDR-Т.



### Предосторожности при подключении выходов

Для защиты внутренних элементов ПК соблюдайте следующие предосторожности:

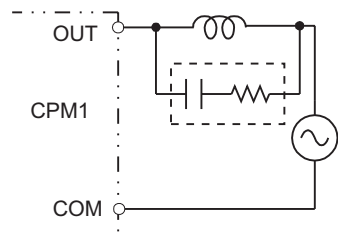
#### Защита от замыканий выходов

Выход или внутренние цепи могут быть повреждены при замыкании подключенной к выходу нагрузки, поэтому рекомендуется устанавливать защитные предохранители на выходы.

#### Индуктивная нагрузка

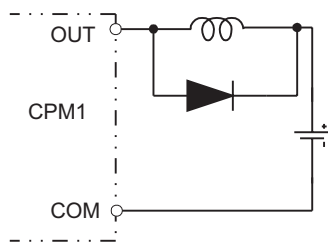
При подключении на выход индуктивной нагрузки параллельно с нагрузкой подключите разрядник защиты от перенапряжений или диод.

Компоненты разрядника защиты от перегрузки должны иметь следующие параметры.



Диод должен удовлетворять следующим условиям:

Напряжение пробоя должно не менее чем в 3 раза превышать номинальное. Средний выпрямляемый ток должен быть 1 А.



### 3.4.5 Соответствие директивам EMC (электромагнитной совместимости)

Каждый CPM1 соответствует стандартам COMMON EMISSION STANDARDS (EN50081-2, организована в июне 1993) по электромагнитной совместимости. Однако помехи, появляющиеся при переключении в 0 или 1 релейных выходов, может не удовлетворять данным стандартам. В таком случае нужно подключить фильтр со стороны нагрузки или принять другие меры вне ПК.

Защитные меры, с помощью которых удовлетворяются требования стандарта, меняются в зависимости от устройств, подключения, конфигурации и т.д. Далее приведен пример защит для уменьшения помех.

#### Меры противодействия

Подробности см. EN50081-2.

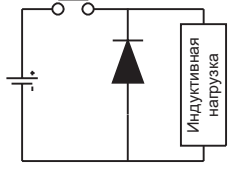
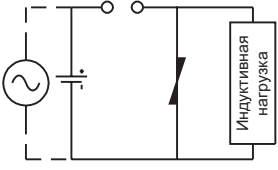
Защита не требуется, когда частота переключения нагрузки не превышает 5 раз в минуту.

Защита требуется, когда частота переключения нагрузки превышает 5 раз в минуту.

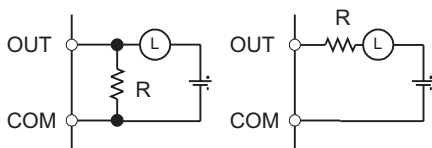
#### Примеры помехозащиты

При переключении индуктивной нагрузки подключите разрядник защиты от перенапряжений, диод и т.д. параллельно нагрузке или контакту, как показано на схеме.

Цепь	Ток		Описание
	Пост.	Перем.	
С RC-цепочкой 	Да	Да	<b>Характеристика</b> Если нагрузкой является реле или соленоид, существует задержка времени между размыканием цепи и сбросом нагрузки. Если питание 24 или 48 В, подключите RC-цепочку параллельно нагрузке. Если напряжение питания 100 ... 200 В, RC-цепочку подключайте между контактами. <b>Требуемый элемент</b> Емкость конденсатора должна быть 1 ... 0.5 мкФ на 1 А тока на контакте и сопротивление резистора должно быть 1 ... 0.5 Ом на 1 В напряжения на контакте. Эти значения, однако, варьируются в зависимости от нагрузки и характеристик реле. Определяйте эти значения экспериментально. Принимайте во внимание, что емкость подавляет искру при размыкании цепи, а резистор ограничивает ток, протекающий в нагрузке, когда цепь снова замыкается. Электрическая прочность конденсатора должна быть 200 ... 300 В. Если подключается цепь переменного тока, используйте конденсатор не полярный.

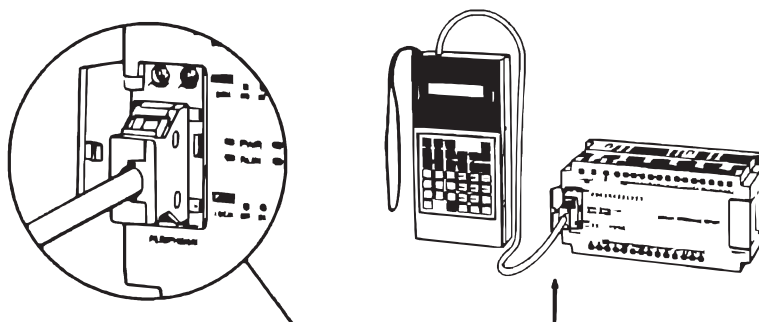
Цепь	Ток		Описание
	Пост.	Перем.	
<p>С диодом</p> 	Нет	Да	<p><b>Характеристика</b></p> <p>Диод, подключенный параллельно нагрузке, превращает энергию, накопленную катушкой, в ток, который затем течет по катушке и превращается в тепло (Джоули) на сопротивлении индуктивной нагрузки.</p> <p>Этот временной интервал между размыканием цепи и моментом сброса нагрузки, вызванный данным методом, дольше, чем вызванный методом разрядника.</p> <p><b>Требуемый элемент</b></p> <p>Обратное напряжение диода должно быть как минимум в 10 раз больше напряжения цепи.</p> <p>Прямой ток диода должен быть не менее тока нагрузки.</p> <p>Обратное напряжение диода может быть больше напряжения питания в два-три раза, если в цепях с низким напряжением применен разрядник.</p>
<p>С варистором</p> 	Да	Да	<p><b>Характеристика</b></p> <p>Метод варистора предотвращает появление высокого напряжения между контактами благодаря характеристике постоянного напряжения варистора. Существует задержка времени между размыканием цепи и сбросом нагрузки.</p> <p>Если питание 24 или 48 В, подключите варистор параллельно нагрузке. Если напряжение питания 100 ... 200 В, варистор подключайте между контактами</p>

При переключении нагрузки с большим пусковым током, такой, как лампа накаливания, подавляйте пусковой ток следующим образом:



### 3.4.6 Подключение периферийного устройства

CPM1 можно подключить к программатору C200H-PRO27-E стандартным C200H-CN222 (2м) или C200H-CN422 (4м) соединительным кабелем. CPM1 можно также подключить к CQM1-PRO01-E. CQM1-PRO01-E поставляется с 2-м кабелем.

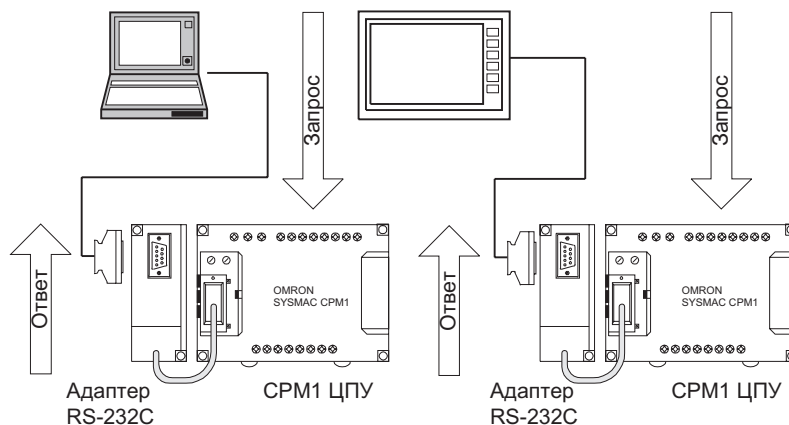


### 3.4.7 Связь HOST LINK

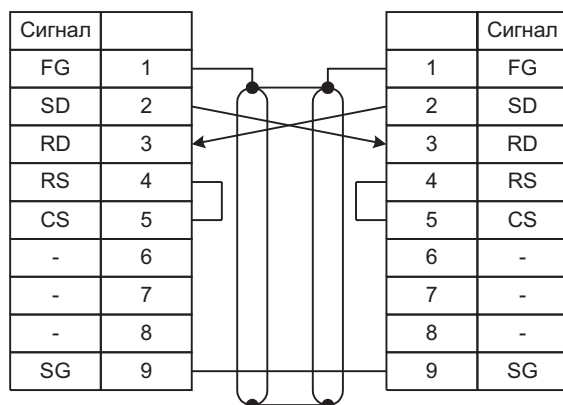
HOST LINK- это система связи в режиме запрос/ответ, в которой команды передаются с управляющего компьютера и соответствующие ответы возвращаются с указанных ПК. Запросы HOST LINK используются для чтения/записи данных в области данных ПК и чтения/записи установочных параметров.

#### HOST LINK 1:1

CPM1 можно подключить к IBM PC/AT совместимому компьютеру или программируемому терминалу с адаптером RS-232C, как показано на схеме.



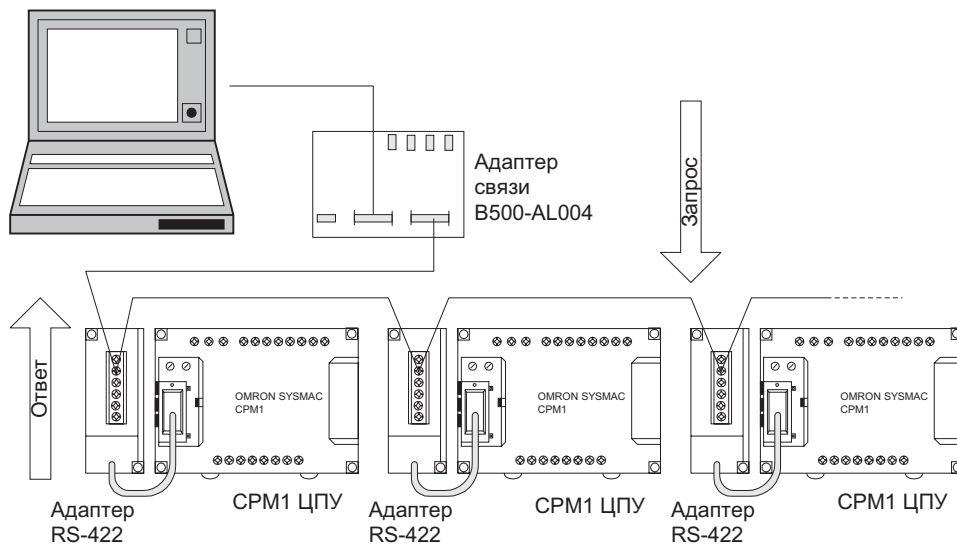
На схеме представлена распиновка кабеля RS-232C, служащего для подключения CPM1 к управляющему компьютеру или программируемому терминалу.



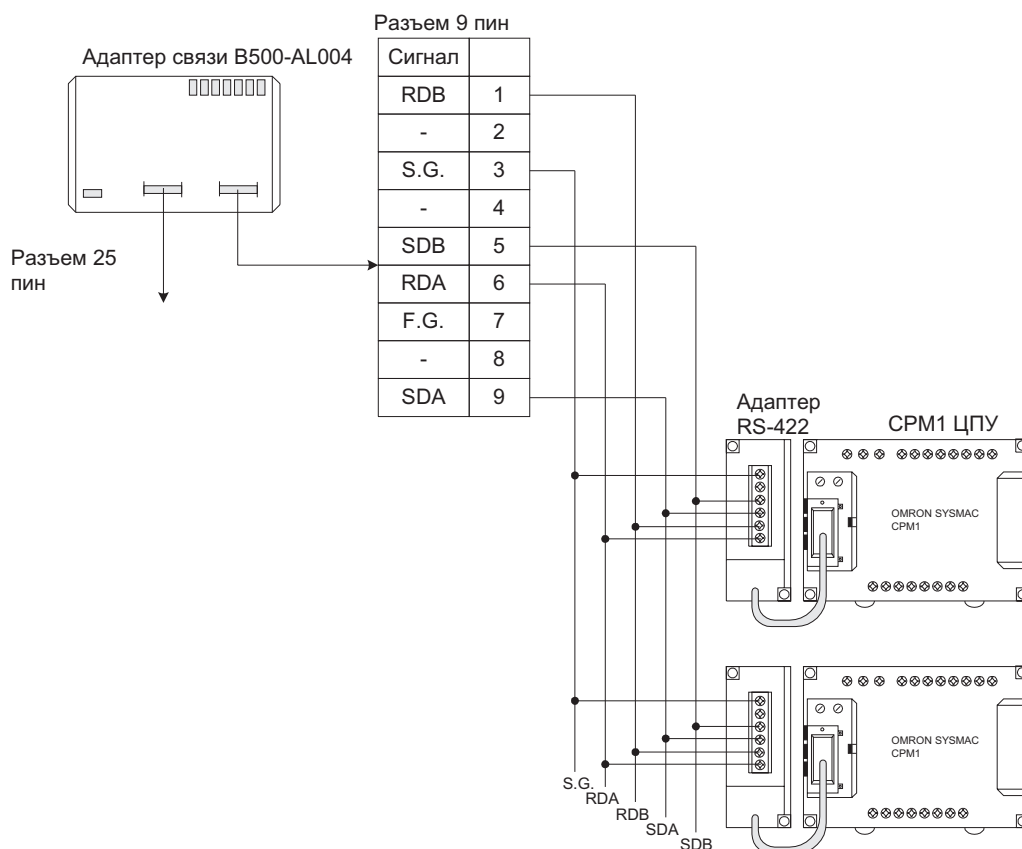
*Замечание* Когда CPM1 подключен к управляющему компьютеру, установите переключатель режимов на адаптере RS-232C в положение HOST.

#### HOST LINK 1:N

До 32 CPM1 и других ПК OMRON можно подключить к IBM PC/AT совместимому компьютеру или программируемому терминалу с помощью адаптера B500-AL004 и RS-422, как показано на схеме.

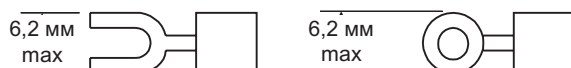


На схеме представлено подключение кабеля RS-422, служащего для соединения CPM1 с адаптером связи B500-AL004. В соединениях адаптер связи/адаптер RS-422 и адаптер RS-422/адаптер RS-422 соедините клеммы SG, RDA, RDB, SDA и SDB с клеммами другого адаптера.



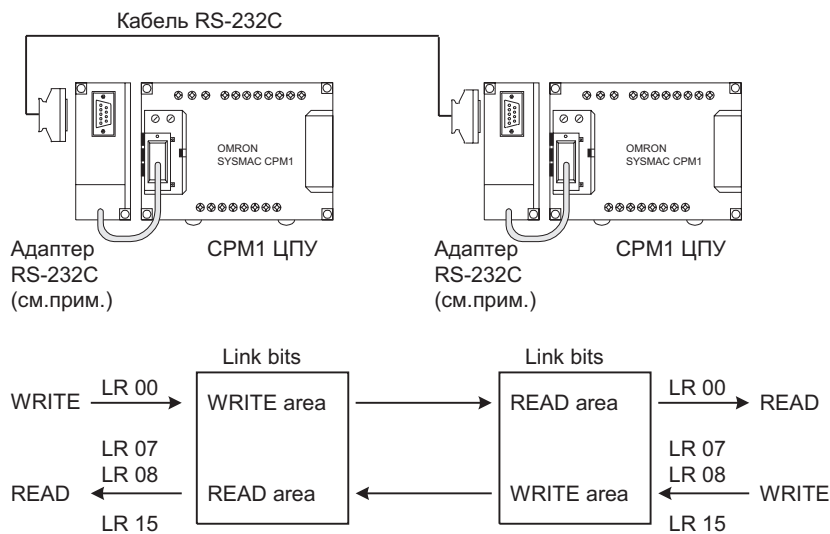
- Замечание*
1. Максимальная длина кабеля RS-422 должна быть 500 м.
  2. Установите переключатель сопротивления в ON (вверх) для адаптеров связи на обоих концах системы HOST LINK и для обоих адаптеров RS-422.

Всегда используйте напрессованные наконечники при подключении адаптеров RS-422. Используйте винты M3 и надежно их зажмите (0.5 ... 0.6 Нм).



### 3.4.8 Связь ПК "1:1"

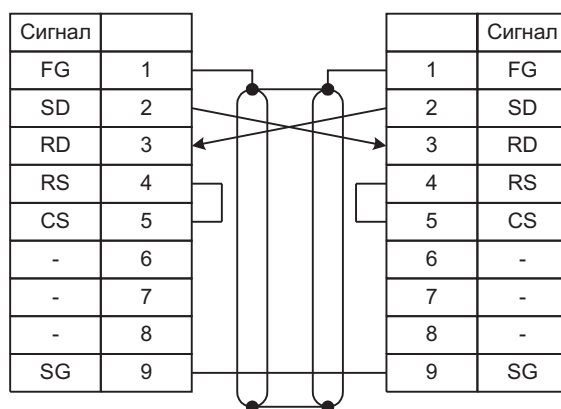
CPM1 можно связать с другим CPM1, CQM1 или C200HS через адаптер RS-232C. Один из ПК служит в качестве ведущего, другой - в качестве ведомого и у них общая область памяти до 256 бит (LR 0000 ... LR 1515).



**Замечание** Связь 1:1 можно использовать только при подключенном адаптере RS-232C (CPM1-CIF01).

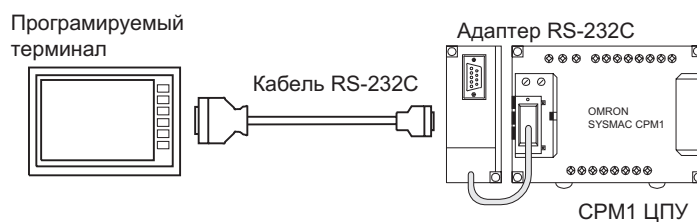
Переключатель DIP на адаптере RS-232C (CPM1-CIF01) установите в NT (вниз).

На следующей схеме показана разводка кабеля RS-232C, который используется для связи CPM1 с другим ПК.



### 3.4.9 Связь NT LINK

Высокоскоростную связь можно осуществлять путем прямого доступа по линии связи NT LINK между CPM1 и программируемым терминалом.



**Замечание** NT LINK можно осуществлять только при подключенном адаптере RS-232C (CPM1-CIF01).

Переключатель DIP на адаптере RS-232C (CPM1-CIF01) установите в NT (вниз).

На следующей схеме показана разводка кабеля RS-232C, который используется для связи СРМ1 с программируемым терминалом.



## **Глава 4. Использование периферийных устройств**

*В данной главе дано описание работы с программатором.*

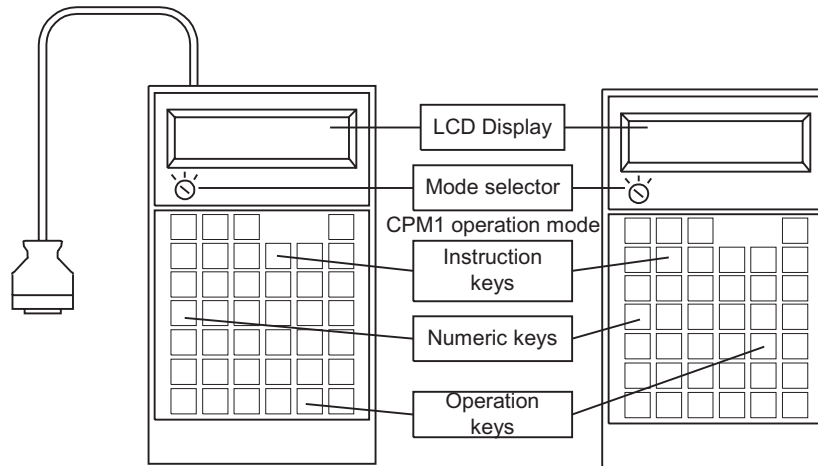
## 4.1 Использование программатора

В данной главе приведена информация о подключении и использовании программатора. Подробности об ошибках, которые могут появиться при работе с программатором, см. 5-4.

### 4.1.1 Совместимые программаторы

При работе с CPM1 можно использовать 2 типа программаторов: CQM1-PRO01-E и C200H-PRO27-E. Назначения клавиш у этих программаторов аналогичны.

Для ввода буквы, указанной в левом верхнем углу клавиши, или верхней функции клавиши с 2 функциями нажмите и держите клавишу SHIFT. Например, клавиша AR/HR на CQM1-PRO01 может определять области AR и HR. Для работы с областью AR следует нажать и отпустить клавишу SHIFT, а затем нажать AR.

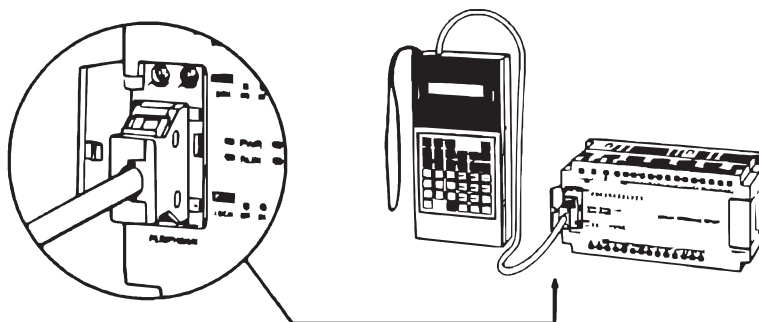


На программаторах CQM1-PRO01-E и C200H-PRO27-E три клавиши имеют разное обозначение. Их функции одинаковы.

Клавиши CQM1-PRO01-E	Клавиши C200H-PRO27-E
AR HR	HR
SET	PLAY SET
RESET	REC RESET

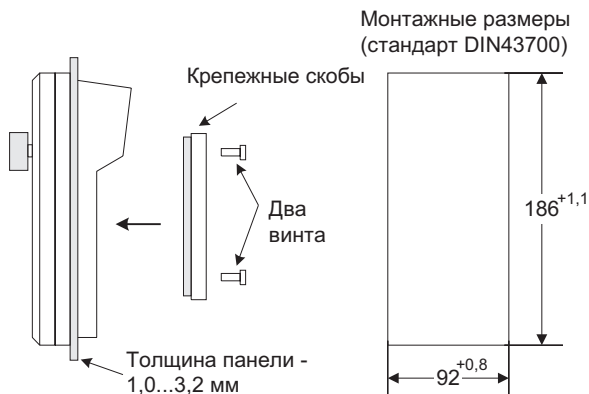
### 4.1.2 Подключение программатора

Подключите соединительный кабель программатора к периферийному порту CPM1 в соответствии с рисунком.

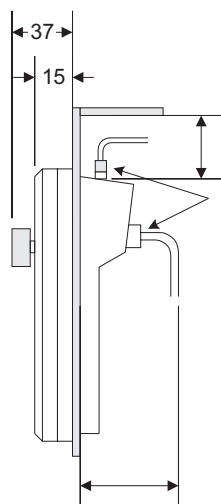


### Установка на панели

Программатор С200Н-PRO27-Е можно устанавливать на панели управления, как показано на рисунку. (Монтажные скобы С200Н-АТТ01 продаются отдельно).



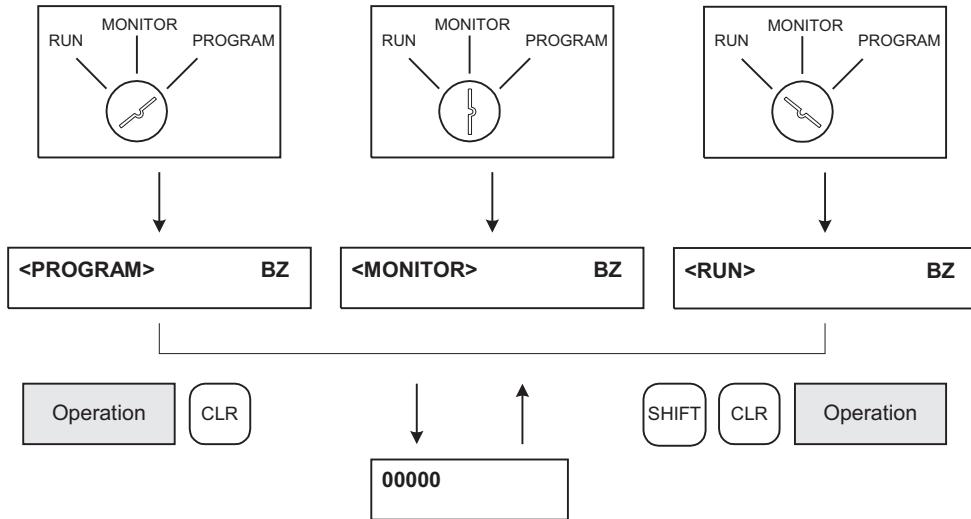
Оставьте свободными не менее 80 мм над программатором для разъема кабеля.



### 4.1.3 Переключение режимов СРМ1

После подключения программатора для изменения режимов работы СРМ1 служит переключатель режимов. На дисплее программатора появляется режим (PROGRAM, MONITOR или RUN).

- Когда на дисплее высвечивается режим работы, операции с клавишами производить нельзя. Для разрешения работы с клавиатурой нажмите CLR для очистки индикации режима работы.
- Если при переключении режимов переключателем нажата клавиша SHIFT, текущая индикация останется на экране программатора и индикация режима не появится.
- Если периферийное устройство, такое как программатор, не подключено, при включении СРМ1 автоматически устанавливается в режим RUN.



**Режим PROGRAM**

В Режиме PROGRAM программа не выполняется. Режим PROGRAM используется для написания и редактирования программ, очистки памяти или проверки программы на синтаксические ошибки.

**Режим MONITOR**

В Режиме MONITOR программа выполняется и входы/ выходы обрабатываются аналогично режиму RUN. Режим MONITOR используется для проверки системы путем просмотра состояния СРМ1, для принудительного включения/сброса битов входа/выхода, изменения заданного и текущего значения таймеров и счетчиков и т. п.

**Режим RUN**

Рабочий режим ПК. Можно просматривать состояние СРМ1 с программатора, но нельзя принудительно включить/сбросить биты входа/выхода, изменять заданное и текущее значение таймеров и счетчиков.

*Внимание!* При первом запуске программы тщательно проверьте систему для предотвращения аварийных ситуаций.

**4.2 Операции программатора**

**4.2.1 Очистка памяти**

Данная операция применяется для очистки всей или части памяти программ и любой области памяти параметров (за исключением тех участков, которые только читаются), а также содержимого памяти программатора. Данная операция возможна только в режиме PROGRAM.

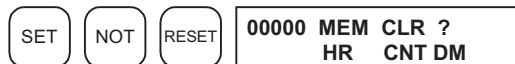
RUN	MONITOR	PROGRAM
НЕТ	НЕТ	ДА

Перед первым программированием или введении новой программы очистите все области памяти.

**Очистка всей памяти**

Для полной очистки памяти применяется следующая последовательности действий:

- 1, 2, 3,... 1. Вызовите исходную индикацию, нажав несколько раз клавишу CLR.
2. Нажмите клавиши SET, NOT, затем RESET чтобы начать операцию.



3. Нажать клавишу MONTR для полной очистки памяти.



**Внимание!** При выполнении данной операции будут стерты установочные параметры (DM 6600 ... DM 6655).

#### Частичная очистка

Имеется возможность сохранить параметры в заданных областях или часть памяти программ. Для сохранения параметров в областях HR, TC или DM нажмите соответствующую клавишу после SET, NOT, RESET. При нажатии MONTR будет очищена та область параметров, которая осталась индикатироваться.

Клавиша HR служит для указания двух областей AR и HR;

клавиша CNT - для областей таймеров и счетчиков;

клавиша DM - для области DM.

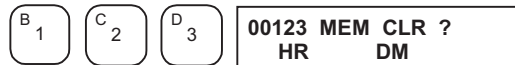
Также возможно сохранение части памяти программ от первого адреса до указанного. После задания областей параметров, подлежащих сохранению, укажите первый адрес памяти программ, подлежащий очистке. Например, введите 030 чтобы оставить в сохранности адреса 000 ... 029, но очистить память с адреса 030 до конца области.

В качестве примера выполните следующие действия для сохранения области таймеров-счетчиков и памяти программ с адресами 000 ... 122:

- 1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации.
2. Нажать клавиши SET, NOT, затем RESET чтобы начать операцию.
3. Нажать клавишу CNT для стирания области таймеров/счетчиков с дисплея.



4. Набрать 123 для задания стартового адреса программы.



5. Нажать клавишу MONTR для очистки заданных участков памяти.



#### 4.2.2 Чтение/очистка сообщений об ошибках

Данная операция применяется для индикации и очистки сообщений об ошибках. Индикация и стирание сообщений о нефатальных ошибках и сообщений, вызванной командой MESSAGE, возможны в любом режиме, но сообщения о фатальных ошибках можно индикатировать и стирать только в режиме PROGRAM.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

Перед вводом новой программы должны быть убраны все признаки неисправностей, сохраненные в памяти. Предполагается, что все причины неисправностей, вызвавших появление сообщений, устранены. Если при попытке стирания сообщения о неисправности звучит зуммер, сначала следует устранить причину и только потом стирать сообщение (Информацию о поиске неисправностей См. гл. 5).

#### Последовательность нажатия клавиш

Для индикации и очистки сообщений нужно выполнить следующие действия:

- 1, 2, 3,... 1. Вызвать исходную индикацию.
2. Нажать клавиши FUN, затем MONTR чтобы начать операцию. Если сообщений нет, появится следующая индикация:



Если сообщения имеются, при нажатии клавиши MONTR будут появляться сообщения в порядке их важности. Нажатие клавиши MONTR убирает текущее сообщение и вызывает следующее по важности. Продолжайте нажимать MONTR до очистки всех сообщений. Ниже приведены примеры сообщений об ошибках:

Ошибка памяти:



Системная ошибка:



Сообщение:



Все сообщения очищены:



### 4.3.3 Операции с зуммером

Данная операция применяется для включения или отключения зуммера, который звучит при нажатии клавиши на программаторе. Зуммер также звучит при появлении ошибки во время работы ПК. Действия, рассматриваемые в данной главе, не влияют на поведение зуммера при ошибках.

Данная операция возможна в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

#### Последовательность нажатия клавиш

Для включения и отключения зуммера выполните следующие действия:

- 1, 2, 3,... 1. Нажать клавиши CLR, SHIFT и CLR для индикации режима. В описанном случае ПК находится в режиме ПРОГРАММИРОВАНИЕ и зуммер включен.



2. Нажмите клавиши SHIFT и 1 для выключения зуммера.



3. Снова нажмите клавиши SHIFT и 1 для выключения зуммера.



### 4.2.4 Установка и чтение адреса памяти программ

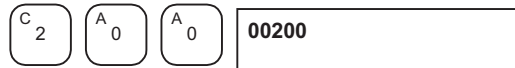
Данная операция применяется для индикации указанного адреса памяти программ в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

При первом вводе программы в памяти записан стартовый адрес 000. Поскольку данный адрес появляется при очистке дисплея, нет необходимости задавать его.

При вводе программы в адрес, отличный от 000, либо читать или менять уже существующую программу, необходимо задавать требуемый адрес.

- 1, 2, 3,... 1. Вызовите исходную индикацию клавишей CLR.
2. Введите требуемый адрес. Ведущие нули набирать необязательно.



3. Нажмите клавишу стрелка вниз.



*Замечание* В режиме RUN , MONITOR будет индигатироваться состояние бита OFF(0) или ON (1).

4. Нажимайте клавиши стрелка вверх и Стрелка вниз для просмотра программы.



#### 4.2.5 Поиск команды

Данная операция применяется для поиска указанной команды в программе и возможна в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

В режиме RUN , MONITOR будет индигатироваться состояние бита OFF(0) или ON(1).

- 1, 2, 3,... 1. Вызовите исходную индикацию клавишей CLR.
2. Введите адрес, с которого нужно начать поиск и нажмите клавишу Стрелка вниз. Ведущие нули вводить необязательно.

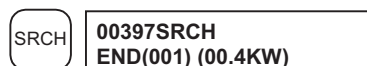


3. Введите команду, которую надо найти, и нажмите клавишу SRCH. В данном примере идет поиск команды LD. В данном примере следующая команда LD встречается по адресу 200, как показано ниже.



4. Нажмите Стрелка вниз для индикации операндов команды или SRCH для продолжения поиска данной команды.

5. Поиск будет продолжаться до команды END или до достижения границы области программ. В данном примере встретилась команда END.



#### 4.2.6 Поиск битового операнда

Данная операция применяется для поиска указанного битового операнда программы и возможна в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

В режиме RUN , MONITOR будет индигатироваться состояние бита.

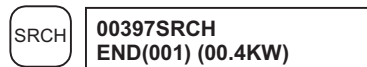
- 1, 2, 3,... 1. Вызовите исходную индикацию клавишей CLR.
2. Введите адрес операнда. Ведущие нули вводить необязательно.



3. Нажмите клавишу SRCH для начала поиска.



4. Нажмите SRCH для продолжения поиска данного битового операнда.
5. Поиск будет продолжаться до команды END или до достижения границы области программ. Например, встретилась команда END.



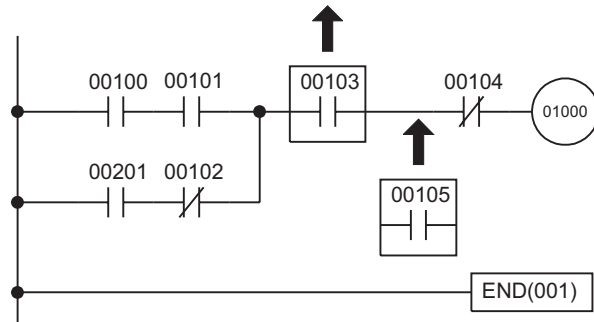
#### 4.2.7 Вставка и удаление команд

Данная операция применяется для вставки и удаления команд из программы. Данная операция возможна только в режиме PROGRAM.

RUN	MONITOR	PROGRAM
НЕТ	НЕТ	ДА

Для демонстрации данной операции вставим команду IR 00105 NO (нормально открытый) по адресу 0206 и удалим команду IR 00103 NO по адресу 0205. Данная операция показана на рисунке.

#### Исходная программа



Адрес	Команда	Операнд
.	.	.
.	.	.
.	.	.
.	.	.
00205	AND	00103
00206	AND NOT	00104
00207	OUT	10000
00208	END (01)	

#### Вставка

Для вставки команды IR 00105 NO по адресу 0206 проделайте следующую последовательность действий:



- 1, 2, 3,... 1. Вызовите исходную индикацию клавишей CLR.  
 2. Введите адрес, по которому будет вставлена команда, и нажмите клавишу Стрелка вниз. Ведущие нули вводить необязательно.



3. Введите новую команду и нажмите INS.



4. Нажмите клавишу Стрелка вниз для вставки новой команды.



*Замечание* Для команд, требующих несколько операндов, введите операнды и затем клавишу WRITE.

### Удаление

Для удаления команды IR 00103 NO по адресу 0205 сделайте следующую последовательность действий:

- 1, 2, 3,... 1. Нажмите клавишу CLR для вызова исходной индикации.  
 2. Введите адрес, по которому будет удалено условие NO и нажмите клавишу Стрелка вниз. Ведущие нули вводить необязательно.



3. Нажмите клавишу DEL.

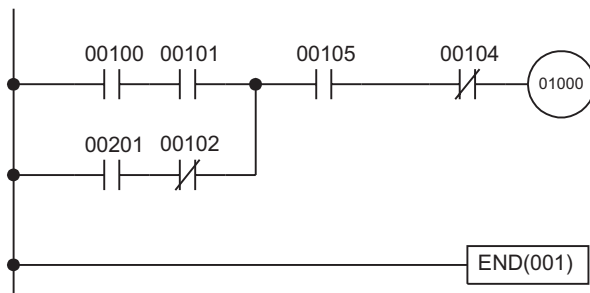


4. Нажмите клавишу Стрелка вверх для удаления указанной команды. Если у команды есть несколько операндов, они удалятся автоматически.



После завершения вставки и удаления клавишами Стрелка вверх и Стрелка вниз просмотрите программу и проверьте, что изменения произведены правильно, в соответствии со следующей схемой:

### Исправленная программа



Адрес	Команда	Операнды
-	-	-
00205	AND	00105
00206	AND NOT	00104
00207	OUT	10000
00208	END (01)	

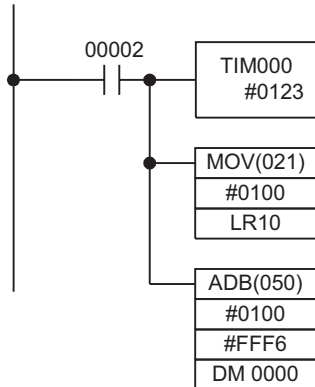
### 4.2.8 Ввод или редактирование программ

Данная операция применяется для ввода или редактирования программ. Ее можно проводить только в режиме PROGRAM.

RUN	MONITOR	PROGRAM
НЕТ	НЕТ	ДА

Одна и та же процедура используется как для ввода программы в первый раз, так и для изменения уже существующей программы. В обоих случаях содержимое памяти программ переписывается.

Для демонстрации работы с данной функцией введем программу, представленную на рисунке.



Адрес	Команда	Операнд
00200	LD	IR 00002
00201	TIM	000
		0123
00202	MOV(21)	#0100
		LR 10
00203	ADB(50)	# 100
		# FFF6
		DM 0000

- 1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации.
2. Определите адрес, с которого будет начинаться программа.
3. Введите адрес, с которого будет начинаться программа, и нажмите Стрелку вниз. Это необходимо для ввода ведущих нулей.



4. Введите первую команду и операнд.



5. Нажмите клавишу WRITE для записи данной команды в память программ. Будет индикативаться следующий адрес программы.



Если Вы ошиблись при вводе, нажмите клавишу СТРЕЛКА ВВЕРХ для возврата на предыдущий адрес программы и снова введите команду. Ошибочная команда будет переписана.

6. Введите вторую команду и операнд. ( В нашем случае нет необходимости вводить номер таймера: поскольку он = 000 ). Нажмите клавишу WRITE для записи данной команды в память программ.



7. Введите второй операнд (123 для задания 12.3 сек.) и нажмите WRITE. Появится следующий адрес программы.



Если при вводе была допущена ошибка, нажмите клавишу СТРЕЛКА ВВЕРХ для возврата на индикацию ошибочного операнда, нажмите клавишу CONT/# и снова 123. Вместо ошибочного значения запишется 123.

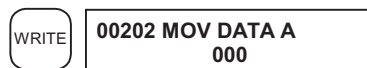
*Замечание* Счетчики вводятся так же, как и таймеры, только вместо TIM надо нажимать CNT.

8. Введите третью команду. Сначала нажмите FUN для указания того, что будет вводиться функция, затем код функции (в данном примере 21)



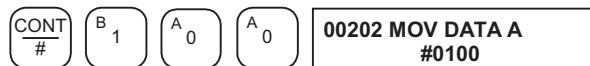
*Замечание* Для того, чтобы изменить команду на команду фронта 0/1 после ввода кода функции нажмите клавишу NOT. Рядом с требуемой командой будет индицироваться @. Для превращения ее в команду не фронта 0/1 снова нажмите NOT. Символ @ исчезнет. Для изменения команд после того, как они введены, пролистайте программу, пока не высветится нужная команда, и нажмите NOT. Рядом с командой появится символ @.

9. Нажмите WRITE для записи команды в память. Появится индикация для ввода первого операнда.

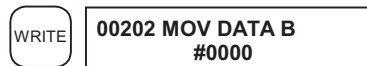


#### Запись 16-ричной, двоично-десятичной константы.

10. Введите первый операнд.

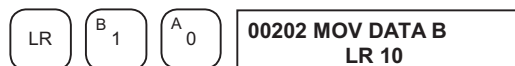


Нажмите WRITE для ввода команды в память. Появится индикация для ввода второго операнда.



#### Запись адреса слова.

11. Введите второй операнд.



Нажмите WRITE для ввода команды в память. Появится индикация для ввода следующей команды.



*Замечание* Если вводится операнд команды, назначение бита или слова можно опустить.

12. Введите следующую команду.



Нажмите WRITE для ввода команды в память.

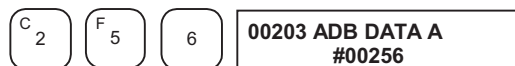


#### Запись десятичного числа без знака

13. Возможно введение в виде десятичного числа без знака.



Введите операнд в диапазоне 0 ... 65535.



*Замечание* Если при вводе была допущена ошибка, нажмите CLR для возврата состояния перед вводом и введите исправленное значение.

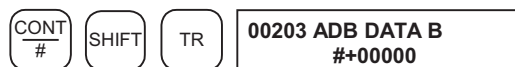
14. Возвратитесь в индикацию ввода 16-ричного числа.



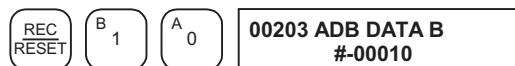
*Замечание* Если был сделан ввод числа вне допустимого диапазона, звучит зуммер и перехода к индикации 16-ричного числа не происходит.



15. Второй операнд вводится как число со знаком.

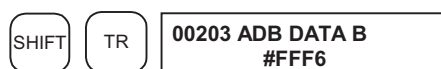


Введите операнд в диапазоне -32 768 ... + 32 767. Для ввода положительного значения служит клавиша SET, отрицательного - RESET.



*Замечание* Если при вводе была допущена ошибка, нажмите CLR для возврата состояния перед вводом и введите исправленное значение.

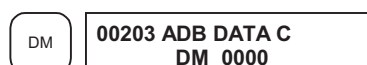
16. Возвратитесь в индикацию 16-ричного числа.



*Замечание* Если был сделан ввод числа вне допустимого диапазона, звучит зуммер и перехода к индикации 16-ричного числа не происходит.



17. Введите последний операнд и нажмите клавишу WRITE.



#### 4.2.9 Контроль программы

Данная операция проверяет ошибки программирования и при обнаружении ошибки индицирует адрес программы и ошибку. Возможна только в режиме PROGRAM.

RUN	MONITOR	PROGRAM
НЕТ	НЕТ	ДА

- 1, 2, 3,...
1. Нажмите клавишу CLR для вызова исходной индикации.
  2. Нажмите клавишу SRCH. Появится строка подсказки, требующая ввести желаемый уровень проверки.

SRCH	00000PROG CHK CHKLEVEL (0-2)?
------	----------------------------------

3. Ввести желаемый уровень проверки (0, 1 или 2). Контроль программы начнется после ввода уровня, и первая обнаруженная будет выведена на индикацию.

A 0	00178CIRCUIT ERR OUT 00200
-----	-------------------------------

*Замечание* Подробности об уровнях контроля см. 5.5 Ошибки программирования.

4. Нажмите клавишу SRCH для продолжения поиска. Будет индигатироваться следующая ошибка. Для продолжения поиска ошибок продолжайте нажимать клавишу SEARCH.

Поиск будет продолжиться, пока не встретится команда END или не будет достигнута граница памяти программ. Если достигнута граница памяти, индикация будет иметь вид:

SRCH	00300NO END INST END
------	-------------------------

Если встретилась команда END, индикация будет иметь вид:

SRCH	00310PROG CHK END(001) (00.3KW)
------	------------------------------------

Если появилась ошибка, отредактируйте программу для исправления ошибки и снова проверьте программу. Продолжайте проверку до тех пор, пока не откорректируете все ошибки.

#### 4.2.10 Просмотр бита, числа, слова

Данная операция применяется для индикации состояния до 16 бит и слов, хотя только 3 слова могут быть отображены на дисплее одновременно. Данная операция возможна в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

##### Чтение программы, затем просмотр

Когда индигатируется адрес программы, состояние бита или слова по этому адресу можно просмотреть, нажав клавишу MONTR.

- 1, 2, 3,...
1. Нажмите клавишу CLR для вызова исходной индикации.
  2. Введите требуемый адрес программы и нажмите Стрелку вниз.

C 2	A 0	A 0	↓	00200READ TIM 000
-----	-----	-----	---	----------------------

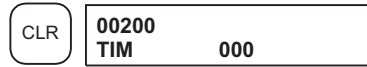
3. Нажмите MONTR для начала просмотра.

MONTR	T000 1234
-------	--------------

Если просматривается состояние бита, его состояние можно изменить операцией Принудительное включение/выключение. Подробности см. гл. 4-3-22. Если просматривается состояние слова, его значение можно изменить

операцией Модификация 16-ричного/двоично-десятичного параметра. Подробности см. гл. 4-3-18.

4. Нажмите CLR для окончания просмотра.



### Просмотр бита

Для просмотра состояния отдельного бит проделайте следующую последовательность действий:

- 1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации.
2. Введите требуемый битовый адрес и нажмите MONTR.



При нажатии Стрелка вверх и Стрелка вниз индицируется состояние предыдущего или следующего битов.

Состояние просматриваемого бита можно изменить операцией Принудительное включение/выключение. Подробности см. гл. 4-3-22.

3. Нажмите CLR для окончания просмотра.



### Просмотр слова

Для просмотра состояния отдельного слова проделайте следующую последовательность действий:

- 1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации
2. Введите адрес просматриваемого слова.



3. Для начала просмотра нажмите MONTR.



При нажатии Стрелка вверх и Стрелка вниз индицируется состояние предыдущего или следующего битов.

Значение просматриваемого слова можно изменить операцией Модификация 16-ричного/двоично-десятичного параметра. Подробности см. гл. 4-3-18.

4. Нажмите CLR для окончания просмотра.



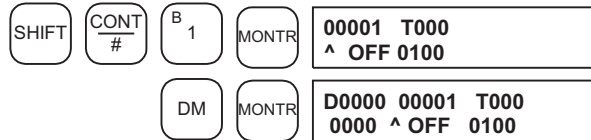
### Просмотр нескольких адресов

Можно одновременно просматривать состояния до 6 бит и слов, хотя только 3 могут быть отображены на дисплее одновременно.

- 1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации.
2. Введите адрес первого бита или слова и нажмите MONTR.



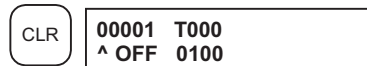
3. Повторите шаг 2 (не более 6 раз) для задания следующих адресов, которые будут наблюдаться.



Если просматриваются 4 и более слов, биты и слова, которых нет на дисплее, вызываются нажатием MONTR, Если нажать только MONTR, изображение сдвинется вправо.

Если ввести более 6 адресов, введенные ранее будут отменены.

4. Нажмите CLR для окончания просмотра самого левого бита и убирания его с дисплея.



5. Нажмите SHIFT + CLR для полного окончания просмотра



*Замечание* Нажмите SHIFT + CLR для возврата в индикацию нескольких адресов. Нажмите SHIFT+MONTR для индикации сохранившегося состояния индикации нескольких адресов. Память состояния индикации нескольких слов может содержать до 6 бит и слов.

#### 4.2.11 Просмотр изменения состояния бита

Данная операция применяется для просмотра изменения состояния отдельного бита с 1/0 и 0/1. При обнаружении изменения состояния это состояние индицируется и звенит зуммер. Данная операция возможна в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

1, 2, 3,... 1. Вызовите просмотр требуемого бита согласно п. 4.3.10. Если индицируются 2 и более бита, требуемый бит должен находиться слева.

В примере будет контролироваться изменение состояния бита LR00.



2. Для просмотра перехода 0/1 нажмите SHIFT и Стрелка вверх. Появится символ U@.



Для просмотра перехода 1/0 нажмите SHIFT и Стрелка вниз. Появится символ D@.



3. При переходе бита 0/1 (задание U@) и 1/0 (задание D@) звучит зуммер.



4. Для окончания индикации переходов и возврата в нормальную индикацию нажмите CLR.



### 4.2.12 Просмотр параметра в двоичном виде

Данная операция применяется для индикации состояния 1 или 0 любого бита. Данная операция возможна в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

- 1, 2, 3,... 1. Вызовите просмотр требуемого слова согласно п.4.3.10. Требуемое слово на дисплее должно быть самым левым, если индицируется 2 и более слов.

```
C000
0000
```

(Индикация слова)

2. Нажмите SHIFT, MONTR для начала двоичного просмотра. Состояние 16 битов выбранного слова высветится в нижней части дисплея. (1 - ON, 0- OFF).

```
SHIFT MONTR C000 MONTR
0000000000000000
```

Состояние принудительно установленных битов индицируется как S, принудительно сброшенных как R.

```
C000 MONTR
000S000000000R0000
```

- Замечание* а. Значение просматриваемых битов можно изменять согласно 4-3-19.  
 б. При нажатии Стрелка вверх и Стрелка вниз можно просматривать состояние битов предыдущего или следующего слов.
3. Для окончания просмотра в двоичном виде и возврата в нормальную индикацию нажмите CLR.

```
CLR C000
0000
```

### 4.2.13 Просмотр трех слов

Данная операция применяется для индикации состояния трех последовательных слов. Данная операция возможна в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

- 1, 2, 3,... 1. Вызовите просмотр первого из трех требуемых слов согласно п. 4-3-10. Если индицируется 2 и более слов, требуемое слово на дисплее должно быть самым левым.

```
C000
89AB
```

(Просмотр слова)

2. Нажмите EXT для начала просмотра трех слов. Состояние требуемого слова и следующих двух высветится, как приведено на рисунке. В примере выбрано слово DM 0000.

```
EXT C002 C001 C000
0123 4567 89AB
```

Для изменения адресов просматриваемых слов нажмите Стрелка вверх и Стрелка вниз.



Значение индицируемых слов можно изменять согласно 4-3-16.

3. Нажмите CLR для окончания просмотра в виде 3 слов и возвратиться в нормальный режим. Будет просматриваться самое правое слово.

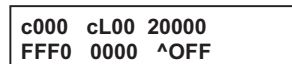


#### 4.2.14 Просмотр в виде десятичного числа со знаком

Данная операция преобразует содержимое указанного слова из 16-ричного со знаком (формат дополнения до двух) в десятичное число со знаком для индикации. Данная операция может проводиться при просмотре входов/выходов, просмотре нескольких адресов или просмотре трех слов.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

1, 2, 3,... 1. Вызовите просмотр требуемого слова. При индикации нескольких слов преобразуется самое левое слово.

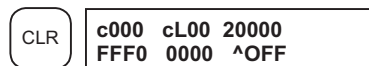


2. Нажмите SHIFT+TR для индикации самого левого слова в виде десятичного со знаком.



Здесь содержимое индицируемого слова можно изменить вводом десятичного числа со знаком. См. 4.3.20.

3. Нажмите клавишу CLR или SHIFT+TR для окончания индикации десятичного со знаком и возвратиться в нормальную индикацию.

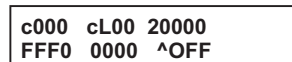


#### 4.2.15 Просмотр в виде десятичного числа без знака

Данная операция преобразует содержимое указанного слова из 16-ричного содержимого слова в десятичное число со знаком для индикации. Данная операция может проводиться при просмотре входов/выходов, просмотре нескольких адресов или просмотре трех слов.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

1, 2, 3,... 1. Вызовите просмотр требуемого слова. При индикации нескольких слов преобразуется самое левое слово.



2. Нажмите SHIFT+TR+NOT для индикации самого левого слова в виде десятичного без знака.



Здесь содержимое индицируемого слова можно изменить вводом десятичного числа без знака. См. 4.3.21.

3. Нажмите клавишу CLR или SHIFT+TR для окончания индикации десятичного без знака и возвратиться в нормальную индикацию.

CLR	c000 cL00 20000 FFF0 0000 ^OFF
-----	-----------------------------------

#### 4.2.16 Изменение значений трех соседних слов

Данная операция применяется для изменения содержимого одного или более из трех последовательных слов, (индицируемых в режиме индикации 3 слов). Данная операция возможна только в режимах MONITOR и PROGRAM.

RUN	MONITOR	PROGRAM
НЕТ	ДА	ДА

- 1, 2, 3,... 1. Вызовите просмотр состояния требуемых слов по алгоритму 4-3-13.

D0002 D0001 D0000 0123 4567 89AB
-------------------------------------

(Индикация трех слов)

2. Нажмите клавишу CHG для начала операции. Курсор появится рядом с содержимым левого слова.

CHG	D0002 3CHCHANG? □123 4567 89AB
-----	-----------------------------------

3. Введите новое значение для левого слова и нажмите клавишу CHG, если нужно изменить более 1 слова. Если больше изменений не требуется, нажмите WRITE для внесения изменений в память.

<sup>B</sup> 1	CHG	D0002 3CHCHANG? 0123 □4567 89AB
----------------	-----	------------------------------------

4. Введите новое значение для среднего слова на дисплее и нажмите клавишу CHG, если нужно изменить правое слово. Если больше изменений не требуется, нажмите WRITE для внесения изменений в память ( в данном примере не требуется).

<sup>C</sup> 2	<sup>D</sup> 3	<sup>E</sup> 4	WRITE	D0002 D0001 D0000 0001 0234 89AB
----------------	----------------	----------------	-------	-------------------------------------

*Замечание* Если нажать CLR перед WRITE, операция будет отменена и возвратится индикация 3 слов без изменений в памяти.

#### 4.2.17 Изменение задания таймеров и счетчиков

Для изменения задания таймеров и счетчиков имеется 2 операции, которые возможны в режимах MONITOR и PROGRAM. В режиме MONITOR задание можно менять во время исполнения программы.

RUN	MONITOR	PROGRAM
НЕТ	ДА	ДА

Задания таймеров и счетчиков можно изменять либо вводом нового значения, либо инкрементом/декрементом текущего значения.

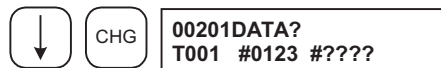
##### Ввод нового задания

Данная операция может применяться для ввода новой константы задания или изменения источника задания с константы на адрес и наоборот. В примере показано, как вводить новую константу и изменять источник задания с константы на адрес.

- 1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации.  
2. Вызовите на индикацию требуемый таймер или счетчик.

TIM	<sup>B</sup> 1	SRCH	00201SRCH TIM 001
-----	----------------	------	----------------------

3. Нажмите Стрелка вниз, затем CHG.



4. В этом месте можно вводить новую константу или менять источник задания с константы на адрес.

а. Для ввода новой константы введите число и нажмите WRITE.



б. Для замены источника задания с константы на адрес введите адрес и нажмите WRITE.



### Инкремент и декремент константы задания

Данная операция может применяться для инкремента и декремента задания. Это возможно только когда задание задается в виде константы.

- 1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации.
2. Вызовите на индикацию требуемый таймер или счетчик.



3. Нажмите Стрелка вниз, CHG и EXT.

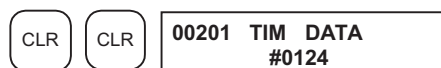


Константа слева - старое задание, число справа будет введено как новое задание в п. 5.

4. Нажмите Стрелка вниз или Стрелка вверх для инкремента или декремента задания.



5. Дважды нажмите CLR для замены задания на новое значение.



### 4.2.18 Изменение значения в виде 16-ричного или двоично-десятичного числа

Данная операция применяется для изменения значения слова, индицируемого в режиме 4.3.10. Данная операция возможна в режимах MONITOR и PROGRAM.

RUN	MONITOR	PROGRAM
НЕТ	ДА	ДА

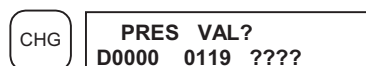
Слова SR 253 ... SR 255 нельзя изменить.

- 1, 2, 3,... 1. Вызовите просмотр состояния требуемого слова согласно 4.3.10. Если индицируется более 2 слов, требуемое слово должно быть слева.



(Индикация слова)

2. Нажмите клавишу CHG для начала операции.



3. Введите новое текущее значение и нажмите WRITE для внесения изменений в память.

При нажатии WRITE операция прекращается и происходит переход в режим нормального просмотра.



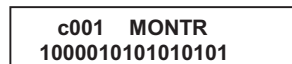
#### 4.2.19 Изменение параметра, представляемого в виде двоичного числа

Данная операция применяется для изменения значения слова, индицируемого в режиме 3-5-15. Данная операция возможна в режимах MONITOR и PROGRAM.

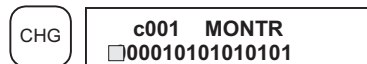
RUN	MONITOR	PROGRAM
НЕТ	ДА	ДА

Биты SR 25300 ... SR 25507 нельзя изменить.

1, 2, 3,... 1. Вызовите просмотр состояния требуемого слова (п. 4.3.12).



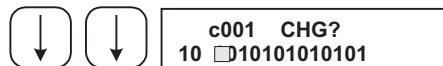
2. Нажмите клавишу CHG для начала операции модификации.



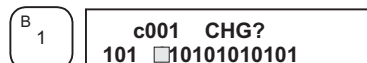
Над битом 15 появится мигающий курсор. Курсор показывает, какой бит можно изменять.

3. Для перемещения курсора и изменения битов служат 3 набора клавиш:

а. Для перемещения влево-вправо: клавиши Стрелка вниз - Стрелка вверх.



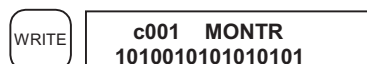
б. Для изменения состояния бита служат клавиши 1 и 0. После нажатия одной из этих клавиш курсор переместится вправо.



с. Для принудительного включения/выключения нажмите SHIFT + SET или SHIFT + RESET. После нажатия одной из этих клавиш курсор переместится вправо. Клавиша NOT уберет принудительное состояние.

*Замечание* Биты в области DM нельзя принудительно включить/выключить.

4. Для занесения изменений в память нажмите WRITE.



#### 4.2.20 Изменение десятичного числа со знаком

Данная операция применяется для изменения значения числа, индицируемого в виде десятичного числа со знаком в диапазоне -32768 ...32767. Оно автоматически преобразуется в 16-ричное (формат дополнения до 2).

Слова SR 253 - SR 255 нельзя изменить.

RUN	MONITOR	PROGRAM
НЕТ	ДА	ДА

1, 2, 3,... 1. Вызовите просмотр состояния требуемого слова в виде десятичного числа со знаком.

c000  
-00016

(Индикация десятичного со знаком)

2. Нажмите клавишу CHG для начала модификации.

CHG PRES VAL?  
C000 -00016

3. Введите новое значение и нажмите WRITE для внесения изменений в память.

При нажатии WRITE операция прекращается и происходит переход в режим просмотра десятичного числа со знаком.

Вводимое значение должно быть в диапазоне -32768 ...32767. Для ввода положительного числа используйте клавишу SET, для ввода отрицательного - RESET.

REC RESET D 3 C 2 7 6 8 WRITE c000  
-32768

Для возврата в нормальный режим индикации нажмите CLR или SHIFT и TR.

Если сделан ошибочный ввод, нажмите CLR для возврата значения, которое было до ввода. Затем введите правильное значение.

#### 4.2.21 Изменение десятичного числа без знака

Данная операция применяется для изменения значения числа, индицируемого в виде десятичного числа без знака в диапазоне 0 ... 65 535. Оно автоматически преобразуется в 16-ричное.

Слова SR 253 - SR 255 нельзя изменить.

RUN	MONITOR	PROGRAM
НЕТ	ДА	ДА

1, 2, 3,... 1. Вызовите просмотр состояния требуемого слова в виде десятичного числа без знака.

c000  
65520

(Индикация десятичного без знака)

2. Нажмите клавишу CHG для начала модификации.

CHG PRES VAL?  
C000 65520

3. Введите новое значение и нажмите WRITE для внесения изменений в память.

При нажатии WRITE операция прекращается и происходит переход в режим просмотра десятичного числа без знака.

Вводимое значение должно быть в диапазоне 0 ... 65 535.

D 3 C 2 7 6 8 WRITE c000  
32768

Для возврата в нормальный режим индикации нажмите CLR или SHIFT и TR.

Если сделан ошибочный ввод, нажмите CLR для возврата значения, которое было до ввода. Затем введите правильное значение.

#### 4.2.22 Принудительные установка/сброс бита

Данная операция применяется для принудительной установки или сброса бита, наибольшее применение находит при отладке программы или проверки подключения выходов.

Данная операция возможна только в режимах MONITOR и PROGRAM.

RUN	MONITOR	PROGRAM
НЕТ	ДА	ДА

- 1, 2, 3,... 1. Просмотрите состояния требуемого бита согласно 4.3.10. Если просматриваются два или более слов, требуемый бит должен быть самым левым.

```
00000 20000
^OFF ^ON
```

(Просмотр нескольких слов)

2. Нажмите SET для принудительного включения бита или RESET для принудительного выключения.

```
SET 00000 20000
    [ ] ON ^OFF
```

Курсор в левом нижнем углу показывает, что имеет место принудительное включение/выключение. Состояние бита остается только пока нажата клавиша. Бит вернется в исходное состояние спустя 1 цикл после отпускания клавиши.

3. Для сохранения состояния после отпускания клавиш нажмите SHIFT+SET, SHIFT+RESET. В этом случае принудительное состояние указывается следующим образом: принудительно включенный бит - S  
принудительно выключенный бит - R

Для возврата бита в исходное состояние нажать NOT или сделать сброс принудительного состояния. Подробности см. 4.3.23.

Принудительное состояние можно убрать сменой режима ПК (если SR 25211 не 0 - тогда принудительное состояние нельзя убрать переключением из PROGRAM в MONITOR) или когда работа прервалась из-за фатальной ошибки или прерывания питания.

#### 4.2.23 Убрать принудительное состояние включено/выключено

Данная операция применяется для восстановления состояния всех битов, которые были принудительно включены или выключены. Данная операция возможна в режимах MONITOR и PROGRAM.

RUN	MONITOR	PROGRAM
НЕТ	ДА	ДА

- 1, 2, 3,... 1. Нажмите клавишу CLR для вызова исходной индикации.  
2. Нажмите клавиши SET, RESET. Появится сообщение-подтверждение.

```
SET RESET 00000FORCE RELE?
```

*Замечание* Если вы ошиблись и нажали не ту клавишу, нажмите CLR и начните с начала.

3. Нажмите клавишу NOT чтобы убрать все принудительные установки.

```
NOT 00000FORCE RELE
    END
```

#### 4.2.24 Изменение индикации с 16-ричного на ASCII

Данная операция применяется для переключения режима индикации слова в виде 16-ричного числа (4 цифры) и ASCII.

Данная операция возможна в любом режиме.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

- 1, 2, 3,... 1. Вызовите просмотр состояния требуемого слова (п. 4.3.10). Если на просмотр вызваны 2 и более слов, требуемое слово должно находиться слева на дисплее.

D0000 D0001 4142 3031
--------------------------

(Индикация нескольких адресов)

2. Для вызова индикации в виде ASCII нажмите TR. Эти режимы индикации взаимно переключаются при нажатии TR.

TR	D0000 D0001 "AB" 3031
----	--------------------------

TR	D0000 D0001 4142 3031
----	--------------------------

#### 4.2.25 Индикация времени цикла

Данная операция применяется для индикации текущего среднего времени цикла (время сканирования). Данная операция возможна в режимах RUN и MONITOR при исполнении программы.

RUN	MONITOR	PROGRAM
ДА	ДА	ДА

- 1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации.  
2. Нажмите клавишу MONTR для индикации времени цикла

MONTR	00000SCAN TIME 012.1MS
-------	---------------------------

При нескольких повторных нажатиях MONTR числа могут быть различными, что вызвано изменением условий исполнения.

### 4.3 Пример программирования

В данной главе описана последовательность действий для написания программы с программатора.

#### 4.3.1 Подготовительные мероприятия

При написании программы для CPM1 в первый раз сделайте процедуры 1 ... 3 (стирание памяти).

- 1, 2, 3,... 1. Установите переключатель режимов на программаторе в PROGRAM и включите питание. На программаторе появится режим ввода пароля.



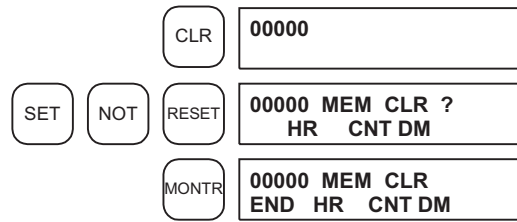
<PROGRAM> PASSWORD
-----------------------

2. Введите пароль, нажав клавиши CLR и затем MONTR.

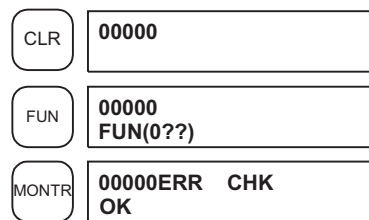
CLR	MONTR	<PROGRAM>
-----	-------	-----------

В данном месте можно нажать SHIFN и 1 для включения или выключения зуммера. (см. 4.3.3.)

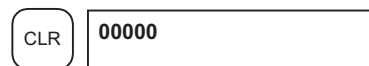
3. Очистите память CPM1 нажатием CLR, SET, NOT, RESET и MONTR. Нажмите несколько раз CLR, если индицируются сообщения об ошибках.



4. Выведите и очистите сообщения об ошибках, нажав CLR, FUN и затем MONTR. Продолжайте нажимать MONTR, пока все сообщения об ошибках не очистятся.



5. Нажмите CLR для вызова исходной индикации программирования (адрес программы 00000). Новую программу можно писать в данный адрес.



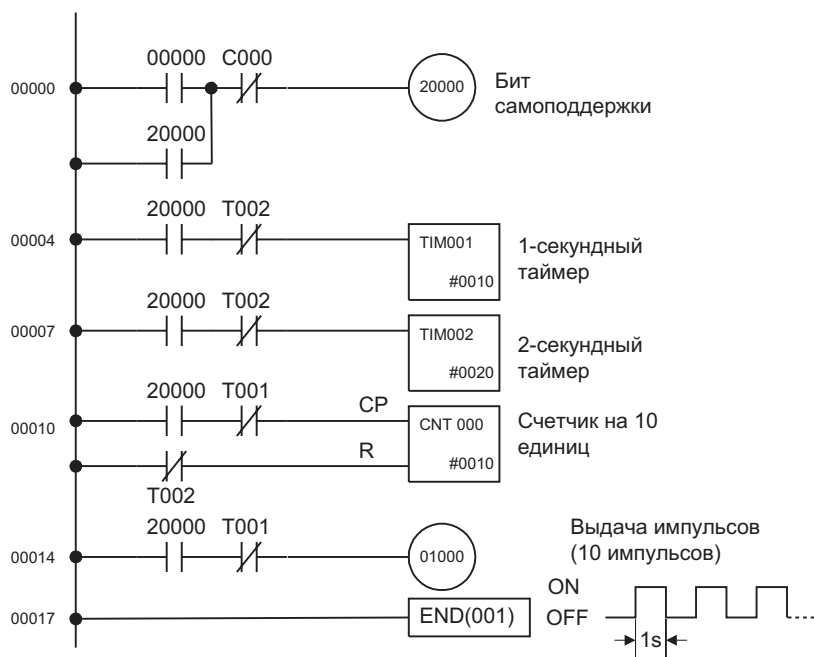
*Внимание!* Перед вводом пароля убедитесь, что переключатель находится в режиме PROGRAM. Тщательно проверьте систему перед первым исполнением программы.

#### 4.3.2 Пример программы

Следующая РКС служит для демонстрации того, как записать программу с программатора. Данная программа вызывает меандр на выходе IR 01000 (1 с в состоянии 1, 1 секунда в состоянии 0) 10 раз после включения IR 00000 в состояние 1.

Мнемоника данной программы представлена в таблице. Процедуры для ввода данной программы с программатора описаны в 4.4.3





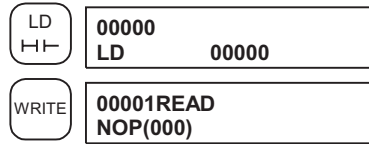
Адрес	Команда	Параметр	Описание процедуры ввода
00000	LR	00000	(1) Бит самоподдержки
00001	OR	20000	
00002	AND NOT	C 000	
00003	OUT	20000	
00004	LD	20000	(2) 1-секундный таймер
00005	AND NOT	T 002	
00006	TIM	001 # 0010	
00007	LD	20000	(3) 2-секундный таймер
00008	AND NOT	T 002	
00009	TIM	002 # 0020	
00010	LD	20000	(1) счетчик на 10 единиц
00011	AND	T 001	
00012	AND NOT	20000	
00013	CNT	000 # 0010	
00014	LD	20000	(5) Выдача импульсов (10 импульсов)
00015	AND NOT	T 001	
00016	OUT	01000	
00017	END(001)		(6) Команда END(001)

### 4.3.3 Процедуры программирования

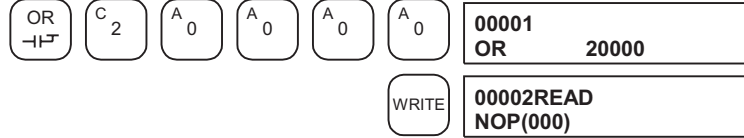
Программа записывается в соответствии с таблицей мнемоники (4.4.2). Процедура выполняется, начиная с исходной индикации. (Перед вводом новой программы очистите память).

#### (1) Ввод бита самоподдержки

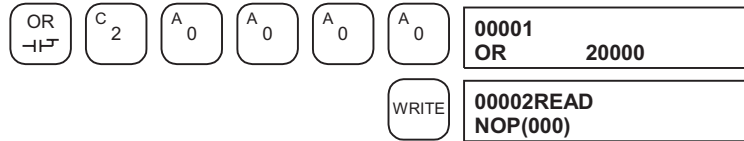
- 1, 2, 3,... 1. Введите нормально открытое условие IR 00000.  
(Необязательно вводить ведущие нули).



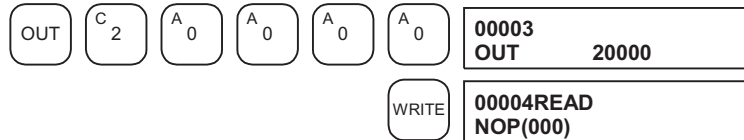
2. Введите OR условия IR 20000.



3. Введите AND с нормально закрытым условием C000.  
(Необязательно вводить ведущие нули).

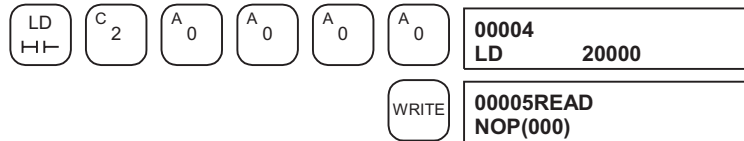


4. Введите OUT условия IR 20000.

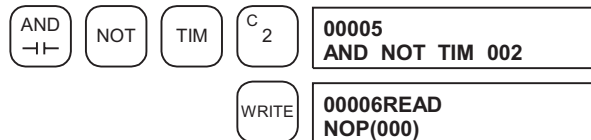


### (2) Ввод 1-секундного таймера

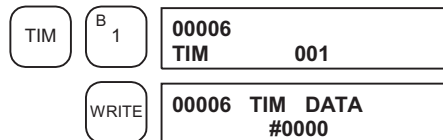
1, 2, 3,... 1. Введите нормально открытое условие IR 20000.



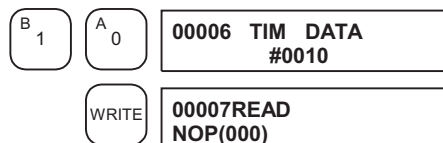
2. Введите AND с нормально закрытым условием T002.  
(Необязательно вводить ведущие нули).



3. Введите 1-секундный таймер T001.



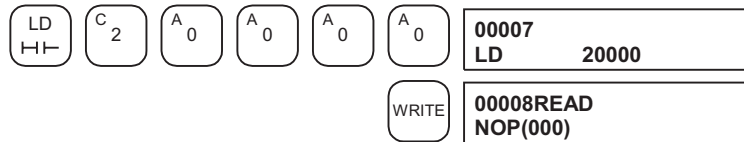
4. Введите Задание для T001 (#0010 = 1.0 с).



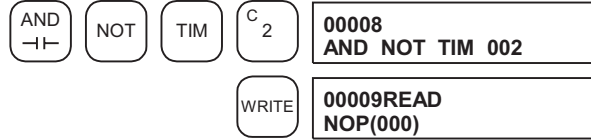
### (3) Ввод 2-секундного таймера

Для ввода 2-секундного таймера проделайте следующие операции:

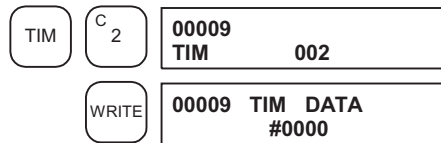
1, 2, 3,... 1. Введите нормально открытое условие IR 20000.



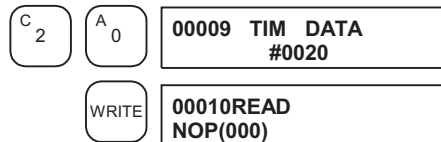
2. Введите AND с нормально закрытым условием T002.  
(Необязательно вводить ведущие нули).



3. Введите 2-секундный таймер T002.



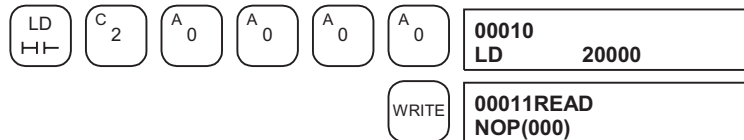
4. Введите Задание для T002 (#0020 = 2.0 с).



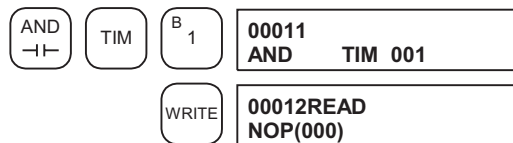
#### (4) Ввод счетчика на 10

Для ввода счетчика на 10 проделайте следующие операции:

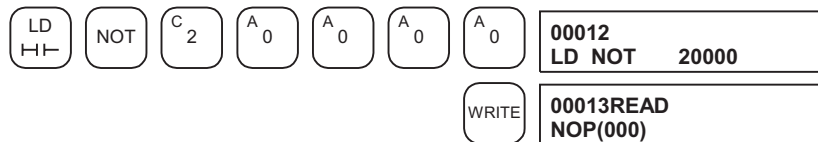
1, 2, 3,... 1. Введите нормально открытое условие IR 20000.



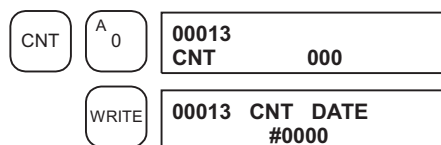
2. Введите AND с нормально закрытым условием T001.  
(Необязательно вводить ведущие нули).



3. Введите нормально закрытое условие IR 20000.



4. Введите счетчик 000.



5. Введите Задание для счетчика 000 (#0010 = 10 единиц счета).

B 1	A 0	00013 CNT DATE #0010
WRITE		00014READ NOP(000)

**(5) Ввод выдачи импульсов**

1, 2, 3,... 1. Введите нормально открытое условие IR 20000.

LD	C 2	A 0	A 0	A 0	A 0	000014 LD 20000
WRITE						00015READ NOP(000)

2. Введите AND с нормально закрытым условием T001.

(Необязательно вводить ведущие нули).

AND	NOT	TIM	B 1	00015 AND NOT TIM 001
WRITE				00016READ NOP(000)

3. Введите OUT команды IR 01000.

(Необязательно вводить ведущие нули).

OUT	B 1	A 0	A 0	A 0	00016 OUT 01000
WRITE					00017READ NOP(000)

**(6) Ввод команды END(001)**

Введите команду END(001)

FUN	00017 FUN(0??)	
A 0	B 1	00017 FUN(001)
WRITE		00018READ NOP(000)

**4.3.4 Контроль программы**

Проверяйте синтаксис программы в режиме PROGRAM для того, чтобы убедиться, что программа введена правильно.

1, 2, 3,... 1. Нажмите CLR для вызова исходной индикации.

00000
-------

2. Нажмите клавишу CLR. Появится подсказка, требующая ввести желаемый уровень контроля.

SRCH	00000PROG CHK CHKLEVEL (0-2)?
------	----------------------------------

3. Введите желаемый уровень контроля (0, 1 или 2). Контроль программы начнется после ввода уровня и первая обнаруженная ошибка будет индигатироваться.

A 0	00178CIRCUIT ERR OUT 00200
-----	-------------------------------

*Замечание* Подробности об уровнях контроля см. 5.5.

4. Нажмите клавишу SRCH для продолжения поиска. Появится сообщение о следующей ошибке. Продолжайте нажимать SRCH для продолжения поиска ошибок.

Поиск будет продолжаться до команды END или достижения границы области памяти.

Если индицируется сообщение об ошибке, отредактируйте программу для исправления ошибок и снова проверьте программу. Продолжайте проверку, пока не будут исправлены все ошибки.

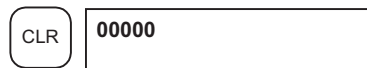
#### 4.3.5 Тестовый прогон в режиме MONITOR

Переключите CPM1 в режим MONITOR и проверьте работу программы.

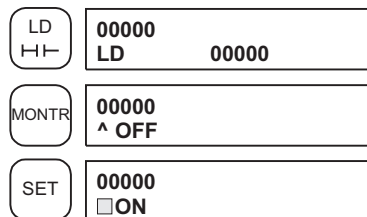
1, 2, 3,... 1. Переключите переключатель режимов программатора в режим MONITOR.



2. Нажмите клавишу CLR для вызова исходной индикации.



3. Принудительно установите бит входа (IR 00000) с программатора для пуска программы.



Курсор с левом нижнем углу дисплея указывает, что имеет место принудительная установка. Бит останется в положении 1, пока удерживается клавиша SET.

4. Если программа выполняется правильно, индикатор выхода 01000 мигнет 10 раз. После 10 миганий индикатор погаснет.

Если индикатор не мигает, в программе ошибка. В этом случае проверьте программу и установите биты в 1/0 для проверки команд.



## **Глава 5. Тестовые прогоны и обработка ошибок**

*В данной главе описаны процедуры тестовых прогонов операций СРМ1, функций самодиагностики и обработка ошибок для идентификации и исправления аппаратных и программных ошибок, которые могут произойти при работе ПК.*

## 5.1 Начальная проверка системы и процедура тестового прогона

### 5.1.1 Начальная проверка системы

После настройки и подключения СРМ1 проверьте следующее. Перед тестовым прогоном обязательно проверьте проводные подключения и соединения.

Параметр	Объект проверки
Питание и подключение входов/выходов	Подключение правильно? Клеммы надежно зажаты? Между наконечниками или проводами нет замыкания? Подробности см. 3.4.
Соединительные кабеля	Все кабеля правильно подключены и закреплены? Подробности см. 3.4.

### 5.1.2 Процедура тестового прогона СРМ1

#### 1, 2, 3,... 1. Подключение питания

Проверьте напряжение питания и подключения клеммника СРМ1.

Проверьте напряжение питания и подключения клеммника устройств входа/выхода.

Включите питание и проверьте, чтобы горел индикатор POWER.

Используйте программатор для установления СРМ1 в режим PROGRAM.

#### 2. Проверка подключения входов/выходов

В режиме СРМ1 PROGRAM проверьте подключение выходов, принудительно включая и выключая выходные биты. Подробности см. 4.3.22.

Проверьте подключение входов с помощью входных индикаторов или просмотра с программатора.

#### 3. Тестовый прогон

Используйте программатор для установки СРМ1 в режим RUN или MONITOR и проверьте, горит ли индикатор RUN.

Проверьте последовательность операций путем принудительной установки/сброса битов и т. д.

#### 4. Отладка

Исправьте обнаруженные ошибки.

#### 5. Сохранение программы

Используйте программатор для записи программы на дискету

Выведите на принтер бумажный экземпляр.

*Замечание* Подробности об использовании программатора и SSS см. гл. 4

### 5.1.3 Предосторожности при обращении с памятью FLASH

Для защиты памяти FLASH соблюдайте следующие меры предосторожности.

#### 1, 2, 3,... 1. Изменения, произведенные в программе, области DM только для чтения (DM 6144 ... DM 6599) или установочных параметрах (DM 6600 ... DM 6655) записываются в память FLASH при переключении режимов работы СРМ1.

Данные изменения будут потеряны, если они не записаны в память FLASH и питание отключилось более, чем на 20 дней (при 25<sup>0</sup>C), поскольку конденсатор поддержки ОЗУ разряжается.

Данные изменения можно сохранить путем переключения СРМ1 в режим RUN или MONITOR или включения СРМ1 вскоре после сделанных изменений.



2. Первая операция CPM1 после изменения в программе, области DM только для чтения (DM 6144 ... DM 6599) или установочных параметрах (DM 6600 ... DM 6655) займет на 600 мс больше, чем обычно. Обязательно принимайте в расчет эту задержку при пуске.

3. Если одна из трех следующих операций выполняется в режиме MONITOR или RUN, CPM1 увеличит время цикла до 600 мс и прерывания будут запрещены, пока программа или установочные параметры переписываются.

Программа изменяется он-лайнными операциями.

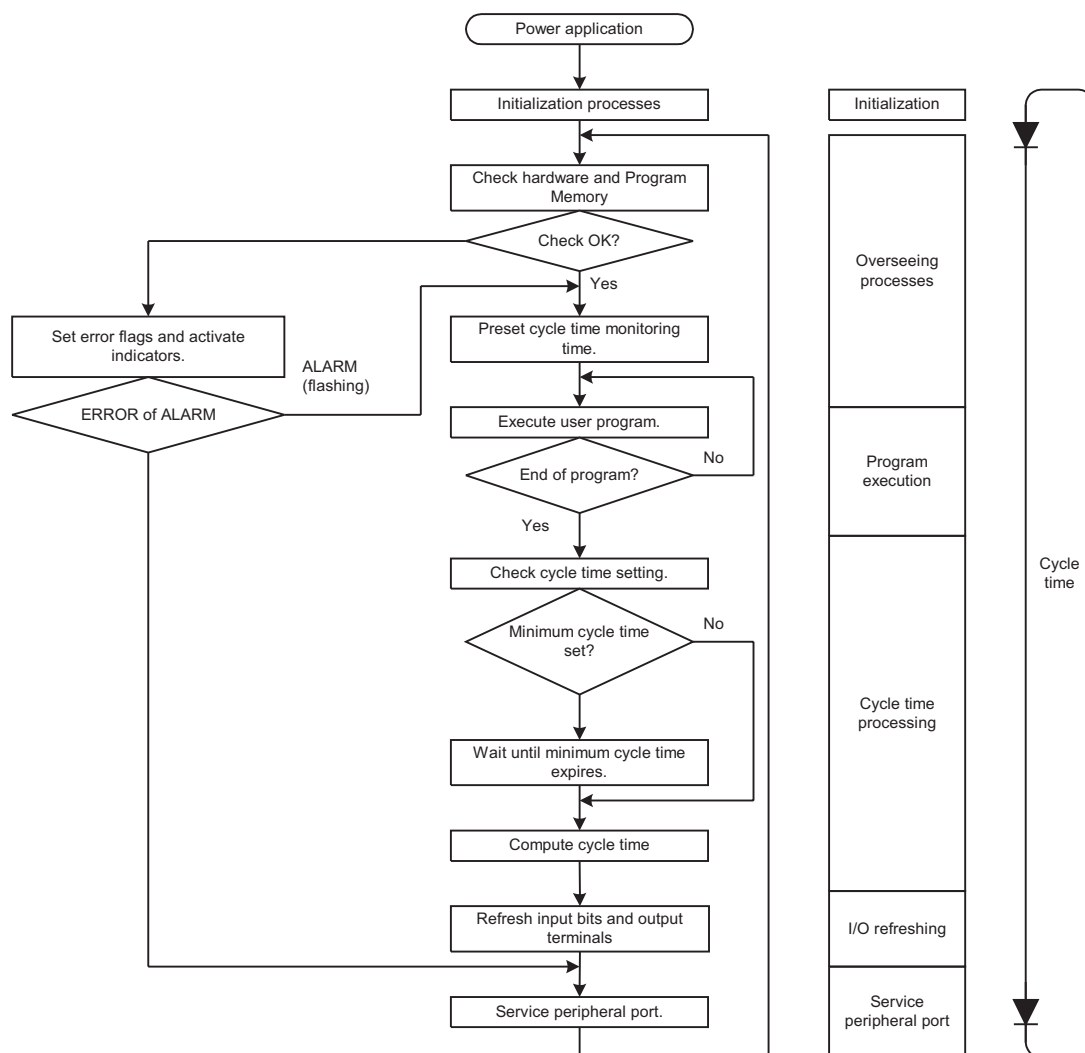
Изменения в области DM только для чтения (DM 6144 ... DM 6599)

Изменения установочных параметров (DM 6600 ... DM 6655)

Сообщение об ошибке SCAN TIME OVER (превышено время цикла) при данных операциях не появляется. При он-лайнных операциях они могут оказать влияние на время реакции на вход.

## 5.2 Цикл CPM1

Общий алгоритм работы CPM1 показан на схеме. Инициализации CPM1 вызывается при включении питания. Если ошибок не обнаружено, последовательно (циклически) выполняются операции диспетчеризации, исполнения программы, обновления входов/выходов и обслуживания периферийных устройств. Среднее время цикла можно наблюдать с программатора.



*Замечание* Процессы инициализации включают очистку областей IR, SR, и AR, установку системных таймеров и проверка блоков входов/выходов.

## 5.3 Функции самодиагностики

В СРМ1 есть различные функции самодиагностики для идентификации и исправления возможных ошибок и сокращения времени простоя.

Ошибки ПК делятся на 2 категории, в зависимости от серьезности ошибок. Фатальные ошибки - наиболее серьезные, которые останавливают работу СРМ1. Нефатальные ошибки менее серьезны и не останавливают работу СРМ1.

### 5.3.1 Нефатальные ошибки

Работа ПК и отработка программы продолжается после появления одной или нескольких таких ошибок. Хотя работа ПК продолжается, причину ошибки нужно выявить и устранить как можно быстрее.

При появлении такой ошибка индикаторы POWER (СЕТЬ) и RUN (РАБОТА) остаются включенными, индикатор ERR/ALM (Неисправность) мигает.

Сообщение	FAL N	Значение сообщения и порядок действий
SYS FAIL FAL ** (* может быть 01 ... 99 или 9B)	01 ... 99	В программе выполнена команда FAL(06). Проверьте номер FAL, определите причину, вызвавшую ее срабатывание, устраните причину и сотрите сообщение.
	9B	Обнаружена ошибка в установочных параметрах. Проверить флаги AR 1300 ... AR 1302 и исправить следующим образом:
		AR 1300 = 1: обнаружен некорректный установочный параметр (DM 6600 - DM 6614) при включении питания. Исправьте установочные параметры в режиме PROGRAM и снова включите питание
		AR 1301 = 1: обнаружен некорректный установочный параметр (DM 6615 - DM 6644) при переходе в режим RUN. Исправьте установочные параметры в режиме PROGRAM и снова включите режим RUN.
		AR 1302 = 1: обнаружен некорректный установочный параметр (DM 6645 - DM 6655) во время работы. Исправьте установочные параметры и сотрите ошибку.
SCAN TIME OVER	F8	Время цикла превысило 100 мс. (SR 25309 = 1) Указывает, что время программного цикла больше рекомендуемого. При возможности сократите цикл. (СРМ1 можно настроить так, чтобы эта ошибка не выявлялась).
Ошибки связи (нет сообщения)	Нет	Если произошел сбой связи по периферийному порту, соответствующий индикатор СОММ прекратит мигать. Проверьте соединительные кабели и перезапуститесь. Проверьте, установлены ли флаги ошибок в AR 0812 =1.

### 5.3.2 Фатальные ошибки

Работа ПК и отработка программы прекратится и все выходы ПК выключаются при любой такой ошибке. Операции СРМ1 будут возобновлены только после выключения ПК и включения или использования программатора для переключения в режим PROGRAM и сброса признаков ошибок.

При прерывании питания все индикаторы на ЦПУ выключены. При других фатальных ошибках горят индикаторы POWER (ПИТАНИЕ) и ERR/ALM (АВАРИЯ). Индикатор RUN (РАБОТА) будет выключен.

Сообщение	FALS N	Значение сообщения и порядок действий
Прерывание питания (нет сообщения)	нет	Было прерывание питание более чем 10 мс. Проверить напряжение и подвод питания. Снова включить питание.

Сообщение	FALS N	Значение сообщения и порядок действий
MEMORY ERR (Ошибка памяти)	F1	AR 1308 = 1: В программе пользователя существует незаданная битовая область. Проверьте программу и исправьте ошибку.
		AR 1309 = 1: Произошел сбой в памяти FLASH. Поскольку число записей в память FLASH превысило заданный уровень, замените ЦПУ.
		AR 1301 = 1: Ошибка контрольной суммы в установочных параметрах (DM 6144 ... DM 6599). Проверьте и откорректируйте установочные параметры в области DM.
		AR 1311 = 1: Ошибка контрольной суммы в установочных параметрах. Проинициализируйте все установочные параметры и введите снова.
		AR 1312 = 1: Ошибка контрольной суммы в программе. Проверьте программу и исправьте обнаруженные ошибки.
NO END INST	F0	В программе отсутствует команда END(01). Запишите в конце программы END(01).
I/O BUS ERROR	C0	Произошел сбой при передаче данных между ЦПУ и блоком входов/выходов. Проверьте соединительный кабель к блоку входов/выходов.
I/O UNIT OVER	E1	Подключено слишком много блоков входов/выходов. Исправьте конфигурацию входов/выходов.
SYS FAIL FALS ** (** может быть 01...99, или 9F)	01 - 99	В программе выполнялась команда FALS (07). Проверьте номер FALS для определения условий, вызвавших сообщение, устраните причину и сотрите сообщение.
	9F	Время цикла превысило "Контрольное время цикла" (DM 6618), вызывающее FALS 9F. Проверьте время цикла и настройте параметр "Контрольное время цикла".

### 5.3.3 Опознавание ошибок

Ошибки ПК опознаются по сообщениям на программаторе, флагам ошибок в областях SR и AR, и кодам ошибок в SR 25300 ... SR 25307.

#### Сообщения об ошибках

Сообщения об ошибках, вызванные функциями самодиагностики, можно прочитать на программаторе или на управляющем компьютере с SSS.

#### Флаги ошибок

Когда функция диагностики обнаруживает ошибку аппаратуры, она включает соответствующие флаги в областях SR и AR.

#### Код ошибки

Когда функция диагностики обнаруживает ошибку, соответствующий код записывается в SR 25300 ... SR 25307. (Код ошибки - это 2-разрядное 16-ричное число).

### 5.3.4 Ошибки, определяемые пользователем

Имеются 3 команды, с помощью которых пользователь может задать свои собственные ошибки или сообщения.

FAL(06) вызывает признак нефатальной ошибки

FAL(07) вызывает признак фатальной ошибки

MSG(46) посылает сообщение на программатор или управляющий компьютер, подключенный к ПК.

**FAILURE ALARM - FAL (06) (Нефатальная ошибка)**

FAL(06) - вызов признака нефатальной ошибки. При исполнении FAL (06) производятся следующие действия:

- 1, 2, 3,... 1. Мигает индикатор ERR/ALM на ЦПУ. Работа ПК продолжается.
2. Номер данной FAL (две десятичных цифры от 01 до 99) заносится в SR 25300 ... SR 25307.

Номера FAL могут расставляться произвольно для индикации определенных условий. Одинаковые номера нельзя использовать для FAL и FALS. Для стирания сообщения FAL устраните причину ошибки, выполните FAL 00 или сотрите сообщение с помощью программатора.

**Severe Failure Alarm - FALS (07) (Признак фатальной ошибки)**

FALS(07) - команда, вызывающая признак фатальной ошибки. При исполнении команды FALS (07) производятся следующие действия:

- 1, 2, 3,... 1. Исполнение программы прерывается и выходы сбрасываются в 0.
2. Индикатор ERR/ALM на ЦПУ горит.
3. Номер данной FALS из двух двоично-десятичных чисел (01-99) заносится в SR 25300 - 25307.
4. Номер FALS заносится в область протокола ошибок. В CQM1 также записывается время появления признака ошибки, если используется кассета памяти с часами (RTC).

Номера FALS могут расставляться произвольно для индикации определенных условий. Одинаковые номера нельзя использовать одновременно для FAL и FALS. Для удаления сообщения FALS переключите ПК в режим PROGRAM, устраните причину сообщения, сотрите признак с помощью программатора.

**Сообщение - MSG (46) (Message)**

Команда MSG(46) применяется для индикации сообщений на программаторе. Сообщения из максимум 16 знаков индицируется, когда условия выполнения данной команды = 1.

**5.4 Ошибки при работе с программатором**

Следующие сообщения об ошибках могут появиться при работе с программатора. Исправьте ошибки, следуя указаниям таблицы, и продолжите работу.

Для работы с неисправностями, которые могут появиться при работе с LSS/SSS или устройством доступа к данным, см. инструкции: Инструкция по работе с LSS, Инструкция по работе с SSS, Инструкция по работе с устройством доступа к параметрам.

Сообщение	Значение сообщения и действия при сообщении
REPL ROM	Была попытка записи в память, защищенную от записи. Установите переключатель защиты от записи в положение OFF (секция 1 переключателя DIP на ЦПУ). В CPM1 установите биты 00 ... 03 DM 6602 в 0.
PROG OVER	Команда в последнем адресе памяти не NOP (00). Сотрите ненужные команды в хвосте программы.
ADDR OVER	Задан адрес, превышающий наибольший в памяти программ. Введите меньший адрес.
SETDATA ERR	Был ввод FALS 00, а 00 не может быть введен. Снова введите данные.
I/O NO. ERR	Был задан адрес, превышающий наибольший адрес зоны данных, т.е. адрес слишком велик. Изучите требования команды и снова введите адрес.

## 5.5 Ошибки программирования

Данные ошибки в синтаксисе программ обнаруживаются при контроле программы операцией “Контроль Программы”.

Имеется 3 уровня проверки программ. Перед проверкой должен быть задан необходимый уровень. В таблице приведены тип ошибок, сообщения и объяснения всех ошибок синтаксиса.

- Уровень контроля 0 проверяет на ошибки типа А,В,С.
- Уровень контроля 1 проверяет на ошибки типа А,В.
- Уровень контроля 2 проверяет на ошибки только типа А.

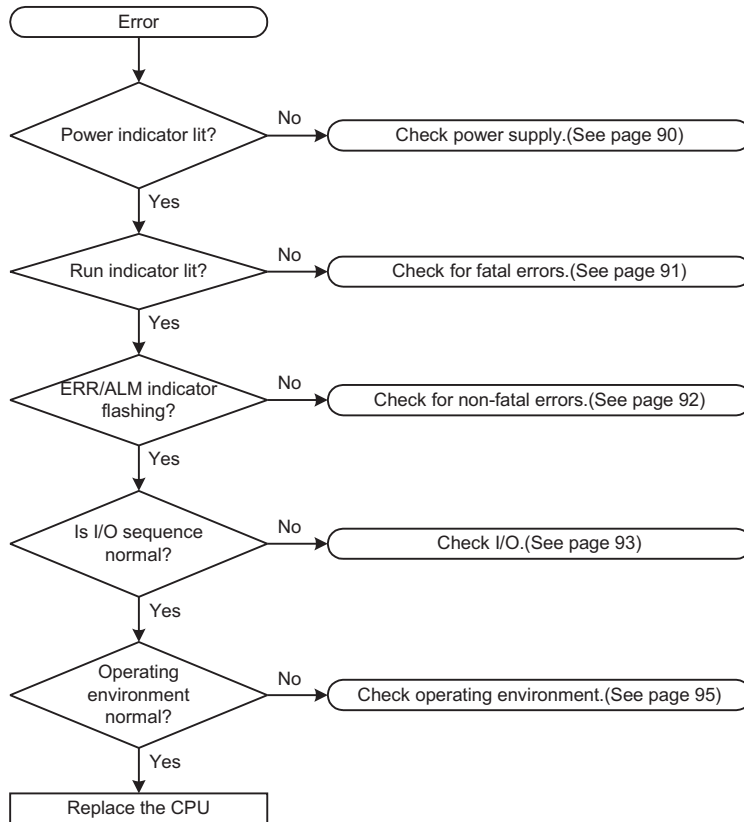
Тип	Сообщение	Значение сообщения и порядок действий
	?????	Программа запорчена, появился несуществующий код. Введите программу снова.
	CIRCUIT ERR	Число логических блоков и команд логических блоков не совпадает, т.е. LD или LD NOT использованы для начала логического блока, результаты которого не используются никакими другими командами, либо применена команда, для которой нет требуемого числа логических блоков. Проверьте программу.
	OPERAND ERR	Операнд команды, лежит вне допустимой зоны. Измените значение операнда, чтобы он лежал в допустимой зоне
	NO END INSTR	В программе отсутствует команда END (001). Запишите END (001) в конце программы
	LOCKN ERR	Команда в неправильном месте программы. Изучите, как пользоваться командой и скорректируйте программу.
	JME UNDEFD	Команда JME (004) отсутствует для команды JMP (005). Скорректируйте номер перехода или вставьте положенную команду JME (004).
	DUPL	Дважды использован один и тот же номер перехода или подпрограммы. Скорректируйте программу, чтобы один номер использовался только для одной из них.
	SBN UNDEFD	Команда SBS (091) запрограммирована для несуществующей подпрограммы. Скорректируйте номер или запрограммируйте подпрограмму.
	STEP ERR	Некорректно использованы STEP (008) с номером секции и STEP (008) без номера секции. Изучите работу команды. Скорректируйте программу.
В	IL-ILC ERR	IL (002) и ILC (003) не используются попарно. Скорректируйте программу, чтобы у каждой IL(002) был свой ILC (003). Хотя это сообщение об ошибке появится, если более одной IL (002) используется с одной IL (003), программа выполнится как написано. Перед отработкой убедитесь, что Ваша программа написана в соответствии с замыслом.
	JMP-JME ERR	JMP (004) и JME (005) не используются в паре. Перед отработкой убедитесь, что Ваша программа написана в соответствии замыслом.
	SBN-RET ERR	Если индицируемый адрес - адрес SBN (092), 2 разные подпрограммы определены одним именем. Измените один из номеров или удалите одну из подпрограмм. Если индицируемый адрес - адрес RET (093), данная команда использована неправильно. Изучите работу команды RET и скорректируйте программу.

Тип	Сообщение	Значение сообщения и порядок действий
С	COIL DUPL	Один и тот же бит управляется (т.е. включается и/или выключается) более чем одной командой. (напр. OUT, OUT NOT, DIFU (13), DIFD (14), KEEP (11), SFT (10)). Хотя это разрешено для некоторых команд, просмотрите работу команды и или убедитесь, что программа написана правильно, или перепишите программу, чтобы каждый бит управлялся одной командой.
	JMP UNDEFD	JME(005) был использован с JMP (004) не с таким номером. Добавьте JMP (004) с таким же номером или удалите JME (005).
	SBS UNDEFD	Существует подпрограмма, которая не вызывается SBS(091). Запрограммируйте вызов подпрограммы в нужном месте или удалите подпрограмму, которая не используется

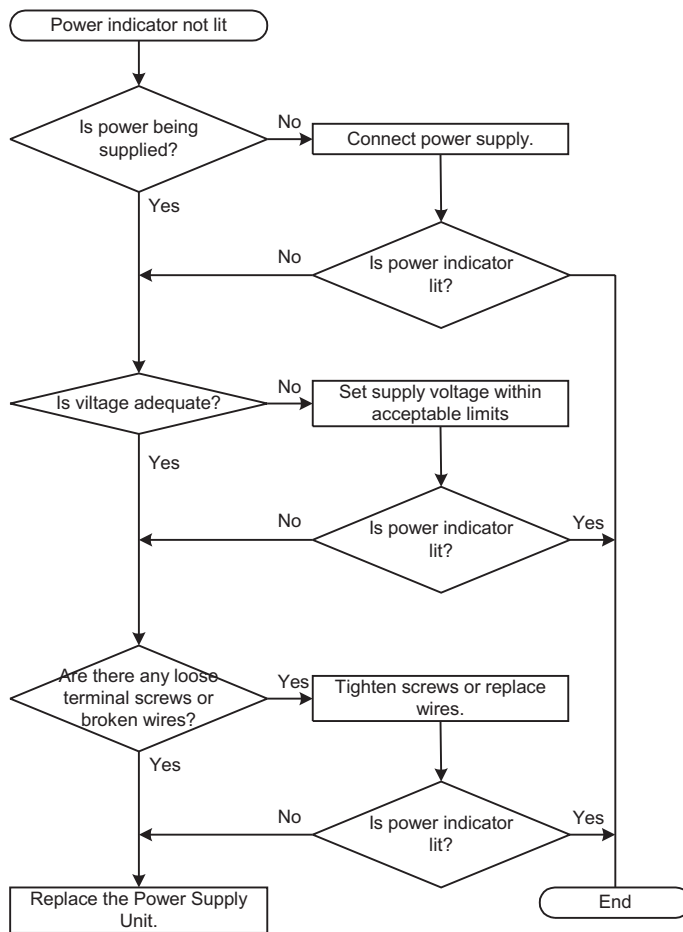
## 5.6 Алгоритмы поиска неисправностей

Для поиска неисправностей, произошедших при работе СРМ1, пользуйтесь следующими алгоритмами.

### Главный алгоритм поиска

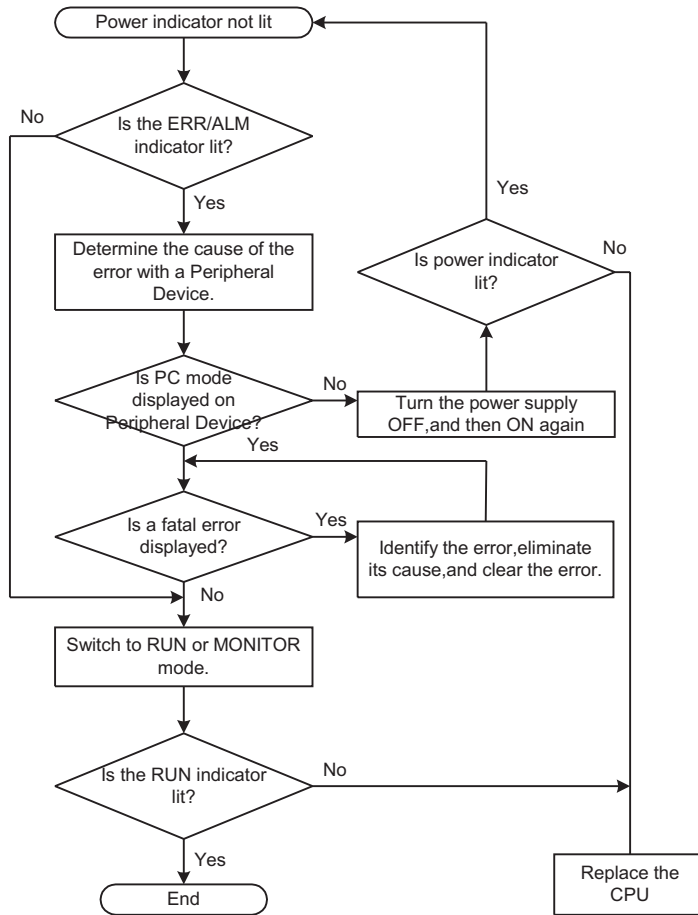


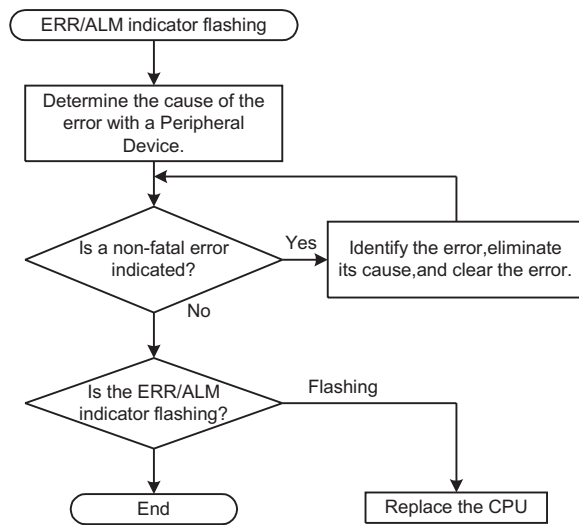
## Проверка питания



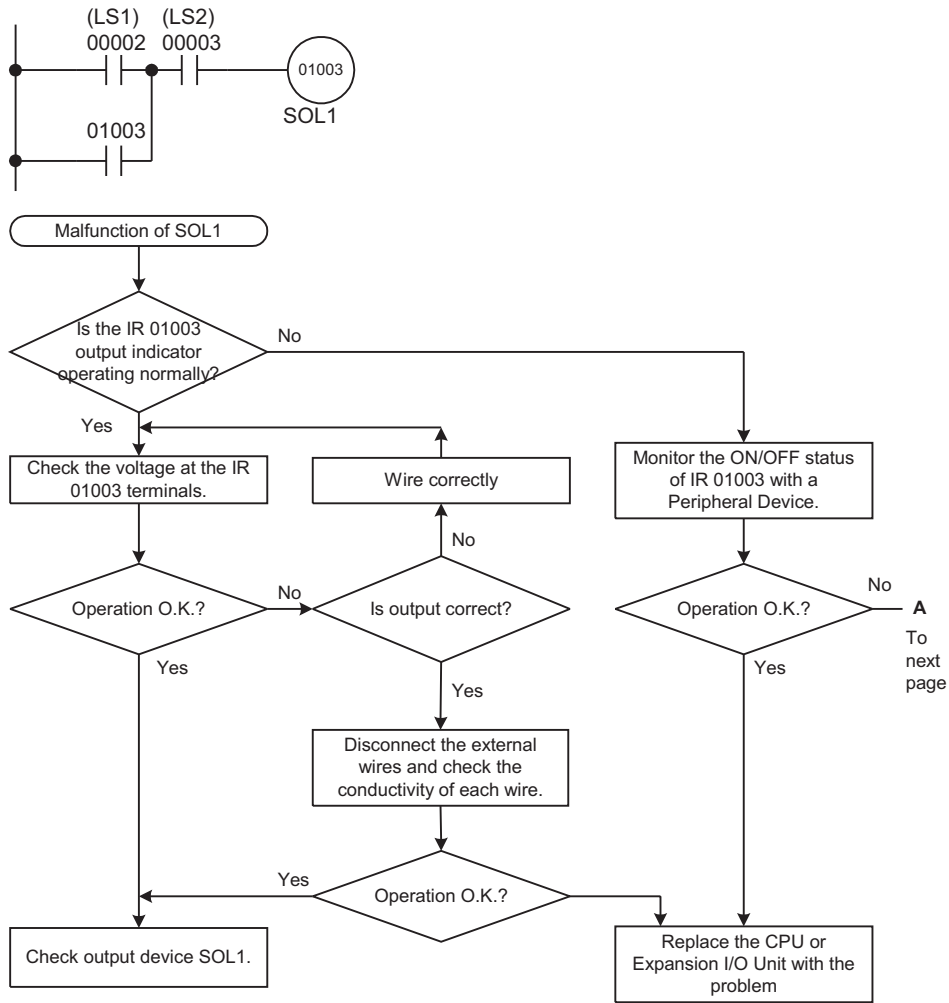


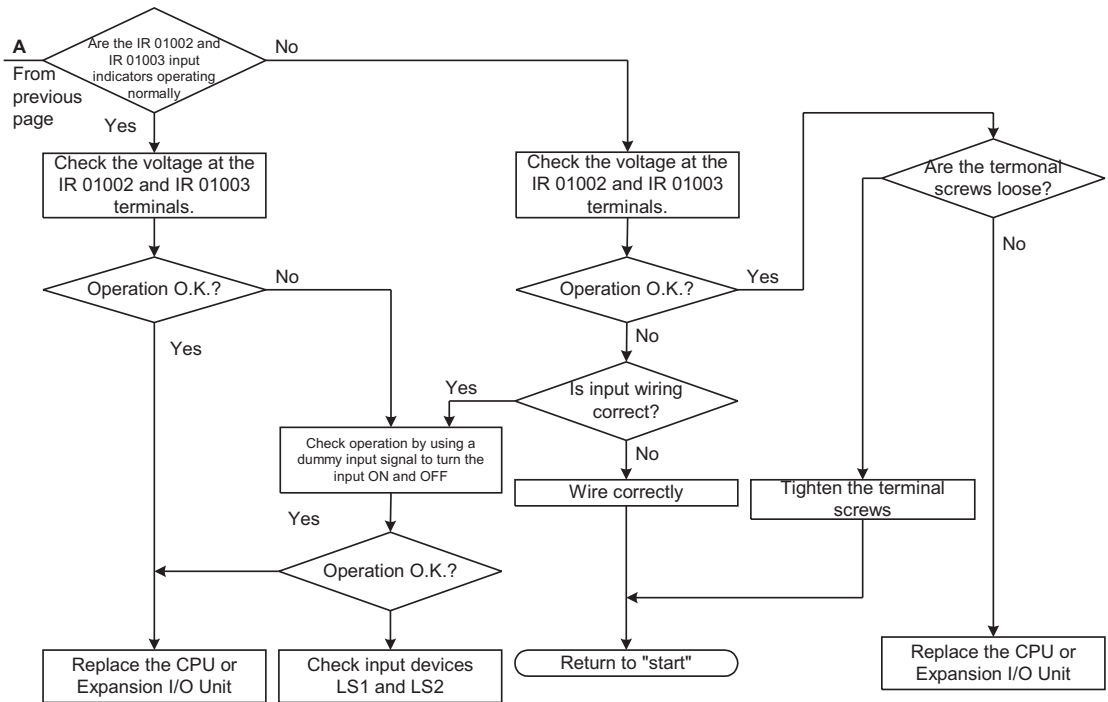
## Поиск фатальных ошибок



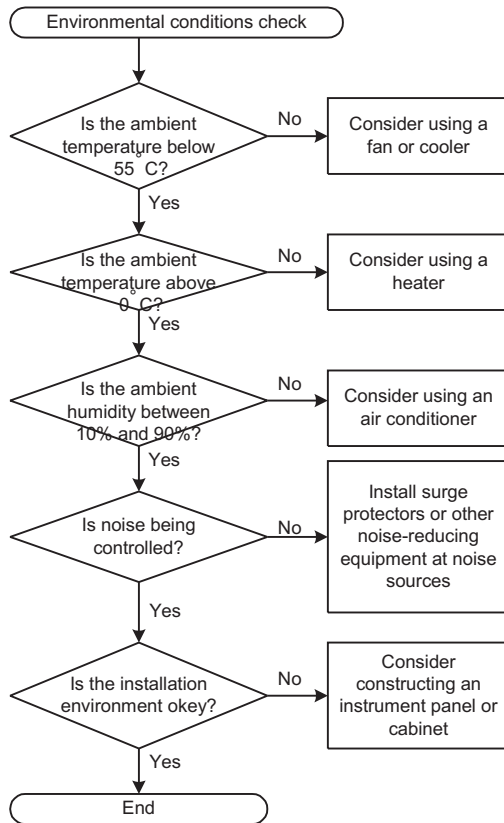
**Поиск нефатальных ошибок**

Проверка входов/выходов





## Проверка окружающей среды





# Приложение А

## Стандартные модели

### ЦПУ

Описание	Входы	Выходы	Питание	Номер модели
ЦПУ с 10 входами/выходами	6 точек	4 точки	100 ... 240 В перем. тока, 50/60 Гц	CPM1-10CDR-A
			24 В пост. тока	CPM1-10CDR-D
ЦПУ с 20 входами/выходами	12 точек	8 точек	100 ... 240 В перем. тока, 50/60 Гц	CPM1-20CDR-A
			24 В пост. тока	CPM1-20CDR-D
ЦПУ с 30 входами/выходами	18 точек	12 точек	100 ... 240 В перем. тока, 50/60 Гц	CPM1-30CDR-A
			24 В пост. тока	CPM1-30CDR-D

### Блок расширения входов/выходов

Описание	Входы	Выходы	Номер модели
Блок расширения с 20 входами/выходами	6 точек	4 точки	CPM1-20EDR

### Адаптеры связи

Описание	Выходы	Номер модели
Адаптер RS-232C	Осуществляет связь между периферийным портом и устройствами RS-232C	CPM1-CIF01
Адаптер RS-422	Осуществляет связь между периферийным портом и устройствами RS-422	CPM1-CIF11

### Периферийные устройства

Наименование	Номер модели	Характеристика
Программатор	CQM1-PRO01-E	В комплекте кабель 2м
	C200H-PRO27-E	Портативный, с задней подсветкой. Нужен кабель C200H-CN222 или C200H-CN422 (см. ниже)
Пакет поддержки SYSMAC	C500-ZL3AT1-E	2 дискеты 3.5 ", IBM PC/AT совместимые
Соединительные кабели	CQM1-CIF02	Соединяет IBM PC/AT или совместимые компьютеры с периферийным портом.
	C200H-CN222	Соединяет программатор C200H с периферийным портом (2м).
	C200H-CN422	Соединяет программатор C200H с периферийным портом (4м).



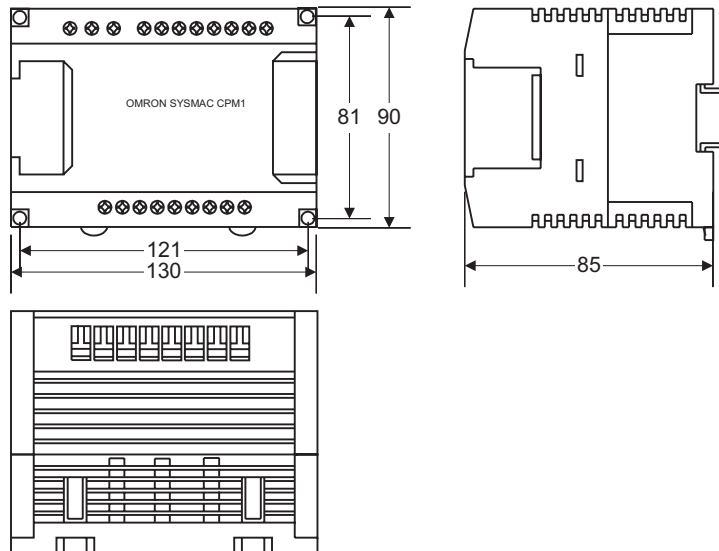


# Приложение В

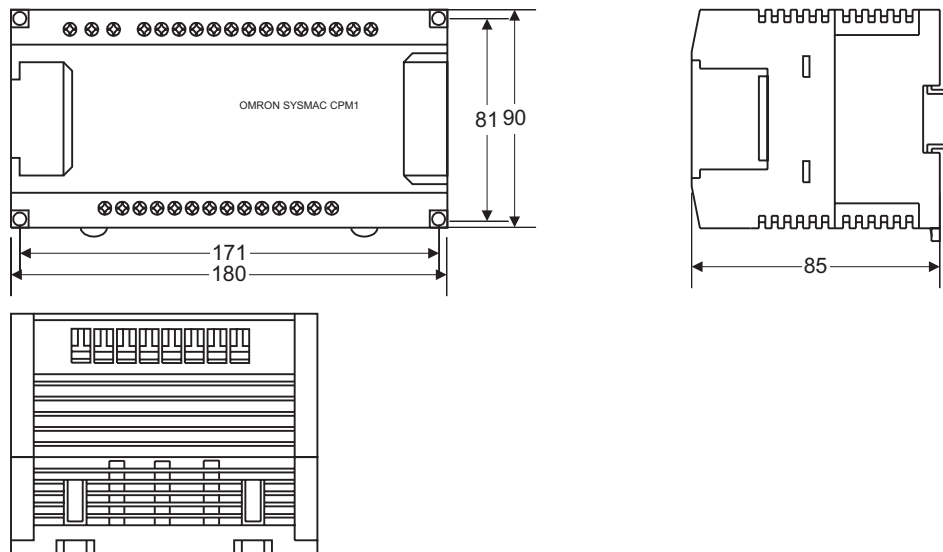
## Габариты

Все размеры приведены в мм.

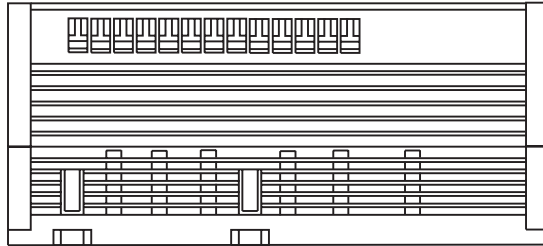
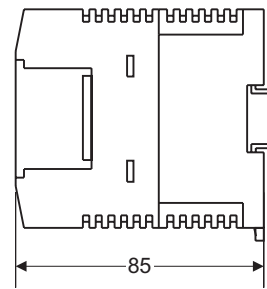
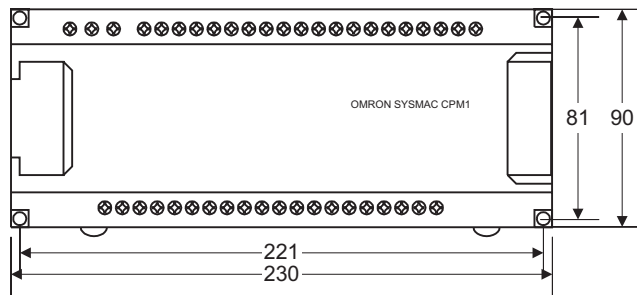
### CPM1-10CDR-



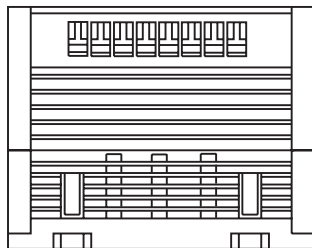
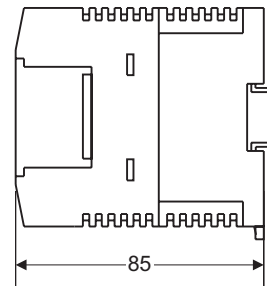
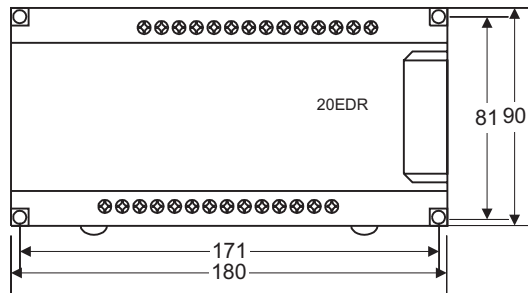
### CPM1-20CDR-



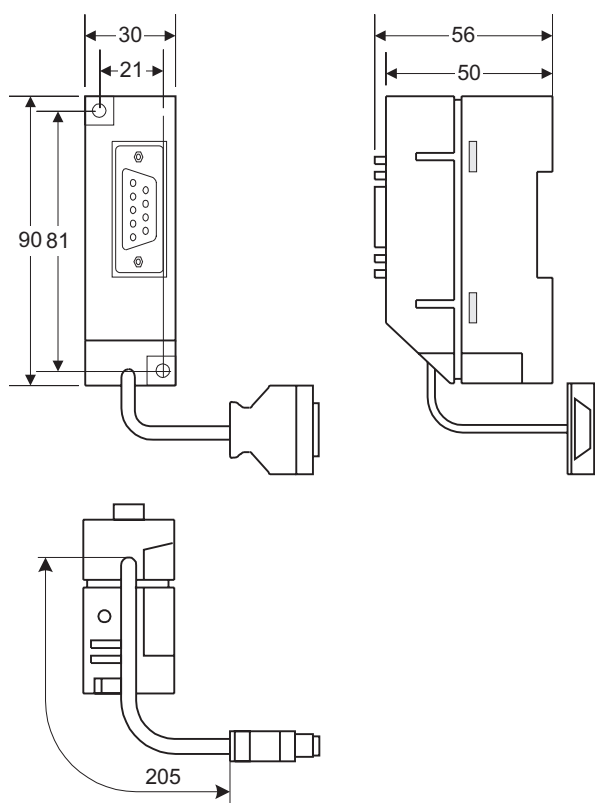
**CPM1-30CDR-**



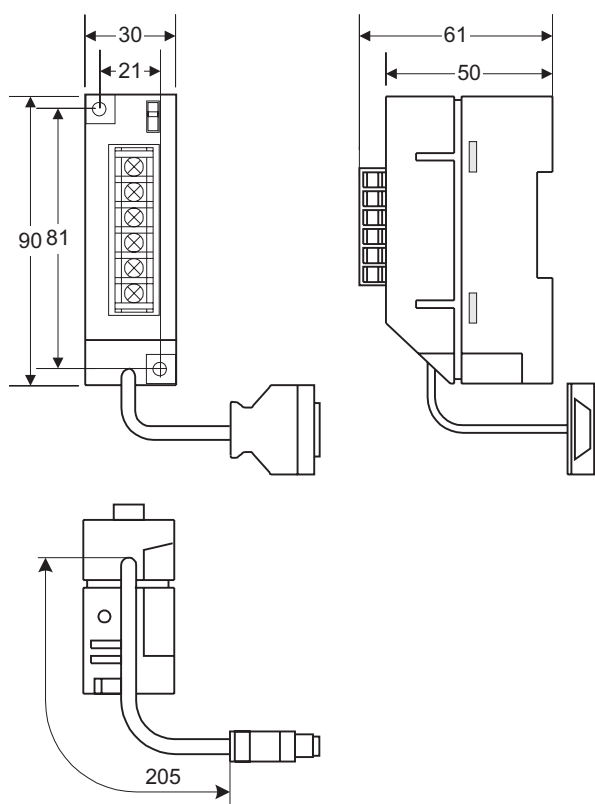
**CPM1-20EDR**



**CPM1-CIF01**



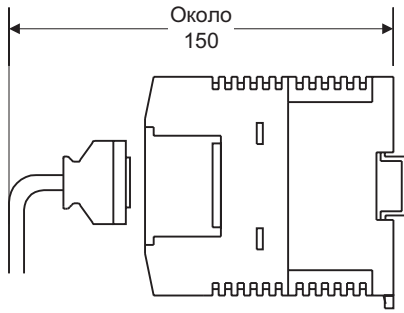
**CPM1-CIF11**



**Габариты с подключенными периферийными устройствами**

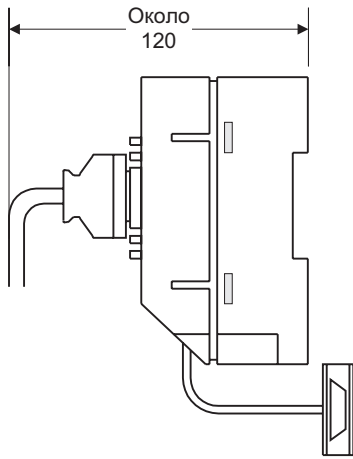
**CPM1-\_\_CDR-\_\_**

Когда подключен адаптер связи или программатор.



**СРМ1-С1F01**

Когда подключен разъем RS-232C



## Глоссарий

### **\*DM**

См. Косвенный адрес (в области DM).

### **1:1 связь**

Связь между двумя ПК, организованная таким образом, чтобы образовать общую область LR.

### **16-ричное представление числа**

Система представления чисел, на базе 16 цифр. В ПК все данные хранятся в двоичной форме, однако ввод и индикация на программирующих устройствах часто производятся в 16-ричном виде для упрощения операций. Каждая группа из четырех двоичных битов эквивалентна одной 16-ричной цифре.

### **AR область**

Область памяти, выделенная для флагов и битов управления.

### **ASCII**

American Standard code For Information Interchange. Служит для кодирования символов при выдаче на принтер и другие внешние устройства.

### **AUTOEXEC.BAT**

Файл MS DOS, содержащий команды, автоматически исполняемые при запуске.

### **СН**

В сообщениях ПК обозначает слово. См. Слово.

### **CONFIG.SYS**

Файл MS DOS, содержащий параметры среды персонального компьютера.

### **CTS**

Clear-to-send (сброс передатчика) - сигнал, используемый при связи между устройствами, который указывает, что приемник готов принимать данные.

### **СУ**

См. Флаг переноса

### **DIN профиль**

Профиль, который входит в монтажные канавки различных устройств, что позволяет на нем быстро и надежно монтировать устройства.

### **DIP-переключатель**

Dual-in-line package - корпус с двухрядным расположением (штырьковых) выводов, который монтируется на плате и служит для рабочих параметров.

### **DM область**

Область данных (Data memory) служащая для хранения только слов данных. Слова в области DM недоступны битами.

### **DM слово**

Слово в области DM

### **EEPROM**

ЭСПЗУ Электрически стираемое ПЗУ. Тип памяти, данные в которой можно стереть и переписать. Это делается через специальные входы микросхемы EEPROM и без удаления микросхемы с устройства, в котором она установлена.

### **EPROM**

СПЗУ Стираемое ПЗУ. Тип памяти, данные в которой можно стереть ультрафиолетом или другими способами, и переписать.

### **FA**

Factory automation - производственная автоматизация.

**FAL - ошибка**

Признак ошибки, вызванный из программы при исполнении команды FAL(06).

**FALS - ошибка**

Признак ошибки, вызванный из программы при исполнении команды FALS(07) или вызванный системой.

**FCS**

См. КСК

**HR область**

Область памяти, которая сохраняет данные при отключенном питании и используется как область рабочих бит.

**I/O**

(Input/output) - входы/выходы

**IBM PC/AT или совместимый**

Компьютер, имеющий аналогичную структуру с IBM PC/AT, на котором работают программы IBM PC/AT.

**IN**

(Input) - вход

**INTERLOCK**

Метод программирования, служащий для того, чтобы работать с несколькими командами, как с группой, чтобы можно было сбросить всю группу вместе, когда не требуются индивидуальные условия. Сблокированная секция нормально выполняется для условия исполнения 1 и сбрасывается при условии исполнения 0.

**JIS**

(Japanese Industrial Standards) - Японские промышленные стандарты.

**LR область**

Область данных, используемая при связи.

**LSS**

(Ladder support software) пакет программ, установленный на IBM PC/AT - совместимом компьютере для работы в качестве программирующего устройства.

**MONITOR**

Режим работы контроллера - отработка программы с возможностью вмешаться в ход отработки (принудительное включения/выключение входов/выходов). Служит для наблюдения или отладки ПК.

**OFF**

Состояние входа или выхода, когда сигнал отсутствует. Обычно говорят об уровне сигнала OFF (выключено) при низком входном напряжении, либо при непроводящем состоянии, но могут быть и противоположные случаи.

**ON**

Состояние входа или выхода, когда сигнал присутствует. Обычно говорят об уровне сигнала ON (включено) при высоком входном напряжении, либо при проводящем состоянии, но могут быть и противоположные случаи.

**OUT**

(Output) - выход

**PROGRAM**

Режим работы ПК - программирование, позволяющий ввод и отладку программ, но не позволяющий работу программы.

**PROM**

Programmable read-only memory: ППЗУ Программируемое постоянное запоминающее устройство. Тип ПЗУ, в котором программа или данные могут быть записаны пользователем, но потом эти данные сохраняются.

**PV**

(PRESENT VALUE) см. текущее значение

**RAM**

Read only memory - оперативное запоминающее устройство (ОЗУ). Не сохраняет данные при отключении питания.

**ROM**

Read-only memory: ПЗУ Постоянное запоминающее устройство. Тип ПЗУ, в которое писать нельзя. Микросхема ROM выпускается в уже готовыми данными или программой и их нельзя изменить. Однако программу или данные можно читать сколько угодно раз.

**RUN**

Режим работы контроллера - рабочий режим, работа по уже отлаженной программе.

**SR область**

Область памяти, которая содержит флаги и другие биты/слова со специальными функциями.

**SSS**

(SYSMAC support software) пакет программ, установленный на IBM PC/AT - совместимом компьютере для работы в качестве программирующего устройства.

**SV**

(SET VALUE) См. заданное значение

**TR бит**

Бит в области TR.

**TR область**

Область памяти, служащая для хранения условий исполнения, так что их можно будет загрузить позже для использования другими командами.

**UM область**

Область памяти, в которой хранится программа пользователя, т. Е. Программа, которая исполняется в настоящее время.

**адрес**

Число, служащее для указания места параметра или команды в памяти.

**адрес бита**

Место в памяти, где находится бит данных. Адрес бита содержит название области памяти, адрес слова и номер бита внутри слова.

**адрес слова**

Место в памяти, где хранятся слова. Адрес слова должен указывать (иногда берется по умолчанию) область данных и номер слова, к которому производится адресация.

**арифметический сдвиг**

Операция сдвига, при которой флаг переноса включается в операцию сдвига.

**базовые команды**

Основные команды лестничной диаграммы (в отличие от дополнительных команд)

**байт**

Единица данных, равная 8 битам, или половине слова.

**бит**

Самая малая единица информации, которую можно представить в вычислительном устройстве. Бит может иметь значение либо 1, либо 0, соответствующее электрическим сигналам ВКЛ или ВЫКЛ. Бит представляется одной двоичной цифрой. Некоторые биты ПК отведены для специальных целей, таких, как сохранение состояния входов от внешних устройств, а некоторые можно использовать для общего назначения в программировании.

**бит входа/выхода**

Бит в памяти ПК, использующийся для сохранения состояния входа/выхода. Биты входа отражают состояние реальных входов (входных зажимов).

Биты выхода содержат состояние для посылки на реальные входы (выходные зажимы).

**бит для разового включения**

Бит, включаемый в 1 или 0 на заданный интервал времени, больше, чем 1 цикл.

**бит перезапуска**

Бит, используемый для перезапуска ПК.

**бит самоподдержки**

Бит, который запрограммирован для поддержания состояния либо 0 либо 1 до установки или сброса с помощью заданных условий.

**битовый операнд**

Бит, заданный как операнд для команды.

**блок**

В терминологии OMRON, сборочная единица ПК.

**блок**

См. Логический блок и блок команд

**блок входа/выхода**

Блок ПК, физически подключенные к входным/выходным устройствам для ввода и выдачи сигналов. Включают блоки входа и блоки выхода, несколько модификаций входных и выходных.

**блок команд**

Группа команд, логически связанные на ЛД. Логический блок включает все командные линии, соединяющиеся друг с другом, от одной или более линий подключенных к левой шине, к одной или более выходных (правых) команд, подключенных к выходной (правой) шине.

**блок питания**

Блок, подключенный к ПК, обеспечивающий напряжение, требуемой другими блоками.

**блочный ПК**

ПК, составленный из отдельных компонентов, или блоков. У данных ПК отдельный блок не идентифицируется как ПК. ПК является функциональным объединением блоком

**вложенность**

Программирование одного цикла в другом, программирование вызова подпрограммы из самой подпрограммы, или программирование одного перехода внутри другого.

**возврат**

Процесс, когда исполнение команд возвращается из подпрограммы в главную программу (обычно в точку, из которой был вызов подпрограммы).

**возвращение**

Процессы копирования данных либо с внешнего устройства, или из области хранения в активную область системы, такую как буфер дисплея.

**возобновление**

Процесс, в котором устройство будет снова пытаться передавать данные после получении ошибки из приемника.

**время исполнения**

Время, требующееся для ЦПУ для выполнения либо отдельной команды, либо всей программы.

**время исполнения команды**

Время, требуемое для исполнения команды. Время для любой команды может варьировать от условий исполнения и используемых операндов.



**время контроля за ответом**

Время, которое устройство ожидает при пересылке данных до тех пор, пока не будет уверенности, что произошел сбой.

**время реакции на вход**

Время, требующееся для выдачи выходного сигнала с ПК в ответ на входной сигнал, полученный с внешнего устройства.

**время сканирования**

См. Время цикла

**время цикла**

Время необходимое для осуществления одного цикла ПК.

**вход**

Сигнал, приходящий с внешнего устройства на ПК. Термин часто используется как общий термин для обозначения входных сигналов.

**вход уменьшения счетчика**

Сигнал входа, служащий для декрементирования счетчика при смене сигнала с 0 на 1.

**входное устройство**

Внешнее устройство, посылающее сигнал в ПК.

**входное/выходное устройство**

Устройство, подключенное к входным или выходным разъемам ПК. Может быть либо частью управляющей системы, либо функционирует для помощи в управлении другими устройствами, либо частью управляемой системы

**входной бит**

Бит области IR, выделенный для хранения состояния входа.

**входной сигнал**

Изменение состояния входа ПК. Обычно говорят об активном сигнале, когда состояние входа изменяется с низкого на высокое напряжения, либо из непроводящего в проводящее состояние.

**входной сигнал сдвига**

Входной сигнал, переход которого из 0 в 1 вызывает сдвиг данных на 1 бит.

**вызов (подпрограммы)**

Процесс, при котором исполнение команд переходит из главной программы к подпрограмме. Подпрограмму можно вызвать либо из программы, либо по прерыванию.

**выход**

Сигнал, посылаемый из ПК на внешнее устройство. Термин часто используется как общий термин для обозначения выходных сигналов.

**выходное устройство**

Внешнее устройство, принимающее сигнал от ПК.

**выходной бит**

Бит области IR, выделенный для хранения состояния, которое должно быть послано на выходное устройство.

**выходной сигнал**

Сигнал, посылаемый на выходное устройство. Обычно говорят о том, что выходной сигнал существует, когда состояние точки выхода изменяется с низкого на высокое напряжения, либо из непроводящего в проводящее состояние.

**выходные команды**

Команды, помещаемые с правой стороны на ЛД и использующие последние условия исполнения в командной линии.

**главная программа**

Вся программа, за исключением подпрограмм и программ прерываний.

**граница области данных**

Самый большой адрес в области данных Назначая операнд, требующий нескольких слов, необходимо убедиться, что не превышен верхний адрес области данных.

**двоично-десятичное вычисление**

Математическое вычисление, использующее числа, представленные в двоично-десятичном виде.

**двоично-десятичное представление числа**

Система представления чисел, когда четыре двоичные бита эквивалентны десятичной цифре.

**двоичное вычисление**

Математическое вычисление, использующее числа, представленные в двоичном виде.

**двоичное представление числа**

Система представления чисел, когда числа записываются только цифрами 0 и 1. Каждая группа из четырех двоичных бит эквивалентна одной 16-ричной цифре. Двоичные данные в памяти для удобства выражают в 16-ричном виде.

**двоичное число без знака**

Двоичное значение, загруженное в память без указания знака.

**двоичное число со знаком**

Двоичное значение, хранящееся в памяти, с битом, который указывает на знак (+ или -).

**декремент**

Уменьшение (обычно на 1)

**десятичное представление числа**

Система счисления, в которой числа представляются на базе 10. В ПК все данные в конце концов хранятся в двоичной форме, 4 двоичных бита часто используются для представления одной десятичной цифры (двоично-десятичная система).

**десятичное число с плавающей точкой**

Десятичное число, представленное как число (мантисса), умноженное на 10 в степени, напр.  $0.538 \times 10^{-5}$ .

**диск данных**

Дискета, служащая для сохранения программы пользователя, содержания области DM и других данных пользователя.

**дискрета**

Единица измерения

**дистанция передачи**

Расстояние, на которое можно передать сигнал.

**длина данных**

В связи, количество бит, которые будут восприниматься как единый блок при передаче данных.

**загрузка**

Процесс копирования данных либо с внешних устройств, либо с области сохранения, в активную часть системы, такую, как буфер дисплея.

**загрузка на верхний уровень**

Процесс передачи программы или данных с низшего уровня или ведомого компьютера на верхний уровень или ведущий компьютер. Если работает программирующее устройство, оно считается управляющим компьютером.

**загрузка с верхнего уровня**

Процесс передачи программы или данных компьютера верхнего уровня, или ведущего компьютера, на ведомый компьютер. При работе с программирующим устройством, оно считается ведущим компьютером.

**заданное значение**

Значение, от которого декрементирующий счетчик начинает отсчет или до которого досчитывает инкрементирующий счетчик (т. Е. Максимальное число) или время, от которого или до которого таймер начинает отсчет.

**задержка OFF (0)**

Задержка от момента, когда сигнал на передающей стороне выключился в 0 (например, на входном устройстве или ПК) и моментом, когда сигнал воспримется как 0 на принимающей стороне (например, на выходном устройстве или ПК).

**задержка ON (1)**

Задержка от момента, когда сигнал на передающей стороне включился в 1 (например, на входном устройстве или ПК) и моментом, когда сигнал воспримется как 1 на принимающей стороне (например, на выходном устройстве или ПК).

**задержка включения входов/выходов**

Либо задержка от посылки сигнала на выход до включения реального выхода

Либо задержка от изменения состояния на реальном входе до того, как сигнал изменении будет воспринят в пк.

**замаскированный бит**

Бит, состояние которого временно становится неэффективным.

**запрограммированное предупреждение**

Предупреждение, выдаваемое в результате выполнения команды, предназначенной для вызова предупреждения, в отличие от предупреждений, выдаваемых системой.

**запрограммированный признак ошибки**

Признак ошибки, выдаваемый в результате выполнения команды, предназначенной для вызова признака ошибки из программы, в отличие от признака ошибки, выдаваемого системой.

**зарезервированное слово**

Слово в памяти, зарезервированное для специальных задач, и недоступный для пользователя.

**зарезервированный бит**

Бит, недоступный для пользователя.

**защита от записи**

Состояние, в котором содержимое устройства хранения информации нельзя изменить.

**И**

Логическая операция над условиями, при которой результат является истиной, если только оба условия истинны. В программировании лестничных диаграмм условия - это обычно состояния бит 1/0 или логическая комбинация таких состояний, называемая условиями исполнения.

**ИЛИ**

Логическая операция над условиями, при которой результат является истиной, если хотя бы одно или два условия истинны. В программировании ЛД условия - это обычно состояния битов (1 или 0) или логическая комбинация таких состояний, называемых условиями исполнения.

**импульс управления**

Сигнал, который подсчитывает счетчик.

**инверсное условие**

См. Нормально закрытое условие.

**инициализация**

Часть процесса пуска, в котором очищается часть адресов памяти, проверяются установочные параметры системы и устанавливаются значения по умолчанию.

**инкремент**

Увеличение на 1

**инкрементирующий вход счетчика**

Входной сигнал, служащий для увеличения счетчика на 1 при изменении сигнала с 0 на 1.

**инсталляция**

Подготовка, необходимая для использования программы или пакета программ, таких как LSS или SSS , на компьютере.

**интерфейс**

Стык между системами или устройствами и обычно включает изменение в способе представления данных. Устройства интерфейса выполняют такие операции, как кодирование, форматирование данных.

**интерфейс RS-232C**

Промышленный стандартный интерфейс для связи.

**интерфейс с управляющим компьютером**

Интерфейс, позволяющий осуществлять связь с управляющим компьютером.

**исключающее ИЛИ**

Логическая операция над условиями, при которой результат является истиной, если истинно одно, и только одно условие. В программировании ЛД условия - это обычно состояния битов (1 или 0) или логическая комбинация таких состояний, называемых условиями исполнения.

**исключающее ИЛИ НЕ**

Логическая операция над условиями, при которой результат является истиной, если только оба условия истинны или оба условия ложны. В программировании ЛД условия - это обычно состояния битов (1 или 0) или логическая комбинация таких состояний, называемых условиями исполнения.

**источник**

Ячейка (ячейки) памяти, откуда команда берет данные, над которыми она совершает действие, в отличие от ячеек, куда помещаются данные. Ячейка (ячейки) памяти, куда помещаются результаты, называется приемник.

**кабель связи**

Кабель для передачи данных между участками системы, отвечающий стандартам RS-232C или RS-422.

**кадр**

Блок данных для приема/передачи, имеющий начало (символ @), код заголовка, информацию, контрольную сумму и оканчивается кодом окончания (терминатор - символы \* ↵)

**код имени команды**

Код имени команды, который указывает, какую команду выполнять

**код ответа**

Код, посылаемый в ответ на передачу данных, который указывает, как прошла передача данных.

**код ошибки**

Цифровой код, выданный для индикации того, что произошла ошибка, и некоторая информация о природе ошибки. Некоторые ошибки генерируются системой; другие задаются в программе оператором.

**код символа**

Числовой (обычно двоичный) код, служащий для представления буквенно-цифрового символа.

**команда пересылки данных**

Команда для пересылки данных из одного места памяти в другое. Данные в источнике не изменяются.

**команда включения на 1 цикл**

Команда, включающая бит операнда в 1 только на 1 цикл, когда условие исполнения на входе изменяется либо с 0 на 1 (DIFU) или с 1 на 0 (DIFD).

**команда ЛД**

Команда в программе, указывающая ПК на действие, которое должно быть произведено, и на данные, которые нужно при этом использовать. Команды можно использовать для простой установки бит в 0 или 1, или для выполнения более сложных операций, таких как преобразование и/или передачи больших блоков данных.

**команда лестничной диаграммы**

Команда, которая представляет условия на ЛД. Другие команды располагаются вдоль правой шины ЛД и называются выходными командами.

**команда логического блока**

Команда для логического объединения условий-результатов логического блока с текущим условием. Текущее условие может быть либо результатом одного условия, либо другого логического блока. Командами логического блока являются команды and load и or load.

**команда сравнения**

Команда, служащая для сравнения данных в различных местах памяти для определения соотношения между этими данными.

**команда, управляющая битами**

Команда, которая управляет состоянием отдельного бита (в отличие от команды, управляющей состоянием целого слова).

**команда фронта 0/1**

Команда, которая выполняется только один раз, когда условие исполнения на входе изменяется с 0 на 1. Команда, срабатывающая не по фронту, выполняется в каждом цикле, пока условие исполнения на входе = 1.

**командная линия**

Группа условий, лежащих на одной горизонтальной линии ЛД. Командные линии могут разветвляться или соединяться для образования блоков команд. Также называется ступенькой.

**константа**

Операнда, в котором указана сама величина (в отличие от адреса). Обозначается символом #.

**контроль на четность - нечетный**

Настройка при связи, когда число бит в состоянии 1 всегда будет нечетным.

**контроль на четность - четный**

Настройка при связи, когда число бит в состоянии 1 всегда будет четным.

**контроль четности**

Проверка на четность для определения, не испорчены ли данные при передаче.

**контрольная сумма**

Сумма, которая передается при связи вместе с пакетом данных. КС пересчитывается для принимаемых данных для того, чтобы убедиться, что полученные данные не запорчены.

**контрольный таймер**

Таймер в системе, который обеспечивает чтобы время сканирования оставалось в заданных рамках. При переходе за эти границы либо выдается либо предупреждение, либо по достижении определенного значения останавливается работа ПК.

**конфигурация ПК**

Совокупность блоков ПК, объединенных в одно устройство.

**конфигурация системы**

Порядок, в котором соединены блоки. Данный термин относится ко всем блокам и устройствам.

**косвенный адрес**

Адрес, содержимое которого указывает на адрес. Содержимое второго адреса будет использовано как операнд.

**КСК (FCS)**

(Frame checksum) - Контрольная сумма конверта. Результат выполнения операции ИСКЛЮЧАЮЩЕЕ ИЛИ с заданной зоной данных. КСК можно подсчитать на передающем и приемном устройствах при передаче данных, чтобы убедиться, что данные переданы правильно.

**ЛД**

(Лестничная диаграмма) См. Релейно-контактная схема

**лестничная диаграмма**

См. Релейно-контактная схема

**линия связи управляющим компьютером**

Линия связи, связывающая ПК с управляющим компьютером для возможности наблюдения или управления с управляющего компьютера.

**логическая команда**

Команда, служащая для логического объединения содержания двух слов и выдачи логического результата в заданное слово результата. Логические команды объединяют все биты с одинаковыми номерами в двух словах и выдают результат в бит с тем же номером в заданном слове результата.

**логический блок**

Группа команд, логически связанных на ЛД, и требующих блоковых команд для связи их с другими командами или логическими блоками.

**маскирование**

“Закрытие” сигнала прерывания так, что вызов прерывания не действует до удаления маски.

**мегабайт**

Единица хранения информации, равная миллиону бит.

**мерцающий бит (flicker bit)**

Бит, который запрограммирован на включение в 1 и 0 с заданной частотой.

**младший бит(слово)**

Младший из группы битов, обычно во всем слове, или группы слов.

**мнемокод**

Форма программирования ЛД, который состоит в списке команд без использования схемы.

**мощность переключения**

Максимальные напряжение/ток, которые реле может переключать в 1 и 0.

**настройка четности**

Настройка количества бит в состоянии 1 в слове или другой единице данных, так что общее количество всегда является либо четным, либо нечетным числом. Контроль на четность обычно используется для проверки правильности данных после передачи для определения, осталось ли число битов в состоянии 1 четным или нечетным.

**НЕ (NOT )**

Логическая операция, инвертирующая состояние операнда. Например, AND NOT указывает, на операцию И с состоянием бита, противоположным текущему.

**нефатальная ошибка**

Ошибка аппаратной или программной части, вызывающая предупредительное сообщение, но не останавливающая работу ПК.

**номер бита**

Число, указывающее положение бита в слове. Бит 00 - самый правый (младший) бит; бит 15 - самый левый (старший) бит.

**номер блока**

Номер, приписанный каждому блоку, для облегчения его идентификации при обращении.

**номер перехода**

Определитель, используемый с командой перехода, для указания точек, с которой и на которую должен быть переход.

**номер прерывания**

Определитель, служащий для идентификации подпрограммы, чтобы можно было осуществить вызов подпрограммы.

**номер сообщения**

Номер сообщения, вызванного командой MESSAGE.

**нормально закрытый вход**

Вход, который нормально закрыт, т. Е. Входной сигнал считается активным, когда входная цепь разомкнута.

**нормально открытый вход**

Вход, который нормально открыт, т. Е. Входной сигнал считается активным, когда входная цепь замкнута.

**область**

Секция в памяти, имеющая свое имя и буквенное обозначение. Есть область данных и область памяти.

**область протокола ошибок**

Область, служащая для хранения записей, в которой указаны время и причина ошибок, которые произошли в системе.

**область связь по линии данных**

Область общих данных, связанная с линией данных.

**область данных**

Область в памяти ПК, выделенная для хранения заданного типа данных.

**область только для чтения**

Область памяти, откуда пользователь может только читать состояние бит, но не может туда писать

**обновление**

Процесс обновления состояния выходов на внешние устройства, чтобы они соответствовали битам выхода в памяти, и обновление входных битов в памяти, чтобы они соответствовали состоянию реальных входов.

**обновление входов/выходов**

Процесс обновления состояния выходов на внешние устройства, чтобы они соответствовали битам выхода в памяти, и обновление входных битов в памяти, чтобы они соответствовали состоянию реальных входов.

**обработка события**

Обработка, вызванная как реакция на событие, например, на прерывание.

**обслуживание**

Процесс, при котором ПК проверяет коннектор или блок, чтобы определить, не требуется ли обслуживание.

**обслуживание периферии**

Обслуживание сигналов с и на периферийные устройства, включая обновление, связь и т. д.

**общая диспетчеризация**

Часть времени цикла ПК, включающая отработку общих задач, требуемых при работе ПК.

**общее число входов/выходов**

Общее число входов/выходов, которыми может управлять ПК. Данное число меняется от около ста для малых ПК до 2 тысяч для самых больших.

**общие данные**

Данные в памяти ПК, к которым имеет доступ другой ПК из одной системы. У каждого ПК выделена специальная область памяти. Каждой ПК записывает в выделенную секцию и читает из секции, выделенной другим ПК, с которым у него общие данные.

**он-лайн**

Режим работы компьютера по линии связи с другим устройствами

**онлайнное редактирование**

Процесс коррекции программы в ПК прямо с программирующего устройства. Возможно только в режимах PROGRAM или MONITOR. В режиме MONITOR программу можно менять во время исполнения.

**операнд**

Значения, приписанные как данные для команды. Операнд может быть либо константой, представляющей числовое значение, которое будет использоваться командой, либо адресом, в котором находятся данные, которые будут использоваться.

**описатель бита**

Операнд, служащий для описания бита в слове, который будет использован в качестве операнда команды.

**описатель цифры**

Операнд, использующийся для назначения цифры или цифр слова, которые используются командой.

**определитель**

Число, служащее как операнд для команд, но которое служит только для определения самой команды, а не данных, которыми оперирует команда. Определители включают номера переходов, номера подпрограмм и т. Д.

**отладка**

Процесс, в котором черновой вариант программы корректируется до тех пор, пока программа не заработает как запланировано. Отладка включает устранение синтаксических ошибок и точную настройку времен и координацию операций управления.

**отрицательная задержка**

Задержка, установления для трассировки данных, при которой регистрация данных начинается на заданное время перед сигналом трассировки.

**оф-лайн**

Режим работы компьютера автономно без связи с другим устройствами

**ошибка аппаратной части**

Неисправность, причина которой аппаратной части электрических компонентов ПК, в отличие от ошибок программной части, причина которых - в программа.

**ошибка инициализации**

Ошибка, произошедшая в аппаратной либо программной части при пуске ПК.

**ошибка синтаксиса**

Ошибка в написании программы. Ошибки могут включать

Ошибки правописания (т. Е. Несуществующий функциональные код )

Ошибки в задании операндов при допустимых параметрах (например, задать для записи биты, в которых возможно только чтение)

Ошибки в применении команд (например, вызов несуществующей подпрограммы).



**ошибка системной программы**

Ошибка, вызванная системной программой.

**память трассировки**

Область памяти, служащая для хранения результатов операции трассировки.

**параметр управления**

Операнд, в котором задано, как выполнять команду. Параметр управления может указывать часть слова, которая используется в качестве операнда, может определять приемник для команды пересылки данных, может указывать размер таблицы данных, используемых в команде и т.д.

**передача**

Процесс передачи данных из одного места в другое в пределах ПК, или между ПК и внешними устройствами. При передаче данных, посылается обычно копия данных, т. Е. Содержимое источника остается неизменным.

**передача данных**

Передача данных из одного места в памяти в другое, либо в пределах одного устройства, либо между разными устройствами, соединенными линией связи или сетью.

**передний фронт импульса**

Точка, где сигнал изменяется с 0 на 1

**переключатель защиты от записи**

Переключатель, служащий для защиты от записи содержимого устройства хранения информации, например, гибкий диск. Если отверстие в левом верхнем углу открыто, информация на нем нельзя изменить.

**переписать**

Изменить содержание ячеек памяти с потерей прошлого состояния.

**переполнение**

Состояние, при котором превышена емкость памяти, отведенная под хранение данных.

**переход**

Тип программирования, когда исполнение переходит с одной точки программы к другому без выполнения команд между ними.

**периферийное устройство**

Устройство, подключенное к ПК для помощи в работе системы. Периферийные устройства включают принтеры, программирующие устройства, внешние устройства памяти и т. Д.

**ПК**

Программируемый контроллер

**по умолчанию**

Значение, автоматически задаваемое ПК, когда пользователь не указывает другое значение. Большинство устройств принимает значение по умолчанию при включении питания.

**подпрограмма**

Группа команд, расположенных отдельно от главной программы и исполняемая только при вызове из главной программы или из прерывания.

**подсказка**

Сообщение или символ, который появляется на дисплее для того, чтобы потребовать ввода от оператора.

**положительная задержка**

Задержка, установленная для трассировки данных, при которой регистрация данных начинается на заданное время после сигнала трассировки.

**порт**

Разъем на ПК или компьютере, служащий для связи с внешними устройствами.

**последовательное подключение**

Соединение, при котором блоки связаны в цепь.

**правая команда**

Команда, расположенная на ЛД справа

**прерывание (сигнал)**

Сигнал, останавливающий нормальное исполнение программы и вызывающий подпрограмму или другие процессы отработки.

**прерывание входов/выходов**

Прерывание, вызванное сигналом со входов/выходов.

**прерывание по расписанию**

Прерывания, которые вызываются автоматически системой через заданные интервалы времени или из места программы, указанного оператором. Прерывание по расписанию вызывают выполнение заданной подпрограммы, которую можно использовать для выполнения команд, которые должны выполняться через заданные интервалы времени.

**префикс области**

Префикс из одной или двух букв для обозначения области памяти в ПК. Все области памяти, за исключением областей IR и SR, при адресации требуют префикса.

**приемник**

Ячейка (ячейки) памяти, куда команда помещает данные, над которыми она совершила действие, в отличие от ячеек, откуда берутся данные. Ячейка (ячейки) памяти, откуда команда берет данные, над которыми она совершает действие, называется источник.

**Принудительная установка**

Процесс принудительной установки в 1 бита с программирующего устройства. Биты обычно устанавливаются в 1 как результат исполнения программы.

**Принудительный сброс**

Процесс принудительного сброса в 0 бита с программирующего устройства. Биты обычно устанавливаются в 0 как результат исполнения программы.

**программа прерывания**

Программа, выполняющаяся как реакция на прерывание.

**программатор**

Переносное программирующее устройство для ПК

**программируемый контроллер**

Вычислительное устройство, способное принимать входные сигналы с внешних устройств и выдавать выходные сигналы на внешнее устройство согласно программы, загруженной в память. Пк используются для автоматизации внешних устройств. Хотя есть пк в виде единого устройства, чаще пк собираются из отдельных блоков.

**программирующее устройство**

Периферийное устройство, служащее для ввода программы в ПК или для изменения или контроля программы, уже загруженной в ПК. Есть специальные программирующие устройства, такие, как программатор, и неспециализированные, такие, как управляющий компьютер.

**программная защита**

Средство защиты данных (от возможных изменений) при помощи программных средств, в отличие от реальных ключей или других аппаратных средств.

**протокол**

Параметры и процедуры, стандартизированные для того, чтобы сделать возможным связь двух устройств, или сделать возможным связь оператора или программатора с устройством.

**прямой выход**

Метод, при котором результаты исполнения программы выдаются не на биты выхода, прямо на реальный выход для устранения влияния времени цикла.

**рабочая область**

Часть памяти, содержащая рабочие слова/биты.

**рабочее слово**

Слово, которое можно использовать для вычисления данных или других операций, т. Е. “рабочее пространство” в памяти. Большая часть области *ig* всегда зарезервированы как рабочие слова. Часть других областей, не используемые для специальных целей, используются в качестве рабочих битов.

**рабочий бит**

Бит в рабочем слове

**размаскирование**

“Открытие” замаскированного сигнала прерывания так, что вызов прерывания действует и прерывание будет обрабатываться.

**размаскированный бит**

Бит, чье значение неэффективно. (см. Замаскированный бит)

**распределенное управление**

Концепция автоматизации, при которой участок системы автоматизации располагается рядом с управляемыми устройствами, т. Е. Управление децентрализуется и “распределяется” по системе. Распределенное управление - концепция, основанная на системе ПК.

**расширенный счетчик**

Счетчик, созданный в программе с последовательным использованием двух и более команд счетчиков. Такой счетчик способен считать большие значения, чем позволяют стандартные команды.

**расширенный таймер**

Таймер, созданный в программе с последовательным использованием двух и более команд таймеров. Такой таймер способен отсчитывать большие интервалы времени, чем позволяют стандартные команды.

**реверсивный регистр сдвига**

Регистр сдвига, который может сдвигать данные в обоих направлениях в зависимости от заданных условий.

**реверсивный счетчик**

Счетчик, который может и увеличивать и уменьшать свое значение в зависимости от заданных условий.

**регистр сдвига**

Одно или более слов, в которых данные сдвигаются на заданное число дискрет вправо или влево с дискретами бит, цифра или слово. В регистре циклического сдвига данные выдвигаются из одного конца и вдвижутся в другой конец регистра. В других регистрах сдвига новые данные (указанные данные, нуль (нули) или единица (единицы)) вдвижутся в один конец, а выдвигаемые с другого конца теряются.

**регулярный импульс**

Импульс на специально отведенном выходе, служащий для организации таймерных операций. Имеются импульсы различной ширины и, следовательно, различной частоты.

**режимы работы**

Один из трех режимов работы ПК: RPROGRAM, MONITOR, RUN.

**резервная копия**

Копия существующих данных, которая сохранится в случае искажения или потери исходных данных

**релейное управление**

Предшественник ПК. При релейном управлении группы реле соединялись для образования цепей управления. В ПК это делается программированием.

**релейно- контактная схема**

Форма программирования, появившаяся из релейных систем управления, которая использует релейные символы для представления алгоритма работы программы. Внешний вид программы похож на лестницЦПУ, отсюда и названия.

**РКС**

См. релейно- контактная схема

**самодиагностика**

Процесс, при котором система сама проверяет свою работу и вызывает предупреждение или признак ошибки в случае ненормальной работы.

**сбой при работе**

Ошибка, которая происходит при работе ПК, в отличие от ошибок инициализации, которая появляется перед фактическим началом работы.

**сброс (RESET)**

Процесс установки бита или сигнала в 0 или изменения текущего значения таймера и счетчика в заданное значение или 0.

**связь по линии данных**

Операция автоматической передачи данных, что позволяет ПК или блокам ПК обмениваться данными с помощью области общих данных.

**сигнал управления**

Сигнал, посылаемый из ПК, для воздействия на операцию системы управления.

**символ лестничной диаграммы**

Символ, используемый для изображения программы в виде ЛД.

**синтаксис**

Форма программы (в отличие от значения)

**синхронное исполнение**

Исполнение программ и операций обслуживания при котором данные работы синхронизированы, так что все операции обслуживания выполняются каждый раз при исполнении программы.

**система ПК**

ПК со всеми блоками, подключенными к внешним устройствам. Границами системы ПК являются:

Сверху: ПК и программа и ЦПУ

Снизу: блоки ПК

**система управления**

Все компоненты (аппаратные и программные), служащие для управления другими устройствами. Система управления включает ПК, программы ПК и устройства входа/выхода, которые служат для управления или обратной связи с управляемой системой.

**системная ошибка**

Ошибка, вызванная системой, в отличие от вызванной исполнением команды, предназначенной для вызова признака ошибки.

**сканирование**

Процесс исполнения программы ЛД. Программа последовательно просматривается с начала до конца и каждая команда выполняется по очереди в зависимости от условий исполнения.

**скорость передачи данных**

Скорость передачи данных операции связи между двумя устройствами, единица измерения - бит/с

**словный операнд**

Слово, заданное как операнд для команды.

**слово**

Единица данных, состоящая из 16 бит. Все области данных состоят из слов. Некоторые области данных доступны только словами. Другие доступны только битами и словами.

**слово входа/выхода**

Слово в области IR, выделенное блоку входов/выходов и использующееся для хранения состояния блока входов/выходов.

**слово результата**

Слово, используемое для помещения результата от исполнения команды

**смещение**

Положительная или отрицательная величина, добавляемая к базе, такой как адрес, для указания на требуемое значение.

**создание общей области данных**

Процесс, при котором создаются области общих данных между двумя или более ПК.

**сообщение о системной ошибке**

Сообщение об ошибке, вызванной системой, в отличие от сообщения, вызванного исполнением команды, предназначенной для вызова сообщения.

**сохранение**

Процесс записи в память для постоянного хранения программы, которая написана в буфере дисплея.

**специальная команда**

Команда, вводимая своим функциональным кодом, в отличие от базовых команд, которые составляют основу ЛД.

**старший бит(слово)**

Старший из группы битов, обычно во всем слове, или группы слов.

**ступенька**

См. Командная линия

**счетчик**

Специально выделенная группа цифр или слов в памяти, служащая для подсчета количества, сколько раз произошел указанный процесс; или область в памяти, доступная битам таймера и счетчика и служащая для подсчета того, сколько раз условие исполнения изменилось с 0 на 1.

**таймер**

Ячейки памяти, доступные командами TIM/CNT и отсчитывающие время вниз от заданного значения. Таймеры включаются в 1 и сбрасываются в соответствии с условиями исполнения.

**текущее значение**

Текущее значение, зарегистрированное в устройстве при работе. Обычно применяется к таймерам и счетчикам.

**терминатор**

Код окончания запроса или ответа при передаче данных

**точка входа**

Место, в котором входной сигнал входит в ПК. Физически соответствуют клеммам или ножкам разъемов на блоке.

**точка входа/выхода**

Место, в котором входной сигнал входит в систему ПК, или в котором выходной сигнал выходит из системы ПК. Физически соответствуют клеммам или ножкам разъемов на блоке. С точки зрения программы, соответствуют битам входов/выходов в области IR.

**точка выхода**

Место, в котором выходной сигнал выходит из ПК. Физически соответствуют клеммам или ножкам разъемов на блоке.

**трассировка**

Операция, при которой выполняется программа и результаты сохраняются для последующего пошагового анализа и отладки.

**трассировка адреса**

Процесс, в котором при исполнении программы регистрируются изменения в указанном месте памяти.

**трассировка данных**

Процесс, когда при исполнении программы регистрируются изменения содержания указанных участков памяти.

**умолчание**

См. По умолчанию

**УПК**

Установочные параметры программируемого контроллера

**управляющий бит**

Бит в области памяти, устанавливаемый либо из программы, либо с программирующего устройства для целей управления, например, бит перезапуска включается в 1 и 0 для перезапуска блока.

**управляющий компьютер**

Компьютер, использующийся для передачи или приема данных с ПК в управляющей системе Host Link. Управляющий компьютер служит для управления данными и управления системой. В качестве управляющего компьютера обычно используется персональный компьютер или бизнес-компьютер.

**условие**

Символ на командной линии ЛД, которая управляет выходной командой. Каждому условию приписан в памяти бит, который определяет его состояние. Состояние бита определяет следующее условие исполнения. Условия соответствуют командам LOAD, LOAD NOT, AND, AND NOT, OR или OR NOT.

**условие исполнения**

Состояние 1 или 0, при которых выполняется команда. Условие исполнения задается логической комбинацией условий на одной командной линии лд (до исполняемой команды).

**установка**

Процесс установки бита или сигнала в 1.

**установочные параметры ПК**

Группа параметров, управляющих работой ПК, задаваемых с программирующего устройства,

**установочные параметры системы**

Установочные параметры системы для программирующего устройства, например, LSS или SSS.

**фатальная ошибка**

Ошибка, вызывающая остановку работы ПК и требующая исправления перед продолжением работы

**флаг**

Специальный бит в памяти, устанавливаемый системой для индикации состояния определенного типа операций. Некоторые флаги, такие, как перенос, может установить оператор, а также можно устанавливать из программы.

**флаг завершения**

Флаг, используемый таймерами и счетчиками, который устанавливается в 1, когда таймер отсчитал заданное время или счетчик отсчитал заданное значение.

**флаг переноса**

Флаг, служащий в математических операциях для сохранения переноса при операциях сложения или умножения, или для указания того, что при вычитании результат отрицателен. Флаг переноса также используется в некоторых операциях сдвига.

**формат ответа**

Формат, задающий данные, требуемые в ответе при передаче данных

**Функциональный код**

2-разрядный код, служащий для ввода команд в ПК.

**цикл**

Повторяющийся цикл работы ЦПУ, включающий отработку ЛД, обслуживание периферийного порта, обновление входов/выходов и т. Д.

**цикл исполнения**

Повторяющийся цикл работы ЦПУ, включающий отработку ЛД, обслуживание периферии, обновление входов/выходов и т. Д.

**циклический регистр сдвига**

Регистр сдвига, в котором данные, выдвигаемые с одного конца, вдвигаются в регистр с другого конца.

**циклическое прерывание**

См. Прерывание по расписанию

**цифра**

Единица представления чисел, состоящая из 4 битов.

**ЦПУ**

Центральное устройство контроллера. Устройство, способное хранить программу и данные, а также выполнять команды, содержащиеся в программе. В программируемом контроллере ЦПУ выполняет программу, обрабатывает входные/выходные сигналы, осуществляет связь с другими устройствами и т. Д.

**ЧС**

Часовая стрелка (в сочетаниях: по ЧС и против ЧС)

**шина**

Линия связи, служащая для обмена данными между подключенными к ней блоками.

**шина ЛД**

Линия на лестничной диаграмме, идущая сверху вниз обычно с левой, а иногда и с правой стороны ЛД. Исполнение команд идет производится сверху вниз вдоль шины ЛД, которая является исходной точкой для всех командных линий.

**электрические помехи**

Случайные изменения в электрических характеристиках, таких, как напряжение, ток, что может помешать нормальной работе устройства.

**ЭСПЗУ**

Электрически стираемое ПЗУ





*Уважаемые Пользователи!*

*Данное Руководство постоянно совершенствуется.*

*В случае, если у Вас будут какие-либо замечания к данному Руководству, просим Вас сообщать о них по следующим телефонам в г. Минске:*

*017 / 229 24 22, 229 28 89*

*или E-Mail: witaly\_z@hotmail.com*

**OMRON**

**OMRON EUROPE B.V.**

Представительство в СНГ

Россия, 107005 Москва

Бригадирский пер. 6

Тел. (095) 258 62 20, 258 62 21

Факс (095) 258 62 80

